A MULTIPLE-LOOP PRIMARY SYSTEM MODEL FOR PRESSURIZED WATER REACTOR PLANT SENSOR VALIDATION

Vol 1

by

SHIH-PING KAO

B. S. Georgia Institute of Technology
(June 1979)

S. M. Massachusetts Institute of Technology
(June 1981)

Submitted to the Department of
Nuclear Engineering in partial
fulfillment of the requirements
for the Degree of

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

July 1984

Signature of Author ______________________
Department of Nuclear Engineering
July 1984

Certified by ______________________
Professor John E. Meyer
Thesis Supervisor

Certified by ______________________
Doctor John H. Hopps, Jr.
Co-Thesis Supervisor

Accepted by ______________________
Professor Allan F. Henry, Chairman
Departmental Graduate Committee

Vol 1

A MULTIPLE-LOOP PRIMARY SYSTEM MODEL FOR PRESSURIZED WATER REACTOR PLANT SENSOR VALIDATION

by

Shih-Ping Kao

Submitted to the Department of Nuclear Engineering on July 3, 1984, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Nuclear Engineering

ABSTRACT

The objective of this research is to develop a faster-than-real time thermal-hydraulic simulation model to assist the operators of a Pressurized Water Reactor (PWR) plant to assess the primary system conditions during operational and anticipated transients. This model may be readily adopted by on-line safety systems such as the Safety Parameter Display System (SPDS) and the Disturbance Analysis Surveilance System (DASS).

In this model, the PWR primary system is divided into the core, reactor vessel upper and lower plenum, hot leg, cold leg, steam generator, and pressurizer components. Within each flow component, a linear enthalpy profile is assumed. In the plenum components, perfect enthalpy mixing is assumed. The drift flux model is used to treat two-phase flow. Furthermore, a form of donor cell differencing scheme is used, hence, flow reversal is permitted.

A two-fluid pressurizer model is developed to treat the pressurizer dynamics. Interfacial mass transfer mechanisms such as rainout, flashing, and wall condensation are included. The surge mass flow rate is determined by an implicit coupling technique between the state equations of the flow path components and the pressurizer.

The momentum equation is decoupled from the energy and mass equations by a semi-implicit differencing method. A multiple-loop integral momentum model is adopted with assumption that flow is uniform in each loop.

A computer code, named SPK (System and Pressurizer Kode) code, is written and tested on the DEC PDP/11 and Honeywell DPS 8/70 M computers, with computational speed of twenty to hundred times faster than real time.

Thesis Supervisor: John E. Meyer
Title: Professor of Nuclear Engineering

Thesis Co-Supervisor: John H. Hopps, Jr.
Title: Research Administrator, CSDL

Thesis Reader: David D. Lanning
Title: Professor of Nuclear Engineering

ACKNOWLEDGEMENTS

I would like to express my greatest appreciation to my thesis supervisor, Professor John E. Meyer, for his guidance, concern, and invaluable expertise. His inspiration contributed a great deal in my learning process. I would also like to thank my thesis co-supervisor, Dr. John H. Hopps, Jr., for his endless support and valuable suggestions. Thanks are also due to Professor David D. Lanning, for his valuable advice and assistant.

I wish to express thanks to my officemates and colleagues who contributed much to my success at MIT:

Walter Strohmayer,

Derek Ebeling-Koning,

Jim Robinson,

Rick Vilim, and

Renato Ornedo.

I am indebted to Karen Sullivan who typed the equations and tables in this thesis in an amazing speed, and to Frank Cross who arranged the typing and printing of this thesis.

This research was supported through the fellowship program at The Charles Stark Draper Laboratory.

Special thanks are due to:

- Linda Tai, for her invaluable friendship and care;
- Fei, for her care and willingness to help;
- Rachel Morton, who helped to turn computing

nightmares into fruitful results;

- Marsha Levine, who helped to locate many technical information; and
- Paul S., Shih-Kuei C., Tolis E., Bo R., Chon-Kwo T., Rick B., Tsing-Tung H., Ming L., Hong-Nian J., Tsang-Lang L., and Alan W. for making life at N.E.D. more enjoyable.

And finally, this thesis is dedicated to my parents, for their never-ending love, support, and encouragement.

Shih-Ping Kaŏ

TABLE OF CONTENTS

page

page

LIST OF FIGURES

FIGURE | page

FIGURE		page

FIGURE page

LIST OF TABLES

NOMENCLATURE

A	Area (m**2)
C	Thermal capacity (J/kg-m**3)
C_o	Void distribution parameter
E	Total internal energy (J)
e	Volume-averaged mixture enthalpy (J/kg-m**3)
g	Gravitational constant (9.81 m/s**2)
H	Specific enthalpy (J/kg)
H'	Mixing-cup specific enthalpy (J/kg)
h	Heat transfer coefficient (W/m**2-C)
k	Thermal conductivity (W/m-C)
L	Pressurizer wall thickness (m)
ℓ	Wall condensate film length (m)
M	Mass (kg)
m	Volume-averaged mixture density (J/kg-m**3)
P	Pressure (Pa)
Q	Power (W)
q"	Heat flux (W/m**2)
T	Temperature (K)
t	Time (s)
U	Over-all heat transfer coefficient (W/m**2-C)
U_{gj}	Drift velocity (m/s)
V	Volume (m**3)
v	Specific volume (m**3/kg)
W	Mass flow rate (kg/s)

Greek Letters

α	Void fraction
Δ	Discretization
ρ	Density (kg/m**3)
τ	Time constant (s)
μ	viscosity (N-s/m**2)
ω	pump speed (rad/s)

Symbols

ℓn	Natural logrithm
< >	Average

Subscripts

C	Core
CL	Cold-leg
f	Saturated liquid
f	Flashing
g	Saturated vapor
h	Pressurizer heaters
HL	Hot-leg
i	Mesh index
l	Liquid
LP	Reactor vessel lower plenum
m	Mixture
P	Pressurizer
ro	Rain-out
rv	Relief-valve

S	Secondary-side
s	Source
sat	Saturation
sc	Spray-condensate
SG	Steam-generator
si	Safety-injection
sp	Spray
su	Surge
sv	Safety-valve
TM	Steam-generator tube-metal
UP	Reactor vessel upper plenum
v	Vapor
w	Pressurizer vessel wall
wc	Wall condensation

Superscript

n	Present time index
n+1	Advanced time index
T	Transpose (of a vector or a matrix)

CHAPTER 1

INTRODUCTION

This thesis addresses the issue of plant sensor validation for Pressurized Water Reactor(PWR) power plants. The goal is to develop an operator assist system, thereby to ensure that validated information is always available to the operators. Information regarding to the operating condition of the Reactor Coolant System (RCS) is the main focus of this research.

1.1 Motivation

In the past decade, safe operation of nuclear power plants has been the dominating issue in the nuclear power industry. Unexpected accidents, such as the fire at Browns Ferry, the stuck-open relief valve at TMI-2, and the steam generator tube rupture at Ginna, have drastically weakened the public's and utilities' trust concerning the safety and reliability of the nuclear power plants. These and other less-publicized accidents have proved that operator interventions can either mitigate or aggravate the consequences. The deciding factor is how fast the operators come to realize the current plant status and the appropriate corrective steps. A computer-based operator assist system is very likely to shorten the time for proper operator action during an accident or transient. The Safety Parameter Display system (SPDS) mandated by the Nuclear

Regulatory Commission [U1] and the Disturbance Analysis and Surveillance System (DASS) proposed by the Electric Power Research Institute [E1] are attempts to help the operators to interpret plant sensor information.

The purpose for an operator assist system such as SPDS is to provide a continuous indication of plant parameters representative of the safety status of the plant. The requirement for the system is that all parameters for display must be validated on a real-time basis, and must not be compromised by failed sensors. In order to meet the stated requirement, the Charles Stark Draper Laboratory has proposed adopting a signal validation methodology that utilizes a parity-space representation and analytic redundancy [M2].

A parity-space representation transforms an array of redundant measurements of a quantity into a parity vector. The transformation suppresses the true value of the underlying variable, leaving only components of the measurement errors as elements of the parity vector. Faults can be isolated by the indication of the magnitude and direction of the parity vector.

Analytic redundancy refers to the analytic solutions obtained by solving the physical relationships, such as the conservation of mass and energy, that exist among the variables being measured or interpreted in a system. The solutions are suitably called "analytic measurements". The purpose of analytic measurement is to provide redundancy in addition to that provided by physical sensors and to increase the reliability of sensor validation. In case of failure of all available sensors, or loss of sensor information due to inactive components, analytic measurement becomes the only source of information.

The basic methodology of the signal validation technique involves the use of both analytic and direct sensor measurements as inputs to the decision/estimators (D/E). The decision/estimator consists of two processes. First, decisions are made concerning the inconsistency of any of its input measurements using the parity-space technique. Secondly, calculations provide an estimated value of the measured variable from its consistent measurements. A typical decision/estimator is illustrated in Figure 1.1. The D/E

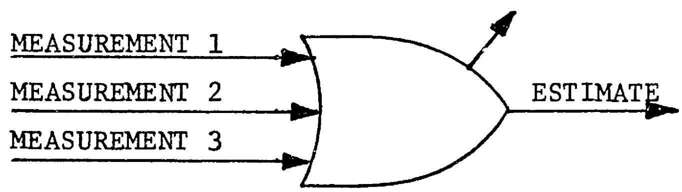


Figure 1.1 A Typical Decision Estimator [M2].

shown in the figure has three input measurements. Although not indicated in the figure, the measurements consist of two direct sensor measurements and one analytic measurement. The D/E is represented by a logical "or" gate. The oblique arrow indicates a detection of any inconsistency or failure among the input measurements. The output of the D/E is the signal representing the estimated value of the variable being measured. The output can be displayed to the operator in graphic form showing the value and trend of the measured variable.

The technique of parity-space representation and analytic redundancy has been used in the aerospace applications, such as the space shuttle, with proven reliability and success. It is felt that a simular approach can be applied to nuclear power plants to increase the reliability of operation of these complex systems. If successful, the benefits will be expressed in terms of minimizing the chance and consequence of accidents caused by mechanical failures or human errors, thus minimizing the costs to, and psychological stress on the public.

1.2 Objective

The objective of this research is to develop computer models for analytic redundancy applied to an operator assist system for a PWR Reactor Coolant System. The requirements for these models are the following:

1) they must operate in much less than real time;
2) they must simulate the behavior of actual plant;
3) they must operate in operational transients under normal or faulted conditions; and

4) they must be developed in a way that they can be tuned or adapted to actual plant information once installed.

Upon meeting these requirements, validated and uninterrupted information flow of key RCS safety parameters will be provided to the operators, thereby assist them in performing and responding to normal and abnormal operations.

1.3 Application

The range of application includes both normal operations such as power level maneuvering, reactor shutdown (or scram), pump coastdown, and natural circulation, and as well as faulted operations such as stuck-open relief valve and other small break loss-of-coolant (LOCA) types of failure. Under faulted operations, a two-phase flow condition could develop in the RCS. Therefore, some two-phase flow situations will be included in the analysis.

However, the calculations are not intended to apply to severe accident conditions, such as large break LOCA and degraded core, in which stratified and/or countercurrent flows may exist.

The models provide analytical information on key safety parameters regarding the condition of the RCS, such as subcooling margin, mass inventory, and heat storage rate.

1.4 Previous Work

There are many faster-than-real-time digital simulation models for Pressurized Water Reactors in the open literature. They differ from those used in the system analysis codes such as RETRAN [M7] and RELAP [R1]. The major difference is that they typically use ten or fewer control volumes to represent the entire RCS, that they are based on a single-phase formulation, and that they use a single-loop representation. A summary of some of these codes are listed in Table 1.1.

The models that use linearized system equations are suitable for analyzing small power or load changes. It has been demonstrated that such a model can be applied to computer control systems that regulate the control rod positions and the steam valve opening [K2] [N2] [C6].

The non-linear models have wider range of applications including reactor scram and loss of load transients. However, they are no longer appropriate after two-phase conditions occur in the RCS. Furthermore, these models cannot simulate asymmetric operations of the RCS. Such models have been applied theoretically to the digital control of reactor power [C1] [M8] [T1], and real-time simulation of PWR operational transients [H1] and anticipated transients without scram (ATWS) [R5].

Although system codes such as RETRAN and RELAP have much wider application than the simulation codes of Table 1.1, their use on a plant computer on real-time bases is not feasible. The model developed in this thesis bridges the

Table 1.1 Summary of some PWR simulation models.

Author	No. of Control Volume		System Model	Pressurizer Model	Applications
	Primary	Secondary			
Nakayama [N2]	10	1	Linearized	Two-volume	Operational transients
Chenini [C1]	10	9	Linearized	Two-volume	Load changes
Kerlin [K2]	10	1	Linearized	One-volume mixture	Load changes
Moore [M8]	1	1	Nonlinear	None	Load changes
Cooper [C6]	6	3	Nonlinear	One-volume mixture	Load changes
Tyler [T1]	6	4	Nonlinear	One-volume mixture	Operational transients
Robinson [R5]	6	1	Nonlinear	Two-volume	Anticipated transients
Hetrick [H1]	6	2	Nonlinear	Two-volume	Operational transients

gap between the real-time simulation codes and the major system codes with respect to computer applications.

Although modeling of the secondary side of the PWR is not included, the primary system model developed in this thesis can be readily coupled with a U-tube steam generator model such as that developed by Strohmayer [S1]. Strohmayer's steam generator model is a three-volume non-linear model for analyzing operational transients on real-time bases.

1.5 Contribution

In this thesis, an advanced on-line simulation model for the PWR Reactor Coolant System has been developed. The characteristics of this model, that are not found in the previous models, are enumerated as below:

1) it is based on a two-phase formulation;
2) a drift-flux model is used to describe the degree of phase separation;
3) multiple-loop configurations up to four loops are provided;
4) a coarse lumped parameter method is used where linear enthalpy profile is assumed in the flow path components other than the lower and upper plena in the reactor vessel;
5) a donor-cell differencing scheme is applied;
6) complete pump characteristic curves are

included;

7) a two-fluid pressurizer model is developed and implicitly coupled with the flow path component models;

8) condensation on the pressurizer wall is determined by an one-lump conduction model;

9) the equations for time derivatives of state variables are solved implicitly using a Gauss reduction method;

10) hydrostatic head is determined by solving for the thermal center in the steam generator;

11) a state space representation method is used so that it is suitable for digital control applications; and

12) provisions are made to simulate small LOCA transients.

In this model, the non-linearities of the state equations are retained. Therefore, it can simulate transients away from the normal operational conditions. The model has included many physical formulations similar in detail as those of major system codes, yet it is able to execute in faster-than-real time.

The model is a useful tool for off-line simulation as well as on-line signal validation and parameter estimation. Its application to signal validation is discussed in this thesis.

1.6 Organization

The structure of the thesis following this chapter is discussed below.

Chapter 2 contains the formulation of the state-space model consisting of the conservation equations of energy and mass for the RCS flow path components. Also included in this chapter are the core and steam generator heat transfer models.

Chapter 3 discusses the derivation of the state-space representation of the two-fluid pressurizer model. Independent validation results of this model are also presented.

Chapter 4 discusses the formulation of the momentum and the pump-rotor speed equations. In addition, the derivation of the loop hydrostatic head is included. Also presented are the results of validation tests of the momentum model.

Chapter 5 discusses the initial-state solutions and the computational procedure of the transient calculations.

Chapter 6 presents the validation results with respect to single- and two-phase natural circulation. Comparisons are made against plant test data.

Chapter 7 discusses the compatability of the models formulated in this thesis with an operator assist system.

Finally, Chapter 8 presents the summary, conclusions, and recommendations of this thesis.

There are a total of ten appendices following the main text. Appendix A gives a derivation of the two-phase conservation equations. Appendix B gives a list of empirical correlations used. Appendix C gives a summary on the derivation of the single-node fuel rod model. Appendix D derives the steam generator heat transfer rate solutions. Appendix E gives the derivation and application of the wall conduction parameters used in the pressurizer model. Appendix F illustrates the pump characteristic curves incorperated. Appendix G derives the solutions for the hydrostatic head of a U-tube steam generator. Appendix H includes the input files used in the natural circulation test runs. Appendix I discusses the input description of the SPK (System and Pressurizer Kode) code. And finally, Appendix J gives the FORTRAN listing of the SPK code.

CHAPTER 2

RCS FLOW PATH COMPONENT THERMAL MODELS

2.1 Introduction

A single-loop representation of a PWR Reactor Coolant System flow diagram is shown in Figure 2.1. The PWR RCS loop consists of a reactor vessel, steam generator, pump, pressurizer, hot leg, and cold leg (pump suction leg plus pump discharge leg).

The reactor vessel (Figure 2.2) contains the core, core support structures, control rod assemblies, and thermal shield. Coolant enters the vessel through an inlet nozzle connected to the cold leg, flows downward in a space between the thermal shield and the cylindrical vessel wall (or the downcomer), makes a 180° turn in the core lower plenum and throught the support structures, flows upward through the fuel assemblies, enters the core upper plenum, and finally, leaves the vessel through an exit nozzle to the hot leg (Figure 2.3).

The steam generator is the tallest component in the RCS, with an overall height of about 20 meters. Figure 2.4 shows a diagram of a U-tube steam generator. The steam generator consists of an evaporator section and a steam drum section. The evaporator section houses the tube bundle. The steam drum section contains the moisture separators and dryers.

On the primary side, coolant enters the inlet plenum,

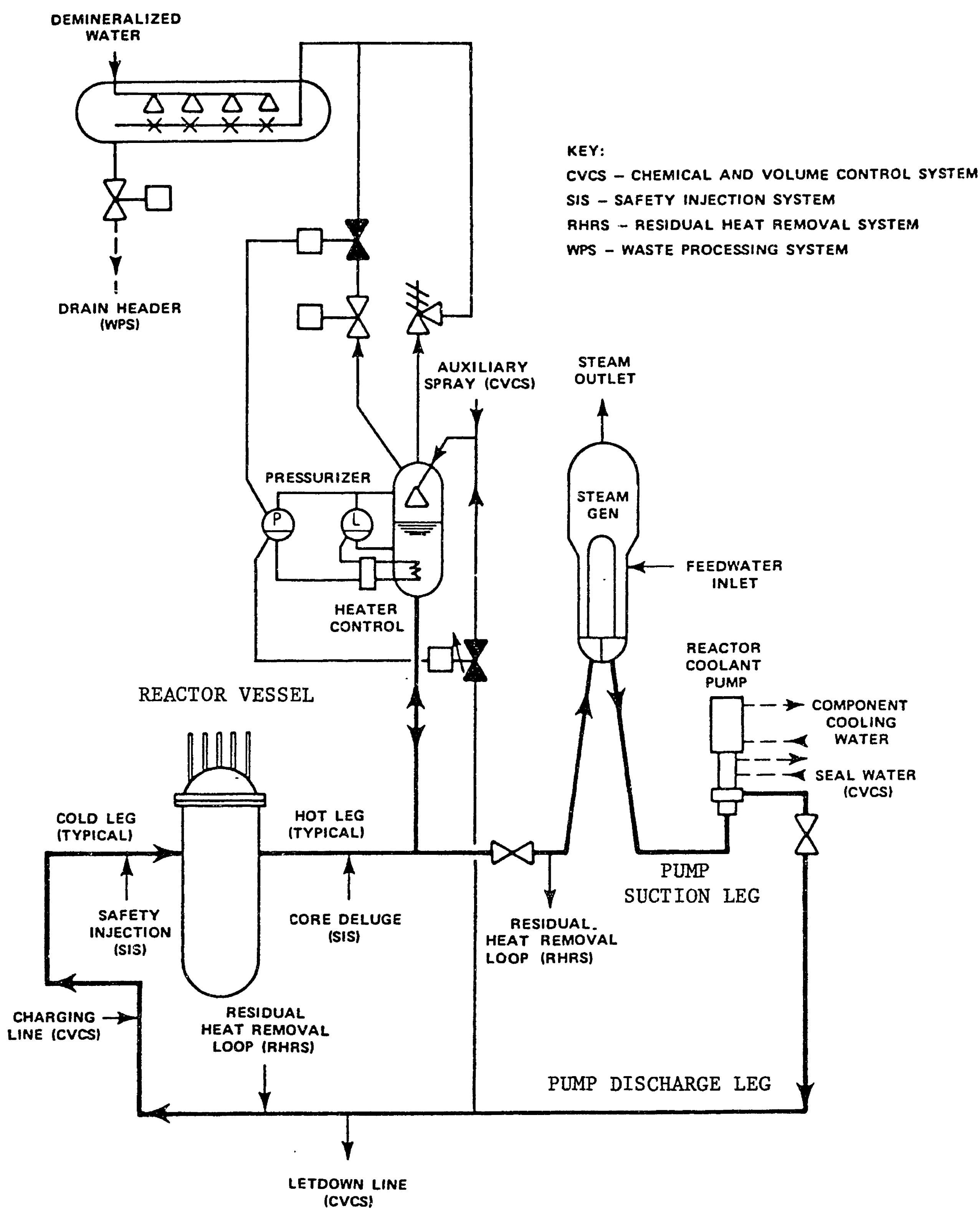


FIGURE 2.1 Reacotr Coolant System [M1].

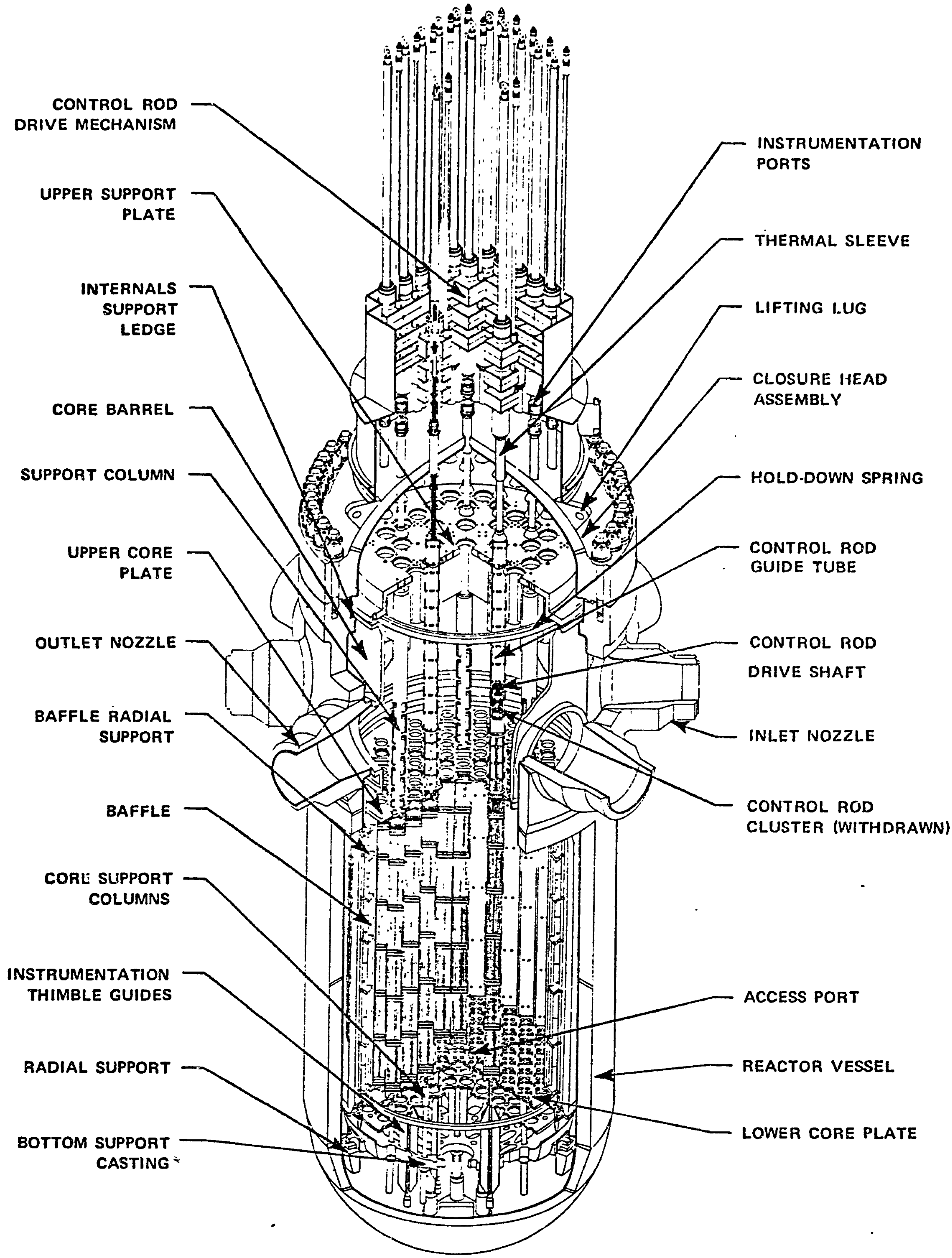


FIGURE 2.2 Reactor Vessel Internals (from Ref. [M1]).

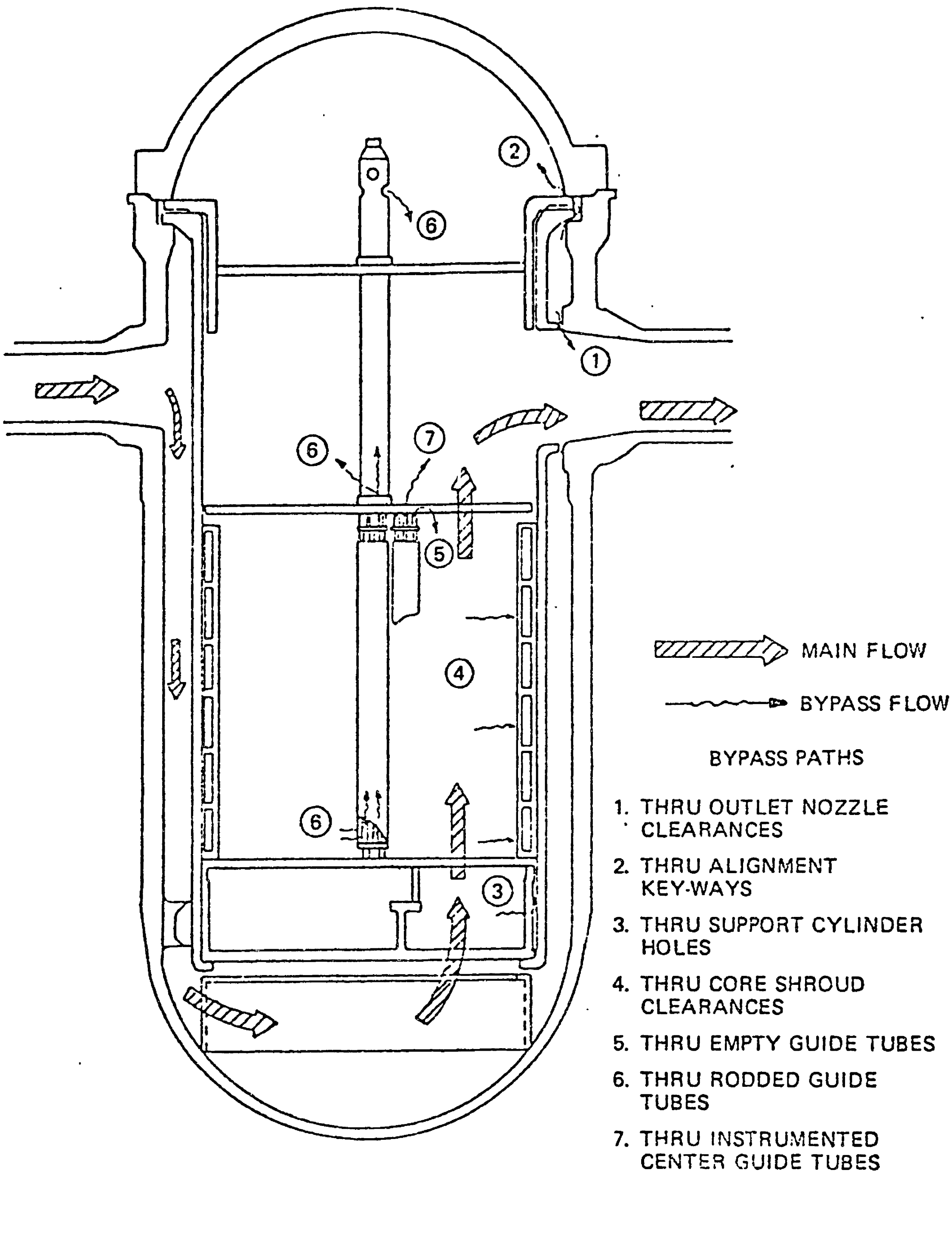


FIGURE 2.3 Reactor Vessel Flow Paths [W4].

No.	Service	No. Req
1	Primary Inlet	1
2	Primary Outlet	2
3	Feedwater Inlet	1
4	Steam Outlet	1
5	Bottom Blowdown	1
6	Pressure Tap	4
7	Liquid Level	8
8	Primary Manway	2
9	Secondary Manway	2
10	Handhole	2

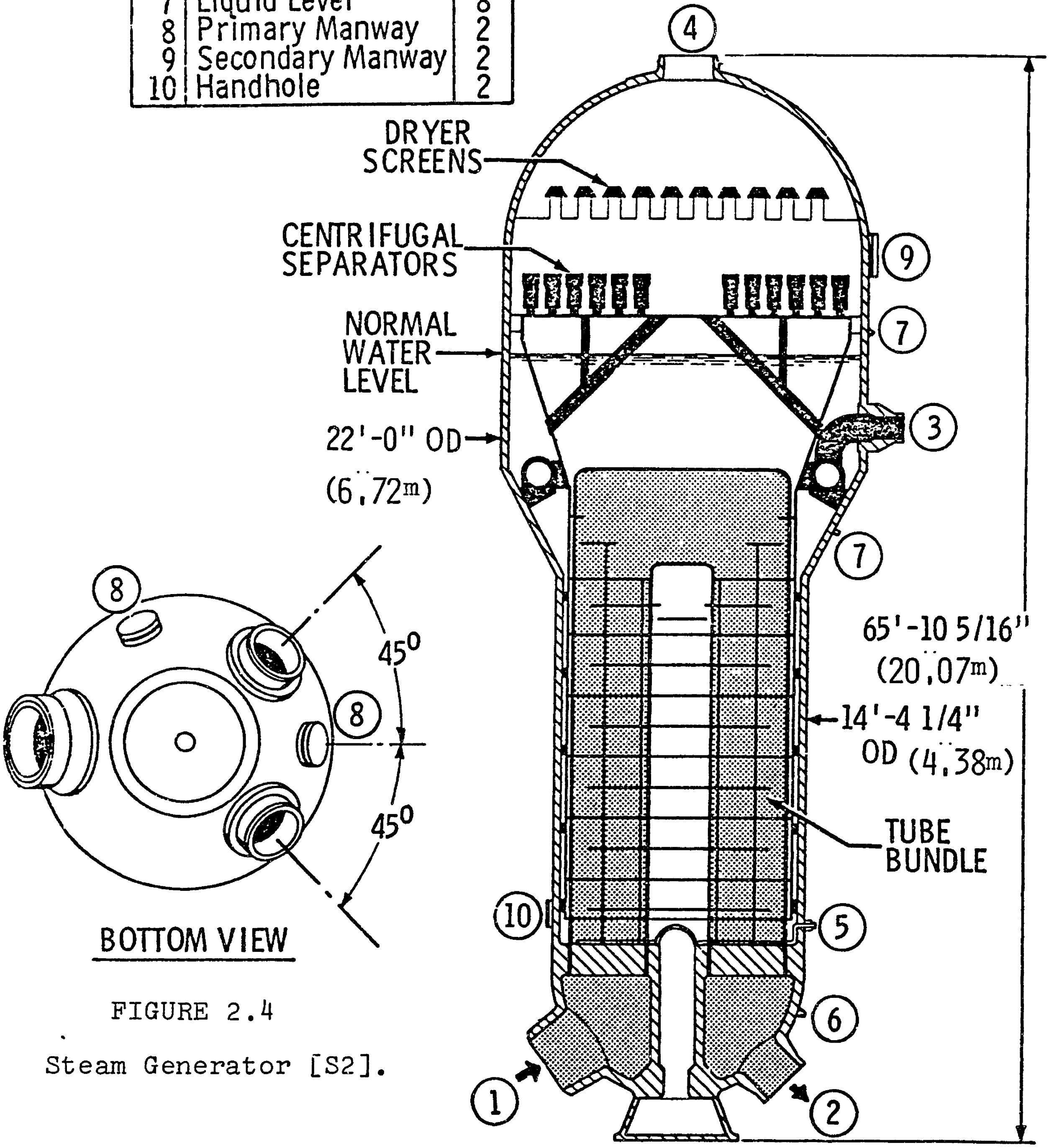


FIGURE 2.4

Steam Generator [S2].

flows inside some five to nine thousand U-shaped tubes, then exits through the outlet plenum. On the secondary side, feedwater enters the steam generator via feedwater nozzle and is distributed in the feedwater mixing region by the feedwater ring. There the feedwater mixes with the saturated liquid returning from the moisture separation devices. The well-mixed fluid flows downward through the annular downcomer region between the tube bundle barrel and the vessel wall. The fluid turns around at the bottom of the downcomer, and flows upward through the tube bundle. The fluid leaves the tube bundle region as two-phase mixture and flows upward through the riser into the steam drum section. After moisture separation and drying, the steam leaves the steam generator through an outlet nozzle, while the saturated water is returned to the feedwater mixing region. [S1] The recirculation flow is driven by the density difference between the fluid in the tube bundle and riser regions and the fluid in the feedwater mixing and downcomer regions.

In this research, only the primary side of the U-tube steam generator is modeled. The primary side is divided into three regions: inlet plenum, U-tubes, and outlet plenum. Pressure on the secondary side is given as a boundary condition for the heat removal calculations.

The piping in the hot and cold legs are the simplest components in the RCS. The piping models are simpler versions of the other RCS flow path components.

Although the pressurizer is a component in the RCS, it does not lie in the recirculation flow path. The pressurizer is modeled as a separate system. The modeling technique and the method of coupling the pressurizer to the RCS flow path components is discussed in Chapter 3.

Each Reactor Coolant Pump is a complex system by itself. The pump model is an integral part of the momentum model. The pump and momentum models are discussed in Chapter 4.

The modeling strategy is to reduce the number of state variables (or to use only a few control volumes while retaining essential physical features).

Figure 2.5 shows a control volume noding scheme for an one-loop Reactor Coolant System. The RCS is divided into the following components: reactor core, reactor vessel upper plenum, hot leg, steam generator tube volume, cold leg, reactor vessel lower plenum, and pressurizer. The reactor vessel upper plenum consists of the core upper plenum and vessel underhead. The hot leg consists of the hot leg and steam generator inlet plenum. The cold leg consists of the steam generator exit plenum, pump suction leg, pump impeller housing, and pump discharge leg. The reactor vessel lower plenum consists of the downcomer and lower core plenum and support structures.

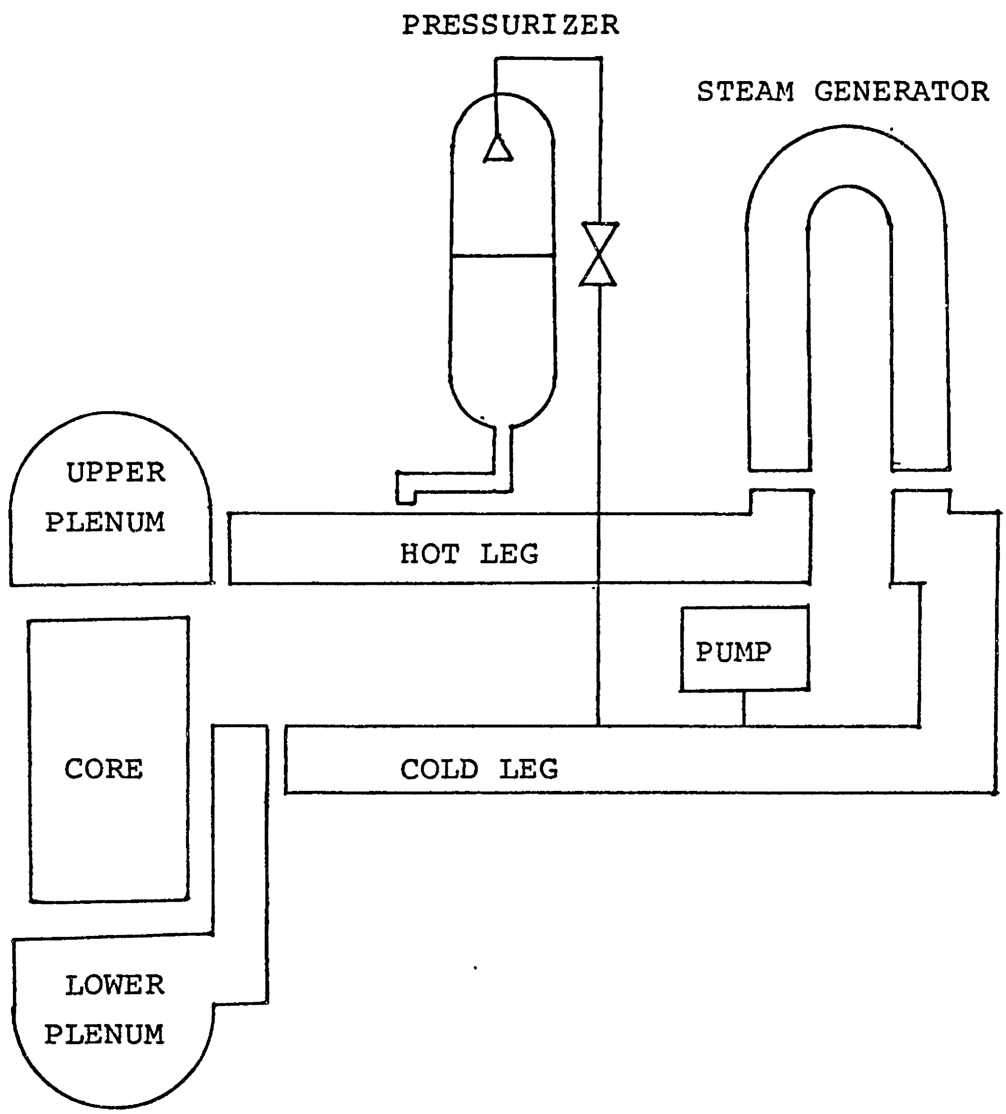


FIGURE 2.5

Reactor Coolant System Control Volume

Noding Scheme

2.2 Energy and Mass Conservation Equations

A typical RCS loop control volume is shown in Figure 2.6. An aligned-mesh scheme and a donor cell method are used in this model. The exit mixing-cup enthalpy, H'_i, is defined in terms of the total mass and energy contents inside the control volume Cell i, M_i and E_i. The external boundary conditions are the conduction heat transfer across the boundary, Q_i, and an external fluid source with mass flow rate, W_s, and enthalpy, H'_s. The state variables for the control volume are chosen to be the pressure, P, and the mixture enthalpy, H_m. The reason for this choice is that the total mass and energy contents are functions of H_m and P only, and that H_m is continuous as the coolant changes from single to two phase within the cell.

The mass and energy conservation equations of the control volume i in Figure 2.6 are, respectively:

$$\frac{dM_i}{dt} = W_{i-1} - W_i + W_s \qquad (2.1)$$

and

$$\frac{dE_i}{dt} = (WH')_{i-1} - (WH')_i + (WH')_s + Q_i \qquad (2.2)$$

where M_i and E_i are defined by:

$$M_i \equiv \int_{V_i} \rho_m(H_m,P)\, dV \qquad (2.3)$$

and

$$E_i \equiv \int_{V_i} \rho_m(H_m,P)\, H_m dV - PV_i \qquad (2.4)$$

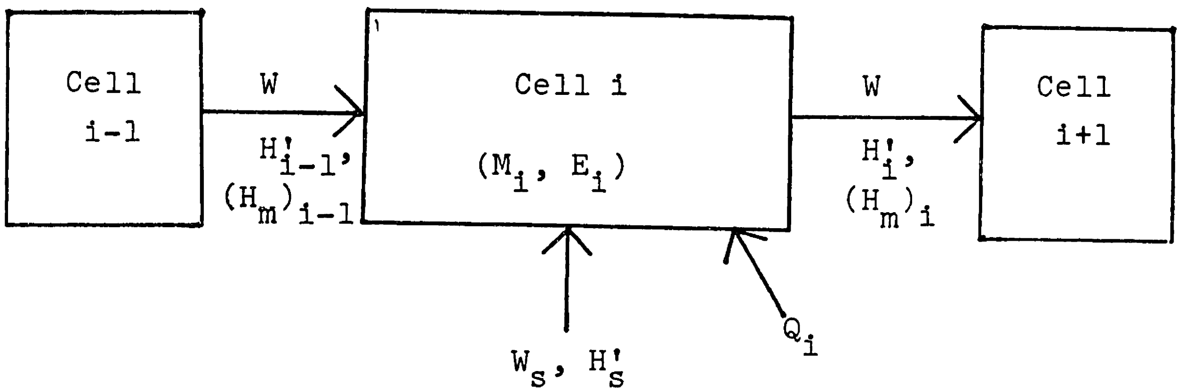


FIGURE 2.6 RCS Flow Path Control Volume

Strohmayer [S1] suggests that the energy conservation equation, Eq. (2.2), can be combined with the mass conservation equation, Eq. (2.1) in a useful manner. The resulting energy equation is independent of enthalpy reference point even if additional approximations are made during the solution procedure. Multiply Eq. (2.1) by the average mixture enthalpy of cell i, then subtract from Eq. (2.2), to obtain:

$$\frac{dE_i}{dt} - \langle H_m \rangle_i \frac{dM_i}{dt} = W_{i-1}(H'_{i-1} - \langle H_m \rangle_i) - W_i(H'_i - \langle H_m \rangle_i) + W_s(H'_s - \langle H_m \rangle_i) + Q_i \tag{2.5}$$

where

$$\langle H_m \rangle_i = 1/2[(H_m)_i + (H_m)_{i-1}] \quad .$$

The advantage of Eq. (2.5) is that if the mass equation is solved approximately, error introduced in the energy equation is reduced by cancellations.

Furthermore, assume that the mass flow rate is approximately uniform within the RCS loop. That is, $W_i = W_{i-1} = W$, which also implies $W_s << W$. Then Eq. (2.5) becomes:

$$\frac{dE_i}{dt} - \langle H_m \rangle_i \frac{dM_i}{dt} = W(H'_{i-1} - H'_i) + W_s(H'_s - \langle H_m \rangle_i) + Q_i \tag{2.6}$$

The single mass flow rate approximation is consistent with the momentum integral model, which is discussed in Chapter 4.

In order to express Eq. (2.6) in terms of state variables, the functional relationships for M_i, E_i, and H_i' in terms of H_m and P must be found. A method suggested by Meyer [M3], which is used in the SSC-L code [A2], defines M_i and E_i in terms of the volume-averaged mixture density, m_i, and the volume-averaged mixture enthalpy, e_i, respectively as:

$$M_i = m_i V_i \tag{2.7}$$

and

$$E_i = e_i V_i - PV_i \tag{2.8}$$

where m_i and e_i are defined by:

$$m_i \equiv \frac{1}{V_i} \int_{V_i} \rho_m(H_m,P)\ dV \tag{2.9}$$

and

$$e_i \equiv \frac{1}{V_i} \int_{V_i} \rho_m(H_m,P)\ H_m dV \tag{2.10}$$

The integrals in Eqs. (2.9) and (2.10) can be evaluated analytically if the profile for H_m and the functional relationship between ρ_m and H_m are known. Assume that the transient is slow enough so that the linear profile always holds. By a linear transformation, m_i and e_i become functions of the inlet and exit mixture enthalpies and the pressure:

$$m_i \equiv \frac{1}{(H_m)_i - (H_m)_{i-1}} \int_{(H_m)_{i-1}}^{(H_m)_i} \rho_m(H_m,P)\ dH_m \tag{2.11}$$

and

$$e_i = \frac{1}{(H_m)_i - (H_m)_{i-1}} \int_{(H_m)_{i-1}}^{(H_m)_i} \rho_m(H_m,P)\, H_m \, dH_m \qquad (2.12)$$

Figure 2.7 shows a typical plot of mixture density versus mixture enthalpy at a constant pressure [P1]. From the plot, a relationship between density and enthalpy can be obtained by assuming that density varies linearly with enthalpy for subcooled liquid, and the specific volume varies linearly with enthalpy a two-phase mixture and superheated vapor [A1]. Therefore, with the above assumptions, the expressions of m_i and e_i under the linear enthalpy profile assumption are obtained and given in Table 2.1.

In single-phase flow and a uniform velocity, H_m and H' are equal to the liquid specific enthalpy, H_ℓ. In two-phase (saturated) flow, a relation between H_m and H' can be obtained via the drift flux model [Z1]. The mixture enthalpy is defined in terms of the static quality, Y, and the pressure:

$$H_m = H_f + Y\, H_{fg} \qquad (2.13)$$

where

$$Y = \frac{\alpha\rho_g}{\alpha\rho_g + (1 - \alpha)\rho_f} \qquad (2.14)$$

The mixing-cup enthalpy is defined in terms of the flow quality, X, and the pressure:

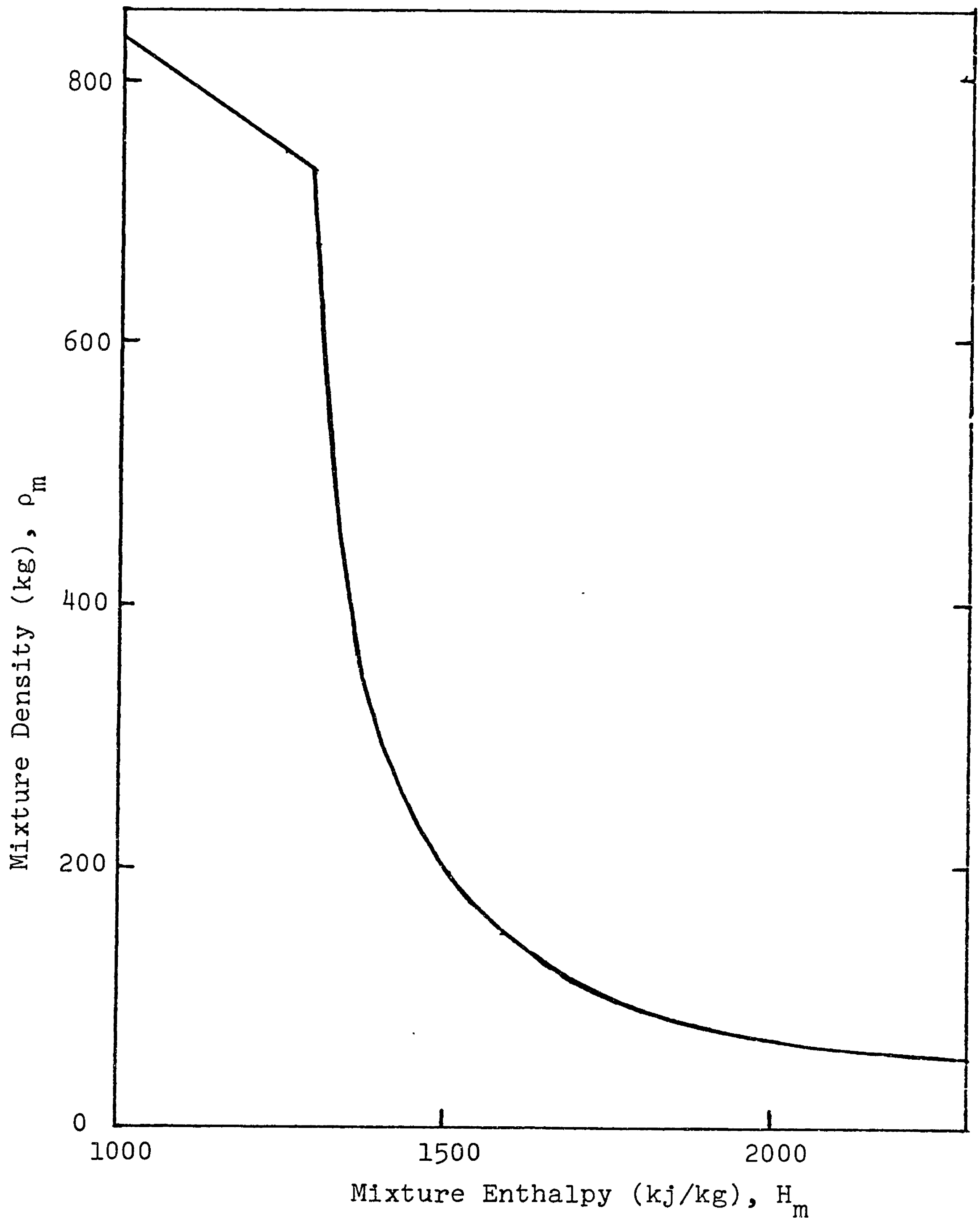


FIGURE 2.7 Mixture Density vs Mixture Enthalpy at 7.5 MPa.

Table 2.1. Expressions of m_i and e_i under linear enthlapy profile assumption.

Condition	m_i	e_i
$(H_m)_{i-1} < (H_m)_i < H_f$	$\frac{1}{2}(\rho_i + \rho_{i-1})_m$	$\frac{1}{6}[(2H_i + H_{i-1})\rho_i + (2H_{i-1} + H_i)\rho_{i-1}]_m$
$(H_m)_{i-1} < H_f < (H_m)_i$	$\frac{1}{2}\beta(\rho_{i-1} + \rho_f)_m + \frac{(1-\beta)\ \ell_n(\frac{v_i}{v_f})}{v_i - v_f}$ where $\beta = (\frac{H_f - H_{i-1}}{H_i - H_{i-1}})_m$, $\frac{1}{V} = \alpha P_g + (1-\alpha)P_f$	$\frac{\beta}{6}[(2H_f + H_{i-1})\rho_f + (2H_{i-1} + H_f)\rho_{i-1})]_m + (1-\beta)\left[\frac{H_i - H_f}{v_i - v_f} + \frac{(H_f v_i - H_i v_f)\ \ell_n(\frac{v_i}{v_f})}{(v_i - v_f)^2}\right]_m$
$H_f < (H_m)_{i-1} < (H_m)_i$	$\frac{\ell_n(\frac{v_i}{v_f})}{v_i - v_{i-1}}$ where $v = 1/(\rho_m)$	$\left[\frac{H_i - H_{i-1}}{v_i - v_{1-1}}\right]_m + \left[\frac{(H_{i-1}v_i - H_i v_{i-1})}{(v_i - v_{i-1})^2}\right] \ell_n(\frac{v_i}{v_{i-1}})$

$$H' = H_f + X\,H_{fg} \tag{2.15}$$

The drift flux model gives the following relation between and X:

$$\alpha = X/\{C_o[x + \frac{\rho_g}{\rho_f}(1 - X)] + \frac{\rho_g U_{gj} A}{W} \tag{2.16}$$

where

C_o = void distribution parameter

U_{gj} = drift velocity.

Correlations for C_o and U_{gj} are given in Appendix B.

Combine Eqs. (2.13) to (2.16), obtain a functional relationship for H' in terms of H_m, P, W, and drift flux parameters, C_o and U_{gj}:

$$H' = H'(H_m, W, P, C_o, U_{gj})$$

or

$$H' = H_f + \frac{[C_o + (\frac{\rho_f}{\rho_g})(\frac{\rho_g U_{gj} A}{W})](H_m - H_f)}{1 - (1 - C_o)(1 - \frac{\rho_f}{\rho_g})(\frac{H_m - H_f}{H_{fg}})} \tag{2.17}$$

We have shown that M_i and E_i are functions of P, $(H_m)_i$, and $(H_m)_{i-1}$; hence, Eq. (2.6) can be expressed as:

$$(\frac{\partial E'}{\partial H_m})_i \frac{d(H_m)_i}{dt} + (\frac{\partial E_i'}{\partial H_{m,i-1}}) \frac{d(H_m)_{i-1}}{dt} + (\frac{\partial E'}{\partial P})_i \frac{dP}{dt}$$

$$= W(H_{i-1}' - H_i') + W_s(H_s' - \langle H_m\rangle_i) + Q_i \tag{2.18}$$

where

$$\left(\frac{\partial E'}{\partial H_m}\right)_i = V_i \left[\left(\frac{\partial e}{\partial H_m}\right)_i - \langle H_m \rangle_i \left(\frac{\partial M}{\partial H_m}\right)_i\right]$$

$$\left(\frac{\partial E_i'}{\partial H_{m,i-1}}\right) = V_i \left[\left(\frac{\partial e_i}{\partial H_{m,i-1}}\right) - \langle H_m \rangle_i \left(\frac{\partial M}{\partial H_m}\right)_{i-1}\right]$$

and

$$\left(\frac{\partial E'}{\partial P}\right)_i = V_i \left[\left(\frac{\partial e_i}{\partial p}\right) - \langle H_m \rangle_i \left(\frac{\partial M_i}{\partial p}\right) - 1\right]$$

Similarly, the mass conservation equation, Eq. (2.1), can be expressed as:

$$\left(\frac{\partial M}{\partial H_m}\right)_i \frac{d(H_m)_i}{dt} + \left(\frac{\partial M_i}{\partial H_{m,i-1}}\right) \frac{d(H_m)_{i-1}}{dt} + \left(\frac{\partial M}{\partial p}\right)_i \frac{dP}{dt} = W_s \qquad (2.19)$$

where

$$\left(\frac{\partial M}{\partial H_m}\right)_i = V_i \left(\frac{\partial m}{\partial H_m}\right)_i$$

$$\left(\frac{\partial M_i}{\partial H_{m,i-1}}\right) = V_i \left(\frac{\partial m_i}{\partial H_{m,i-1}}\right)$$

and

$$\left(\frac{\partial M}{\partial P}\right)_i = V_i \left(\frac{\partial m}{\partial P}\right)_i$$

The partial derivatives in Eqs. (2.18) and (2.19) are given in Table 2.2.

Eqs. (2.18) and (2.19), however, have an undesirable numerical behavior. That is, any abrupt changes at the cell

Table 2.2 **Partial** derivatives in Eqs. (2.18) and (2.19).

Partial	Expression
$\left(\frac{\partial m}{\partial H}\right)_i$	$\frac{(\rho_m)_i - m_i}{(H_m)_i + (H_m)_{i-1}}$
$\left(\frac{\partial m_i}{\partial H_{i-1}}\right)$	$\frac{m_i - (\rho_m)_{i-1}}{(H_m)_i + (H_m)_{i-1}}$
$\left(\frac{\partial e}{\partial H}\right)_i$	$\frac{(\rho_m H_m)_i - e_i}{(H_m)_i + (H_m)_{i-1}}$
$\left(\frac{\partial e_i}{\partial H}\right)_{i-1}$	$\frac{e_i - (\rho_m H_m)_{i-1}}{(H_m)_i + (H_m)_{i-1}}$
$\left(\frac{\partial m}{\partial p}\right)$ & $\left(\frac{\partial e_i}{\partial P}\right)$	$\frac{\partial f}{\partial p} = \frac{f(P + \Delta P) - f(P)}{\Delta P}$ where $f = m_i, e_i$

inlet are propagated immediately to the exit. To avoid this "numerical diffusion", a form of donor-cell differencing suggested by Weaver [W1] is adopted. The method is extended here to account for the vapor slip effect using the drift-flux model.

2.2.1 Positive Mass Flow Rate Case

For the case of positive mass flow rate, replace the first two terms of Eq. (2.18) by the following:

$$\left[\left(\frac{\partial E'}{\partial H_m}\right)_i + \left(\frac{\partial E'_i}{\partial H_{m,i-1}}\right)\right] \frac{d(H_{m,i-1})}{dt} = \left(\frac{\partial E'}{\partial H_m}\right)_{i,i-1} \frac{d(H_{m,i})}{dt} \tag{2.20}$$

where

$$\left(\frac{\partial E'}{\partial H_m}\right)_{i,i-1} = V_i \left\{\left[\left(\frac{\partial e}{\partial H_m}\right)_i + \left(\frac{\partial e_i}{\partial H_{m,i-1}}\right)\right.\right.$$

$$\left.\left. - \langle H_m\rangle_i \left[\left(\frac{\partial m}{\partial H_m}\right)_i + \left(\frac{\partial m_i}{\partial H_{m,i-1}}\right)\right]\right]\right\}$$

Hence, Eq. (2.18) becomes:

$$\left(\frac{\partial E'}{\partial H_m}\right)_{i,i-1} \frac{d(H_m)_i}{dt} + \left(\frac{\partial E'}{\partial P}\right)_i \frac{dP}{dt} = W(H'_{i-1} - H'_i)$$

$$+ W_s(H'_s - \langle H_m\rangle_i) + Q_i \tag{2.21}$$

Finite-difference Eq. (2.21) implicitly, obtain:

$$\left(\frac{\partial E'}{\partial H_m}\right)_{i,i-1} \Delta(H_m)_i^{n+1} + \left(\frac{\partial E'}{\partial P}\right)_i \Delta P^{n+1} = \Delta t W^{n+1}(H'_i - H'_{i-1})^{n+1}$$

$$+ \Delta t W_s^{n+1}\left[(H'_s)^{n+1} - \langle H_m\rangle_i^n\right] + \Delta t Q_i \tag{2.22}$$

where

$$\Delta P^{n+1} = P^{n+1} - P^{n};$$

and where the new mixing-cup enthalpy, $(H')^{n+1}$, can be expressed as:

$$(H')^{n+1} = (H')^{n} + \left(\frac{\partial H'}{\partial H_m}\right)^{n} \Delta H_m^{n+1} \tag{2.23}$$

where

$$\frac{\partial H'}{\partial H_m} = \frac{C_o + \left(\frac{\rho_f}{\rho_g}\right)\left(\frac{\rho_g U_{gj} A}{W}\right)}{\left[1 - (1 - C_o)\left(1 - \frac{\rho_f}{\rho_g}\right)\left(\frac{H_m - H_f}{H_{fg}}\right)\right]^2} \tag{2.24}$$

In Eq. (2.23), the pressure dependence is neglected. In single-phase flow, the right hand side of Eq. (2.24) is reduced to one.

Substitute Eq. (2.23) into Eq. (2.22), obtain:

$$\left(\frac{\partial E'}{\partial P}\right)_i^n \Delta P^{n+1} + \left[\left(\frac{\partial E'}{\partial H_m}\right)^n_{i,i-1} + \Delta t W^{n+1} \left(\frac{\partial H'}{\partial H_m}\right)^n_i\right] \Delta (H_m)_i^{n+1}$$

$$- \left[\Delta t W^{n+1} \left(\frac{\partial H_i'}{\partial H_{m,i-1}}\right)^n\right] \Delta (H_m)_{i-1}^{n+1}$$

$$= \Delta t W_s^{n+1} \left[(H_s')^{n+1} - \langle H_m\rangle^n{}_i\right] + \Delta t W^{n+1} (H_{i-1}' - H_i')^n + \Delta t Q_i^{n+1} \tag{2.25}$$

2.2.2 Negative Mass Flow Rate Case

When the flow is reversed, that is, $W < 0$, the derivation of Eq. (2.25) must be modified accordingly.

Referring to Figure 2.6, the fluid now enters Cell i from Cell i+1. Using a donor-cell differencing method, Eq. (2.22) becomes:

$$\left(\frac{\partial E'}{\partial H_m}\right)_{i,i-1} \frac{d(H_m)_{i-1}}{dt} + \left(\frac{\partial E'}{\partial P}\right)_i \frac{dP}{dt} = W(H'_{i-1} - H'_i) + W_s(H'_s - \langle H_m\rangle_i) + Q_i \qquad (2.26)$$

Finite-difference Eq. (2.26) implicitly, obtain:

$$\left(\frac{\partial E'}{\partial P}\right)_i \Delta P^{n+1} + \left\{\left(\frac{\partial E'}{\partial H_m}\right)^n_{i,i-1} - \Delta t W^{n+1} \left(\frac{\partial H'}{\partial H_m}\right)^n_{i-1}\right\} \Delta(H_m)^{n+1}_{i-1} + \left[\Delta t W^{n+1} \left(\frac{\partial H'}{\partial H_m}\right)^n_i\right] \Delta(H_m)^{n+1}_i = \Delta t W^{n+1}(H'_{i-1} - H'_i)^n + \Delta t W^{n+1}_s\left[(H'_s)^{n+1} - \langle H_m\rangle^n_i\right] \qquad (2.27)$$

2.3 Plenum Models

Unlike other flow path components in the RCS, the reactor core upper and lower plena are flow junctions. Since there are multiple enthalpies at the cell boundaries, the integrands of Eqs. (2.11) and (2.12) are not defined for the plenum cells. Therefore, instead of assuming a linear enthalpy profile, a uniform mixing assumption is applied. The conservation of mass and energy equations for the plenum models are, respectively:

$$\frac{d\langle M\rangle_i}{dt} = \sum_j W_j - \sum_k W_k \qquad (2.28)$$

where j = flow in, k = flow out;

and

$$\frac{d\langle E\rangle_i}{dt} = \sum_j (WH')_j - \sum_k (WH')_k \tag{2.29}$$

where

$$\langle E\rangle_i = \langle M\rangle_i (H_m)_i - P\, V_i \tag{2.30}$$

and

$$\langle M\rangle_i = V_i (\rho_m)_i \tag{2.31}$$

Multiply Eq. (2.28) by $(H_m)_i$, then subtract from Eq. (2.32) to obtain an alternative form of the energy equation for the reasons mentioned in Sec. 2.2:

$$\langle M\rangle_i \frac{d(H_m)_i}{dt} - V_i \frac{dP}{dt} = \sum_j W_j [H_j' - (H_m)_i] - \sum_k W_k [H_k' - (H_m)_i] \tag{2.32}$$

Using the single mass flow rate approximation, the time rate of change of the mass in the plenum is:

$$\frac{d\langle M\rangle_i}{dt} = V_i \left(\frac{\partial \rho_m}{\partial H_m}\right)_i \frac{d(H_m)_i}{dt} + V_i \left(\frac{\partial \rho_m}{\partial P}\right)_i \frac{dP}{dt} \tag{2.33}$$

2.3.1 Reactor Vessel Lower Plenum Model

The block diagram of a reactor vessel lower plenum with two cold legs is shown in Figure 2.8. Coolant from the cold legs mixes thoroughly in the lower plenum, then enters the

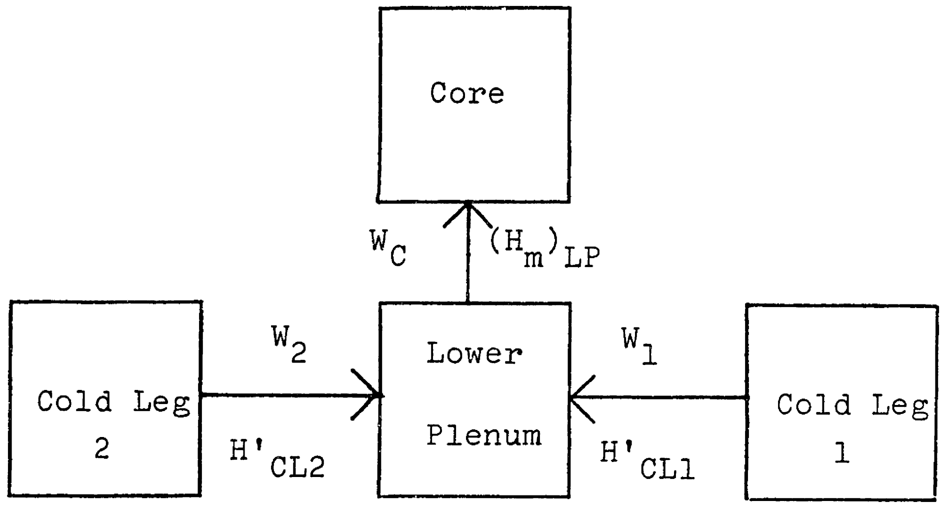


FIGURE 2.8 Reactor Vessel Lower Plenum Flow Diagram.

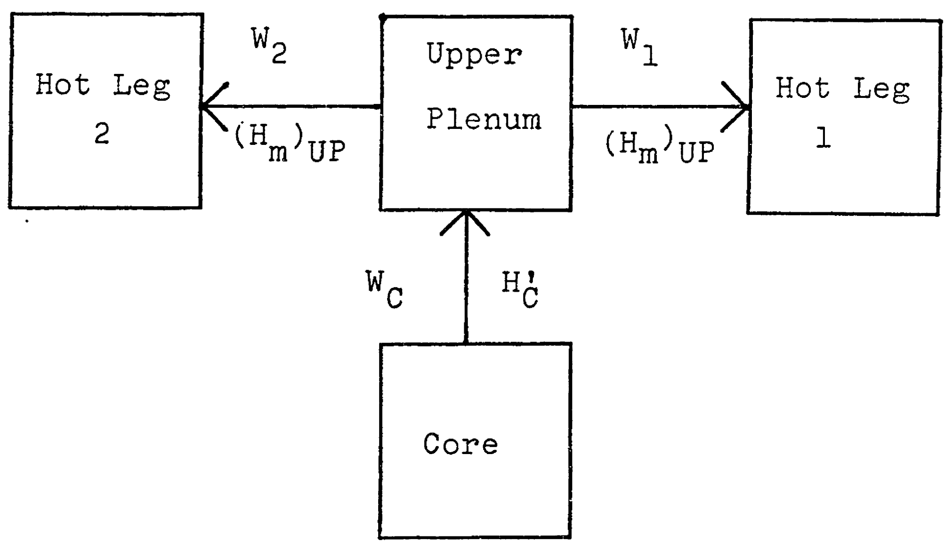


FIGURE 2.9 Reactor Vessel Upper Plenum Flow Diagram.

core region. The energy equation for the lower plenum is:

$$M_{LP} \frac{d(H_m)_{LP}}{dt} - V_{LP} \frac{dP}{dt} = \sum_{i} W(i) \left[(H')_{CL}(i) - (H_m)_{LP}\right] \tag{2.34}$$

If flow reverses in one of the loops, the energy equation is changed to the following:

$$M_{LP} \frac{d(H_m)_{LP}}{dt} - V_{LP} \frac{dP}{dt} = \sum_{i \neq j} W(i) \left[(H')_{CL}(i) - (H_m)_{LP}\right] \tag{2.35}$$

where j = loop with reverse flow.

2.3.2 Reactor Vessel Upper Plenum Model

Figure 2.9 shows the block diagram of a reactor vessel upper plenum with two hot legs. The energy equation for the upper plenum is:

$$M_{UP} \frac{d(H_m)_{UP}}{dt} - V_{UP} \frac{dP}{dt} = W_C \left[H_C' - (H_m)_{UP}\right] \tag{2.36}$$

If flow reverses in one of the loops, Eq. (2.36) becomes:

$$M_{UP} \frac{d(H_m)_{UP}}{dt} - V_{UP} \frac{dP}{dt} = W_C \left[H_C' - (H_m)_{UP}\right] - \sum_{j} W(j) \left[(H')_{HL}(j) - (H_m)_{UP}\right] \tag{2.37}$$

2.4 Over-all Mass Conservation Equation

Since we have adopted a single mass flow rate assumption, we can no longer get meaningful information from the mass conservation equation for each individual control volume. Instead, the over-all flow path mass conservation equation must be used. Finite-difference Eq. (2.19) implicitly, and sum the equation for all RCS flow path control volumes, obtain:

$$\sum_i \left(\frac{\partial M}{\partial P}\right)_i^n \Delta P^{n+1} + \sum_{\text{flow } i} \left[\left(\frac{\partial M}{\partial H_m}\right)_i^n \Delta(H_m)_i^{n+1} + \left(\frac{\partial M}{\partial H_m}\right)_{i-1} \Delta(H_m)_{i-1}^{n+1}\right]$$
$$+ \sum_{\text{plenum } j} \left(\frac{\partial M}{\partial H_m}\right)_j^n \Delta(H_m)_j^{n+1}$$
$$= \Delta t\left[W_{ch} + W_{si} - W_{\ell d} - W_{su} - \sum_i (W_b)_i\right]^{n+1} \quad (2.38)$$

2.5 Core Heat Transfer Model

In the present model, a lumped core heat transfer model is adopted. The objective is to determine the heat transfer rate from the fuel rods to the bulk coolant. The heat transfer is characterized by a single-node model developed by Meyer [M4]. The approach is to use the steady-state temperature distributions and uniform properties (evaluated at the average temperature) assumptions in each region. A summary of this model is given in Appendix F.

The nuclear power deposited in the fuel is given by:

$$N = (1 - f_D)\, N_{FS} + f_D N_D \qquad (2.39)$$

where

N = normalized nuclear power;

N_{FS} = normalized fission power;

N_D = normalized decay power; and

f_D = fraction of decay power at steady state equilibrium conditions.

The decay power is obtained by summing the powers of seven decay-product groups:

$$N_D = \sum_{i=1}^{7} (\beta_D C_D)_i \qquad (2.40)$$

where

$(\beta_D)_i$ = fraction of decay power for Group i ($\sum_i (\beta_D)_i = 1$); and

$(C_D)_i$ = normalized decay power for Group i;
= 1 at steady state.

The time rate of change of $(C_D)_i$ is given by the following first order differential equation:

$$\frac{d(C_D)_i}{dt} = (\lambda_D)_i \left[N_{FS} - (C_D)_i\right] \qquad (2.41)$$

where $(\lambda_D)_i$ is the decay constant for Group i. The values for β_D's and λ_D's are given in Table 2.3. The decay power group constants are derived from the ENDF/B-IV calculations based on 10^{13} seconds of irradiation time [B6], using the method described by Nakayama [N2]. The seven groups cover

Table 2.3. Decay power group constants.

Group, i	% yield, f_i	Fraction $(\beta_D)_i$	Decay Constant $(\lambda)_i$, s^{-1}
1	0.65	0.097	1.28
2	1.47	0.220	0.152
3	1.58	0.237	1.93×10^{-2}
4	1.25	0.187	1.88×10^{-3}
5	0.88	0.132	1.43×10^{-4}
6	0.48	0.072	1.25×10^{-5}
7	0.37	0.055	2.20×10^{-7}

Notes: 1) $f_i = 100\ (\beta_D)_i\ f_D$

2) $f_D = 0.668$

the time span from 0.1 to 10^6 seconds after scram.

Eqs. (2.40) and (2.41) enable us to determine the decay power with the effect of power history before scram taken into consideration.

In the present model, we have assumed that the fission power N_{FS} can be obtained from the direct measurment by the neutron detectors; therefore, it is regarded as an input.

2.6 Steam Generator Heat Transfer Model

The heat transfer rate in the steam generator, Q_{SG}, is claculated using a zero-dimensional analysis, defined in terms of the log-mean temperature difference and the overall heat transfer coefficient [W2] [S1]. That is:

$$Q_{SG} = U_o A_o \Delta T_{\ell m} \tag{2.42}$$

where

$\Delta T_{\ell m}$ = log-mean temperature difference;

A_o = total outside surface area of tubes; and

U_o = overall heat transfer coefficient based on outside surface area of tubes.

The overall heat transfer coefficient is given by:

$$1/U_o = \frac{A_o}{h_i A_i} + \frac{r_o \ \ell n(r_o/r_i)}{k_t} + \frac{1}{h_o} + r_{ff} \tag{2.43}$$

where

h_i = heat transfer coefficient on the tube side,

h_o = heat transfer coefficient on the shell side, and

r_{ff} = fouling factor.

In case of positive flow and heat transfer into the secondary side, the log-mean temperature difference is calculated by:

$$\Delta T_{\ell m} = \frac{T_{in} - T_{out}}{\ell n \left(\frac{T_{in} - T_S}{T_{out} - T_S}\right)} \quad (2.44)$$

where

T_{in} = tube inlet temperature;

T_{out} = tube outlet temperature; and

T_S = the secondary-side temperature, which is assumed equal to the saturated temperature.

Eq. (2.42) is a steady-state heat transfer equation, and Eq. (2.44) is based on an exponential temperature profile along the tube length. Application of these two equations in transient calculations is supported by arguing that the transport time for the primary fluid on the tube-side is short relative to the time span of transients of interest so that the steady-state temperature profile is maintained, and that the time constant for the heat transfer process is very short compared to the fluid transport time so that heat transfer can be assumed to occur instantaneously [S1].

In the case when the tube outlet temperature is below the secondary saturation temperature, Eq. (2.44) becomes undefined. This can occur when the primary flow rate is low and the secondary pressure is increasing rapidly. Strohmayer [S1] has developed a method to overcome this difficulty by replacing the log-mean temperture difference with the following average temperature difference:

$$\Delta\langle T\rangle_{SG} = \eta(T_{in} - T_S) + (1 - \eta)(T_{out} - T_S) \tag{2.45}$$

where the weighting factor, η , is defined by:

$$\eta(T_{in} - T_S) + (1 - \eta)\varepsilon = \Delta T^*_{\ell m} \tag{2.46}$$

where the modified log-mean temperature difference, $\Delta T^*_{\ell m}$, is given by:

$$\Delta T^*_{\ell m} = \frac{T_{in} - T_{out} - \varepsilon}{\ell n \left(\frac{T_{in} - T_S}{\varepsilon}\right)} \tag{2.47}$$

and ε is a small preselected number, i. e.:

$$0 < \varepsilon < (T_{in} - T_{out}).$$

Eq. (2.42) is solved implicitly by adopting a method developed by Strohmayer [S1], given in Appendix C.

2.7 Critical Flow Model

Two critical flow models are incorporated to calculate the mass flow rate at the break.

For incompressible liquid flow, the following equation based on the Second Law of Thermodynamics and the Gibbs equation is used [L1]:

$$W_b = 0.61\ A_b[2\rho_\ell(P_p - P_b)]^{1/2} \qquad (2.48)$$

where

A_b = break area;

ρ_ℓ = up-stream liquid density;

P_p = system pressure, and

P_b = break ambient pressure.

For two-phase critical flow, a critical condition can be obtained from a two-phase momentum conservation equation [L1] as:

$$W_b = A_b\ \left[\frac{dP}{d(1/\rho')}\right]^{1/2} \qquad (2.49)$$

where

$$1/\rho' = v_f + X\ v_{fg} \qquad (2.50)$$

assuming the flow is homogeneous and the phases in thermodynamic equilibrium. To evaluate Eq. (2.49), first expand the derivative as:

$$\frac{d(1/\rho')}{dt} = \left.\frac{\partial(1/\rho')}{\partial P}\right|_X + \left.\frac{\partial(1/\rho')}{\partial X}\right|_P \frac{dX}{dP} \qquad (2.51)$$

Using the relation given by Eq. (2.50), the partial derivatives are:

$$\frac{\partial(1/\rho')}{dP} = \frac{dv_f}{dP} + X\ \frac{dv_{fg}}{dP} \qquad (2.52)$$

and

$$\frac{\partial(1/\rho')}{\partial X} = v_{fg} \tag{2.53}$$

For isentropic process,

$$ds = 0 = d(H') - dP/\rho' \tag{2.54}$$

where

$$H' = H_f + X\, H_{fg} \tag{2.55}$$

Hence,

$$dH' = \left(\frac{\partial H'}{\partial P}\right) dP + \left(\frac{\partial H'}{\partial X}\right) dX \tag{2.56}$$

where

$$\left(\frac{\partial H'}{\partial P}\right) = \frac{dH_f}{dP} + X\,\frac{dH_{fg}}{dP} \tag{2.57}$$

and

$$\left(\frac{\partial H'}{\partial X}\right) = H_{fg} \tag{2.58}$$

Combining Eqs. (2.54) through (2.58), obtain:

$$\frac{dx}{dP} = -\frac{\left[(1-x)\,\frac{dH_f}{dp} + X\,\frac{dH_g}{dp} - 1/\rho'\right]}{H_{fg}} \tag{2.59}$$

Substitute Eqs. (2.52), (2.53), and (2.59) into Eq. (2.49), obtain:

$$W_b = A_b \{-(\frac{v_{fg}}{H_{fg}}) [(1 - X) \frac{dH_f}{dP} + X \frac{dH_{fg}}{dP} - 1/\rho']$$

$$+ [(1 - X) \frac{dv_f}{dp} + X \frac{dv_f}{dP}] \qquad (2.60)$$

2.8 State Space Model

The state equations of the RCS flow path components consist of energy equations for each component and the overall mass conservation equation. The state variables are the mixture enthalpies in each component and the system pressure. For a single-loop representation, there are a total of seven state equations. For multiple-loop representations, the number of state equations is equal to 3+4m, where m is the number of loops. Since implicit finite-difference method is used, ordering of the state equations is changed if reverse flow occurs in one or more loops. Therefore, the state space model must be reconstructed if flow reverse occurs.

2.8.1 Positive Flow Case

The state space model for the positive flow case is represented by the following vector difference equation:

$$\underline{\underline{E}}^n \, \Delta\underline{x}^{n+1} = \underline{\underline{F}}^n \, \underline{u}^{n+1} + \underline{r}^n \qquad (2.61)$$

The order of state equations in Eq. (2.61) is: over-all flow path mass, reactor-vessel lower plenum energy, core energy, reactor-vessel upper plenum energy, hot leg energy, steam generator energy, and cold leg energy equations. Where $\underline{x}$ is the state vector:

$$\underline{x} = [p, (H_m)_{LP}, (H_m)_C, (H_m)_{UP}, (H_m)_{HL}(i), (H_m)_{SG}(i), (H_m)_{CL}(i)]^T$$

$\underline{u}$ is the boundary-condition vector:

$$\underline{u} = [W_{su}, W_{si}, W_{\ell d}, W_{ch}, W_b(i), Q_{th}, Q_{SG}(i)]^T$$

and $\underline{r}$ is the residual vector:

$$\underline{r} = (0, r_2, r_3, r_4, r_5, r_6(i), r_7(i)]^T$$

where the elements of r are given in Table 2.4. Matrices $\underline{\underline{E}}$ and $\underline{\underline{F}}$ have the following structures:

$\underline{\underline{E}}$ =

e_{11}	e_{12}	e_{13}	e_{14}	$e_{15}(i)$	$e_{16}(i)$	$e_{17}(i)$
e_{21}	e_{22}	0	0	0	0	0
e_{31}	e_{32}	e_{33}	0	0	0	0
e_{41}	0	e_{42}	e_{43}	0	0	0
$e_{51}(i)$	0	0	$e_{52}(i)$	$e_{53}(i)$	0	0
$e_{61}(i)$	0	0	0	$e_{62}(i)$	$e_{63}(i)$	0
$e_{71}(i)$	0	0	0	0	$e_{72}(i)$	$e_{73}(i)$

Table 2.4. Elements of vector $\underline{r}$.

Element	Expression
r_2	$\Delta t \sum_i W^{n+1}(i) \; [(H')^n_{CL}(i) - (H_m)^n_{LP}]$
r_3	$\Delta t \; W^{n+1}_C \; [(H_m)^n_{LP} - (H')^n_C]$
r_4	$\Delta t \; W^{n+1}_C \; [(H')^n_C - (H_m)^n_{UP}]$
$r_5(i)$	$\Delta t \; W^{n+1}(i) \; [(H_m)^n_{UP} - (H')^n_{HL}(i)]$
$r_6(i)$	$\Delta t \; W^{n+1}(i) \; [(H')^n_{HL}(i) - (H')^n_{SG}(i)]$
$r_7(i)$	$\Delta t \{W^{n+1}(i) \; [(H')^n_{SG}(i) - (H')^n_{CL}(i)]$ $- W^{n+1}_b(i) \; [H'_{CL} - \langle H_m \rangle_{CL}]^n\}$ $+ \Delta t \; W^{n+1}(i) \; \Delta P^{n+1}_{pump}(i)/\rho_{CL}(i)$

and

$\underline{\underline{F}}$ =

f_{11}	f_{12}	f_{13}	f_{14}	$f_{15}(i)$	$f_{16}(i)$	$f_{17}(i)$	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	f_{31}	0
0	0	0	0	0	0	0	0	0
$f_{51}(i)$	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	$f_{61}(i)$
0	$f_{71}(i)$	0	$f_{72}(i)$	0	0	0	0	0

where the elements of $\underline{\underline{E}}$ and $\underline{\underline{F}}$ are given in Tables 2.5 and 2.6 respectively. For the single-loop representation, the modification in Eq. (2.61) is to replace W_C by m times W(1).

Eq. (2.61) can be solved readily once the boundary conditions, or $\underline{u}$, are determined. The safety-injection, let-down, and charging mass flow rates and enthalpies are assumed given. However, the surge mass flow rate and enthalpy must be determined by simultaneously solving the state equations of the flow-path components and the pressurizer. The coupling of flow-path components and pressurizer is discussed in Section 2.9 and Chapter 3.

2.8.2 Negative Flow Case

If reverse flow occurs in Loop j, i.e., if $W(j) < 0$, the state vector equation of Eq. (2.61) must be reconstrcuted, as:

Table 2.5. Elements of matrix $\underline{\underline{E}}$.

Element	Expression
e_{11}	$V_{LP}\left(\frac{\partial \rho}{\partial P}\right)_{LP} + \left(\frac{\partial M'}{\partial P}\right)_{C} + V_{UP}\left(\frac{\partial \rho}{\partial P}\right)_{UP} + \sum_i \left[\left(\frac{\partial M'}{\partial P}\right)_{HL}(i) + \left(\frac{\partial M'}{\partial P}\right)_{SG}(i) + \left(\frac{\partial M'}{\partial P}\right)_{CL}(i)\right.$
e_{12}	$V_{LP}\left(\frac{\partial \rho}{\partial H}\right)_m + \left(\frac{\partial M_C}{\partial H_{m,Lp}}\right)$
e_{13}	$\left(\frac{\partial M'}{\partial H_m}\right)_C$
e_{14}	$V_{UP}\left(\frac{\partial \rho}{\partial H_m}\right)_{UP} + \sum_i \left(\frac{\partial M_{HL}}{\partial H_{m,UP}}\right)(i)$
$e_{15}(i)$	$\left(\frac{\partial M'}{\partial H_m}\right)_{HL}(i) + \left(\frac{\partial M_{HL}}{\partial H_{m,SG}}\right)(i)$
$e_{16}(i)$	$\left(\frac{\partial M'}{\partial H_m}\right)_{SG}(i) + \left(\frac{\partial M_{SG}}{\partial H_{m,CL}}\right)(i)$
$e_{17}(i)$	$\left(\frac{\partial M'}{\partial H_m}\right)_{CL}(i)$
e_{21}	$-V_{LP}$
e_{22}	$M_{LP} + \Delta t W_C$
$e_{23}(i)$	$-\Delta t W(i)\left(\frac{\partial H'}{\partial H_m}\right)_{CL}(i)$
e_{31}	$\left(\frac{\partial E'}{\partial P}\right)_C$
e_{32}	$-\Delta t W_C$

Table 2.5. Elements of matrix $\underline{\underline{E}}$ (continued).

Element	Expression
e_{33}	$(\frac{\partial E'}{\partial H_m})_C + \Delta t W_C (\frac{\partial H'}{\partial H_m})_C$
e_{41}	$-V_{UP}$
e_{42}	$-\Delta t W_C (\frac{\partial H'}{\partial H_m})_C$
e_{43}	$M_{UP} + \Delta t W_C$
$e_{51}(i)$	$(\frac{\partial E'}{\partial P})_{HL}(i)$
$e_{52}(i)$	$-\Delta t W(i)$
$e_{53}(i)$	$(\frac{\partial E'}{\partial H_m})(i) + \Delta t W(i) (\frac{\partial H'}{\partial H_m})_{HL}(i)$
$e_{61}(i)$	$(\frac{\partial E'}{\partial P})_{SG}(i)$
$e_{62}(i)$	$-\Delta t W(i) (\frac{\partial H'}{\partial H_m})_{HL}(i) + \Delta t (\frac{\partial Q_{SG}}{\partial H_m})_{HL}(i)$
$e_{63}(i)$	$(\frac{\partial E'}{\partial H_m})_{SG}(i) + \Delta t W(i) (\frac{\partial H'}{\partial H_m})_{SG}(i)$ $+ \Delta t (\frac{\partial Q_{SG}}{\partial H_m})(i) + \frac{1}{2} \frac{(V \rho C)_{TM}}{(C_p)_{SG}(i)}$
$e_{71}(i)$	$(\frac{\partial E'}{\partial p})_{CL}(i)$
$e_{72}(i)$	$-\Delta t W(i) (\frac{\partial H'}{\partial H_m})_{SG}(i)$
$e_{73}(i)$	$(\frac{\partial E'}{\partial H_m})_{CL}(i) + \Delta t W(i) (\frac{\partial H'}{\partial H_m})_{CL}(i)$

Table 2.6 Elements of matrix $\underline{\underline{F}}$.

Element	Expression	Element	Expression
f_{11}	$-\Delta t$	$f_{17}(i)$	$-\Delta t$
f_{12}	Δt	f_{31}	Δt
f_{13}	$-\Delta t$	$f_{51}(1)$	$-\Delta t\left[H'_{su} - (H_m)_{HL}(1)\right]$
f_{14}	Δt	$f_{61}(i)$	$-\Delta t$
$f_{15}(i)$	$-\Delta t$	$f_{71}(i)$	$\Delta t\left[H'_{si} - (H_m)_{CL}(i)\right]$
$f_{16}(i)$	$-\Delta t$	$f_{72}(i)$	$\Delta t\left[H'_{ch} - (H_m)_{CL}(i)\right]$

$$(\underline{\underline{E}}')^n \, \Delta(\underline{x}')^{n+1} \;=\; (\underline{\underline{F}}')^n \, \underline{u}^{n+1} + (\underline{r}')^n \qquad (2.62)$$

where the state vector $\underline{x}'$ is:

$$\underline{x}' \;=\; [P,\ (H_m)_{LP},\ (H_m)_C,\ (H_m)_{UP},\ \ldots,\ (H_m)_{HL}(j),$$
$$(H_m)_{SG}(j),\ (H_m)_{CL}(j),\ \ldots]^T$$

the residual vector $\underline{r}'$ is:

$$\underline{r}' \;=\; [0,\ r_2,\ r_3,\ r_4,\ \ldots,\ r_5(j),\ r_6(j),\ r_7(j),\ \ldots]^T$$

The elements of $\underline{r}'$ are given in Table 2.7. The coefficient matrix $\underline{\underline{E}}'$ has the following structure:

$\underline{\underline{E}}'$ =

e_{11}	e_{12}	e_{13}	e_{14}	...	$e_{15}(j)$	$e_{16}(j)$	$e_{17}(j)$	...
e_{21}	e_{22}	0	0		0	0	0	
e_{31}	e_{32}	e_{33}	0		0	0	0	
e_{41}	0	e_{42}	e_{43}		$e_{44}(j)$	0	0	
⋮								
$e_{51}(j)$	$e_{52}(j)$	0	0		$e_{53}(j)$	0	0	
$e_{61}(j)$	0	0	0		$e_{62}(j)$	0	0	
$e_{71}(j)$	0	0	0		0	$e_{72}(j)$	$e_{73}(j)$	
⋮								

Table 2.7. Elements of vector $\underline{r}'$.

Element	Expression
r_2	$\Delta t \sum_{i \neq j} W(i) \left[(H')_{CL}(i) - (H_m)_{LP}\right]$
r_3	$\Delta t W_C \left[(H_m)_{LP} - (H')_C\right]$
r_4	$\Delta t \sum_{j} \left[(H_m)_{UP} - (H_m)_{HL}(j)\right]$
$r_5(j)$	$\Delta t W(j) \left[(H')_{CL}(j) - (H_m)_{LP}\right]$
$r_6(j)$	$\Delta t W(j) \left[(H')_{SG}(j) - (H')_{CL}(j)\right]$
$r_7(j)$	$\Delta t W(j) \left[(H')_{HL}(j) - (H')_{SG}(j)\right]$

and the Matrix $\underline{\underline{F}}'$ has the following structure:

$\underline{\underline{F}}' =$

f_{11}	f_{12}	f_{13}	f_{14}	...	$f_{15}(j)$	$f_{16}(j)$	$f_{17}(j)$	0	...	...
0	...					0	0			
0	...					0	0	0	f_{31}	0
0						0	0			
⋮										
0	$f_{51}(j)$	0	$f_{52}(j)$	0	...	0	0			
0	0	0	...			0	0		0	$f_{61}(j)$
$f_{71}(1)$	0	0	...							
⋮										

Elements in $\underline{\underline{F}}'$ different than those in $\underline{\underline{F}}$ are listed in Table 2.8, and for $\underline{\underline{F}}'$ and $\underline{\underline{F}}$, the different elements are listed in Table 2.9.

2.9 Solution Technique

The solutions are obtained by a Gauss elimination process. The state equations, Eqs. (2.61) and (2.62), are reduced to the following equation by an forward elimination:

$$\Delta P^{n+1} = K_{11} W_{su}^{n+1} + K_{12} \qquad (2.63)$$

where K_{11} and K_{12} are constants. By a similar elimination process, the pressurizer state equations (Chapter 3) are reduced to an equation same as Eq. (2.63) but with different constants:

Table 2.8. Elements different in $\underline{\underline{E}}'$ than $\underline{\underline{E}}$.

Element	Expression
$e_{15}(j)$	$\left(\frac{\partial M'}{\partial H_m}\right)_{CL}(j)$
$e_{17}(j)$	$\left(\frac{\partial M'}{\partial H_m}\right)_{HL}(j)$
e_{22}	$M_{LP} + \Delta t \sum_{i \neq j} W(i)$
$e_{23}(j)$	0
e_{43}	$M_{UP} + \Delta t \left[W_C + \sum_j W(J)\right]$
$e_{44}(j)$	$-\Delta t W(j) \left(\frac{\partial H'}{\partial H_m}\right)_{HL}(j)$
$e_{51}(j)$	$\left(\frac{\partial E'}{\partial P}\right)_{CL}(j)$
$e_{53}(j)$	$\left(\frac{\partial E'}{\partial H_m}\right)_{CL}(j) - \Delta t W(j) \left(\frac{\partial H'}{\partial H_m}\right)_{CL}(j)$
$e_{62}(j)$	$\Delta t W(j) \left(\frac{\partial H'}{\partial H_m}\right)_{CL}(j)$
$e_{63}(j)$	$\left(\frac{\partial E'}{\partial H_m}\right)_{SG}(j) - \Delta t W(j) \left(\frac{\partial H'}{\partial H_m}\right)_{SG}(j)$
$e_{71}(j)$	$\left(\frac{\partial E'}{\partial P}\right)_{HL}(j)$
$e_{72}(j)$	$\Delta t W(j) \left(\frac{\partial H'}{\partial H_m}\right)_{SG}(j)$
$e_{73}(j)$	$\left(\frac{\partial E'}{\partial H_m}\right)_{HL}(j) - \Delta t W(j) \left(\frac{\partial H'}{\partial H_m}\right)_{HL}(j)$

Table 2.9. Elements different in $\underline{\underline{F}}'$ than $\underline{\underline{F}}$.

Element	Expression
$f_{51}(j)$	$\Delta t\left[H'_{si} - (H_m)_{CL}(j)\right]$
$f_{52}(j)$	$\Delta t\left[H'_{ch} - (H_m)_{CL}(j)\right]$
$f_{71}(1)$	$-\Delta t\left[H'_{su} - (H_m)_{HL}(1)\right]$

$$\Delta P^{n+1} = K_{11} W_{su}^{n+1} + K_{12} \quad (2.64)$$

Combining Eqs. (2.63) and (2.64), the surge mass flow rate at time n+1 is determined by the following:

$$W_{su}^{n+1} = (K_{22} - K_{12})/(K_{11} - K_{21}) \quad (2.65)$$

Then, by backward substitution all the state variables at time n+1 are determined. This method allows us to implicitly couple the pressurizer state equations with those of flow path components, which makes the calculations more stable than an explicit method.

The time step size, however, is limited by the transport time through the shortest flow path, which is in the reactor core. In the present calculation, the time step size is chosen to be the fluid transport time through the core, and is allowed to vary with flow rate. Since the calculation ranges from full flow to pump coastdown and natural circulation, variable time step size is particularly advantageous in saving computing time.

2.10 Summary

The Reactor Coolant System is divided into following components: reactor core, reactor vessel upper plenum, reactor vessel lower plenum, hot leg, cold leg, steam generator U-tubes, and pressurizer. The first six components are called flow path components. Except for the reactor vessel plena and the pressurizer, the enthalpy profile is assumed to be linear at all times within each control volume. A drift-flux formulation is used in two-phase flow conditions. In the plena, a uniform mixing model is used. That is, enthalpy profile is flat and equal to the average enthalpy in the control volume.

A state-space model is constructed consisting of the energy equations and the overall mass conservation equation for the flow path components. The state variables are the mixture enthalpies in each component and the system pressure. In this model a donor-cell differencing method is used. Flow is assumed uniform within each loop and allowed to reverse direction. The state-space model for the flow path components is implicitly coupled with that of pressurizer. The computer model has provision accommodating up to four independent loops.

Heat transfer in the core is characterized by a single-node fuel-rod transient conduction model. In this model, fission power is regarded as an input. Decay power is determined by summing the powers produced by seven decay-product groups.

Steam generator heat transfer rate is characterized by a steady-state, zero-dimensional analysis, defined in terms of the log-mean temperature difference and the overall heat transfer coefficient. Heat transfer between the primary and secondary side is solved implicitly.

CHAPTER 3

Pressurizer Model

3.1 Introduction

The pressurizer provides features for pressure control in the Reactor Coolant System. A compressible vapor space gives room for expansion and the contents of a liquid space form a supply for contraction of the RCS coolant. Figure 3.1 shows a sketch of a PWR pressurizer. The design requirements of a PWR pressurizer are that it must be able to compensate the RCS coolant volume variations, without water reaching the relief valves or uncovering the heaters, for the following operational transients [C4] [G1]:

1) Power ramp at a rate of 5% full power per minute.
2) Negative step power change of 10% full power.
3) Reactor trip.
4) Turbine trip.

During full power steady state operation, about half of the pressurizer volume is occupied by saturated liquid, and half saturated vapor. During load changes, the pressurizer controls the RCS pressure by maintaining the liquid level according to the programmed set-point as a function of average coolant temperature, as shown in Figure 3.2. The average coolant temperature is programmed independently to vary as a function of power level, as shown in Figure 3.3. The pressurizer level is controlled by following the

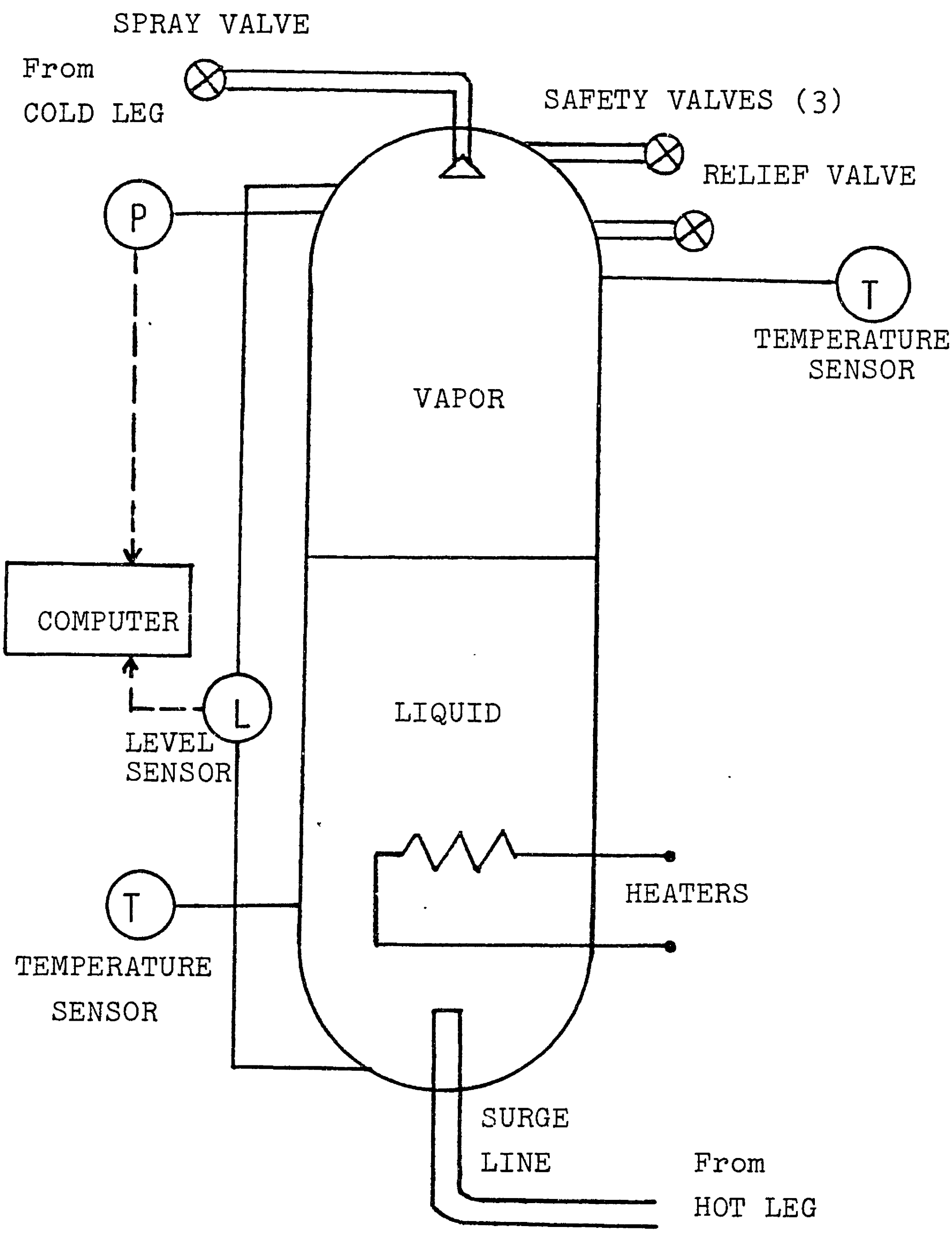


FIGURE 3.1 Schematic of a Pressurizer.

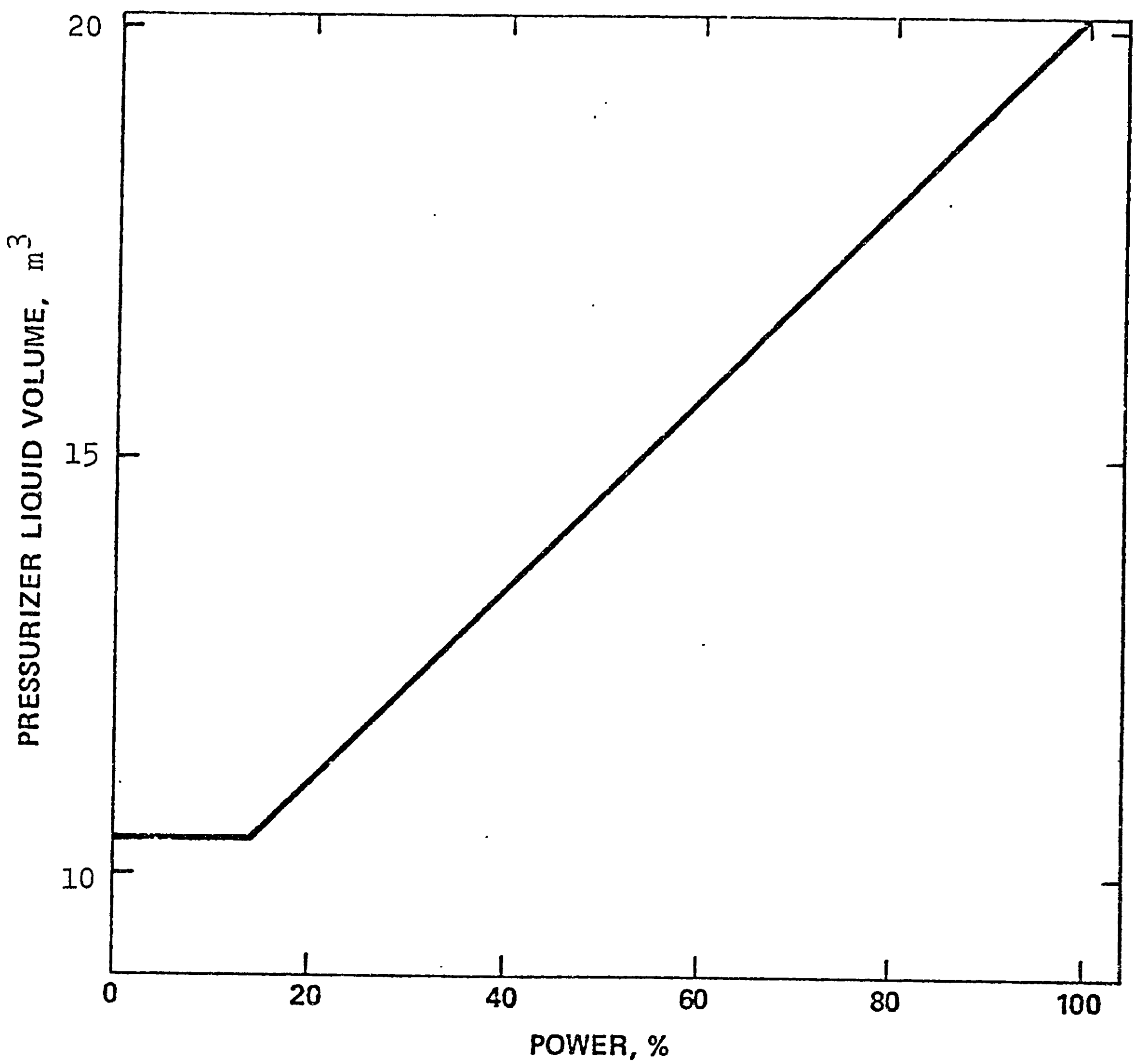


FIGURE 3.2 Pressurizer Liquid Volume Program.
(Adopted from Ref. [W4])

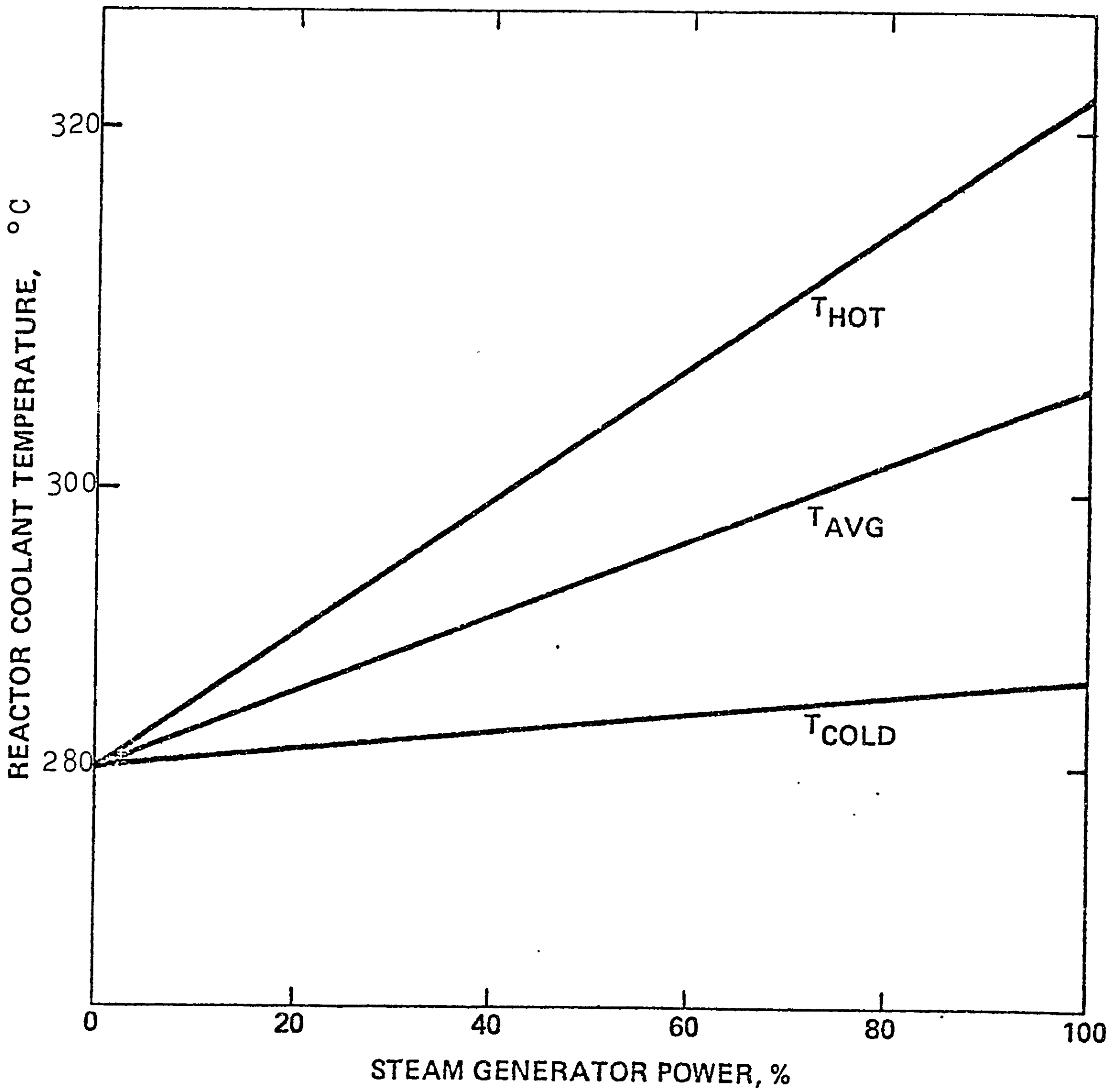


FIGURE 3.3 Temperature Control Program.
(Adopted from Ref. [W4])

pre-programmed control actions given in Figure 3.4.

A decrease in power results in contraction of the coolant, hence, lowers the pressurizer level and causes the system pressure to fall. The loss of pressure is partially offset by flashing of liquid into steam in the pressurizer. A pressure controller activates the pressurizer heaters to raise the pressure by generating more steam. If the pressurizer level drops sufficiently below its programmed set-point, charging pumps in the Chemical and Volume Control System (CVCS) are automatically started to add coolant to the RCS, thereby acting to restore the pressurizer level.

An increase in power results in expansion of the coolant, and causes a surge of liquid into the pressurizer compressing the steam and raising the pressure. The pressure controller opens the pressurizer spray valve drawing subcooled water from the cold leg into the steam dome. The spray condenses some of the steam in the pressurizer and lowers the pressure. If the pressurizer level rises sufficiently above its programmed set-point, letdown valves in the CVCS are automatically opened to cause coolant to leave the RCS, and thereby to restore the pressurizer level.

3.2 Literature Review

The most commonly used pressurizer models in the literature are the homogeneous thermal-equilibrium model and the two-region thermal non-equilibium model.

PRESSURIZER LEVEL ERROR (m)	ACTION[a]
+0.965	HIGH LEVEL ERROR ALARM
+0.330	ENERGIZE ALL PRESSURIZER HEATERS & BACKUP SIGNAL TO STOP 2 BACKUP CP'S
+0.279	ALL PRESSURIZER HEATERS OFF
-0.102	STOP BACKUP CP 2
-0.152	STOP BACKUP CP 3
-0.229	START BACKUP CP 2
-0.356	START BACKUP CP 3
-0.381	LOW LEVEL ERROR ALARM AND BACKUP SIGNAL TO START ALL CP'S

[a] CP = CHARGING PUMP

FIGURE 3.4 Pressurizer Control Actions Resulting From High or Low Level Error Signals. (Adopted from Ref. [W4])

The homogeneous model is often used for real-time simulation, where computation speed is at a premium [C1] [T1]. The disadvantage of the homogeneous model is its physical limitations. For example, in an actual insurge transient, the cold insurge coolant compresses and superheats the vapor. The vapor temperature and the pressurizer pressure increase (the pressure rise is depressed somewhat by steam condensing on the vessel walls). However, if the homogeneous model were used, the cold insurge coolant would also act to reduce the the pressurizer temperature, and may actually cause the pressure to drop.

The two-region model is more detailed than the homogeneous model and naturally more complicated. In the two-region model, pressurizer is divided into vapor and liquid regions. The two regions have the same pressure, but are thermodynamically independent. However, there are mass exchanges between them through spontaneous condensation (or rainout) and spontaneous evaporation (or flashing).

There are many two-region models in the literature. However, they differ mainly in the constitutive relations for rainout and flashing mass flow rates. Nahavandi [N1] and Baron [B4] used an explicit method, where rainout and flashing mass flow rates are determined by the qualities in the vapor and liquid regions. Transport delay models are incorporated to account for droplet-falling and bubble-rising time delays. Recently, Baltra [B1] and Geffray [G1] used a similar approach with different transport delay models.

In these four models the surge mass flow rate must be given as an input. That is, an insurge (without spray) always means increasing pressure, and an outsurge (without heater action) always means decreasing pressure. No flashing nor rainout is allowed for increasing pressure.

In the steam drum model by Moeck [M7], which also was used to treat pressurizer dynamics, rainout and flashing mass flow rates are functions of the rate of change of pressure, determined by using an isentropic assumption. Again, insurge or outsurge must be known as input to this model.

It should be pointed out that all the models aformentioned have neglected the vessel wall, which may act as a heat sink or a heat source in the pressurizer.

A three-region model proposed by Baggoura [B3] divides liquid volume into upper and lower regions. The lower region contains the cold insurge liquid. The upper region contains the hot liquid. A mixing coefficient is used to account for the interaction between the two. Empirical correlations are used to determine mass flow rate of flashing and rainout. Wall condensation mass flow rate is calculated by solving a one-dimensional Fourier heat conduction equation using Simpson's rule of integration. Unlike the others, this model uses hot-leg pressure as the boundary condition instead of the surge mass flow rate. The surge mass flow rate is calculated by including a fine-mesh surge line momentum model.

More recently, Kim [K3] proposed a more elaborate model which divides the fluid volume in the pressurizer into four regions: cold liquid, hot liquid, vapor, and non-condensable gas. The cold liquid region accounts for the observation made by Kim that during an insurge transient the insurge cold liquid mixes only slightly with the hot liquid already present (causing a thermal stratification in the liquid region). The non-condensable gas region accounts for the presence of hydrogen, nitrogen, and fission gases. Furthermore, the model also divides the vessel wall into three regions adjacent to the cold liquid, hot liquid, and vapor regions respectively, to account for the thermal interactions between the fluids and the vessel wall. The fluid-wall interactions are responsible for wall condensation and "flashing suppression"--that is, the cold wall takes heat away from the hot liquid during an outsurge followed by an insurge of cold liquid, and thus reduces the flashing of hot liquid by a certain amount. To simplify calculations, a number of assumptions are made, such as the vapor is always saturated and the temperature drop in the wall condensate film is neglected.

3.3 Two-Fluid Pressurizer Model

3.3.1 Conservation Equations of Mass and Energy

A pressurizer model using a state-space representation is adopted in this research to treat the pressurizer dynamics. This model is based on two-fluid assumptions, namely:

1) Uniform properties for each phase.
2) Neither phase can be metastable.
3) Rainout to maintain saturated vapor to prevent subcooling.
4) Flashing to maintain saturated liquid to prevent superheating.
5) Spray enters the liquid as a saturated liquid.
6) No heat transfer at vapor-liquid interface.

By the above assumptions, the pressurizer can be in any one of the following four conditions at any instant of time:

1) Liquid subcooled and vapor superheated.
2) Both phases saturated.
3) Liquid saturated and vapor superheated.
4) Liquid subcooled and vapor saturated.

Since the compressibility of vapor is much greater than that of liquid, pressure is much more sensitive to the condition of vapor than liquid. Therefore, to model accurately the constitutive relations in the vapor region is essential to good pressure prediction. Non-condensable gas has been shown to affect pressurizer behavior [L2]. However, the quantity of non-condensable gas in the pressurizer is not measured, and a large quantity of it is produced only in core uncovering accidents. Therefore, non-condensable gas is neglected in the present model. The hot liquid is important in the calculation of flashing mass flow rate. The cold liquid behaves essentially like a piston. The volumes for both liquids must be known as functions of time. A direct application of a two-fluid model assumes perfect mixing between the hot and cold liquids os ised here. However, suitable model adjustments are made to distinguish hot/cold effects. For example, the volume of the cold liquid is determined by solving the conservation equation of mass explicitly.

Figure 3.5 shows the schematic of a two-fluid pressurizer model. The external boundary conditions are spray mass flow rate and enthalpy, W_{sp} and H_{sp}, relief and safety valve discharge mass flow rates, W_{rv} and W_{sv}, surge mass flow rate and enthalpy, W_{su} and H_{su}, and heater power, Q_h. The interfacial boundary conditions are mass flow rates for condensation on spray and wall condensation, rainout, and flashing (W_{sc}, W_{wc}, W_{ro}, and $W_{f\ell}$, respectively). Some of these boundary conditions are specified, such as spray and heater power. The rest of them are unknowns which require additional constitutive relations in order to solve the conservation equations in a closed form. To model these constitutive relations adequately is one of the major challenges in pressurizer modeling.

The mass and energy conservation equations of the two-fluid pressurizer model are as follows:

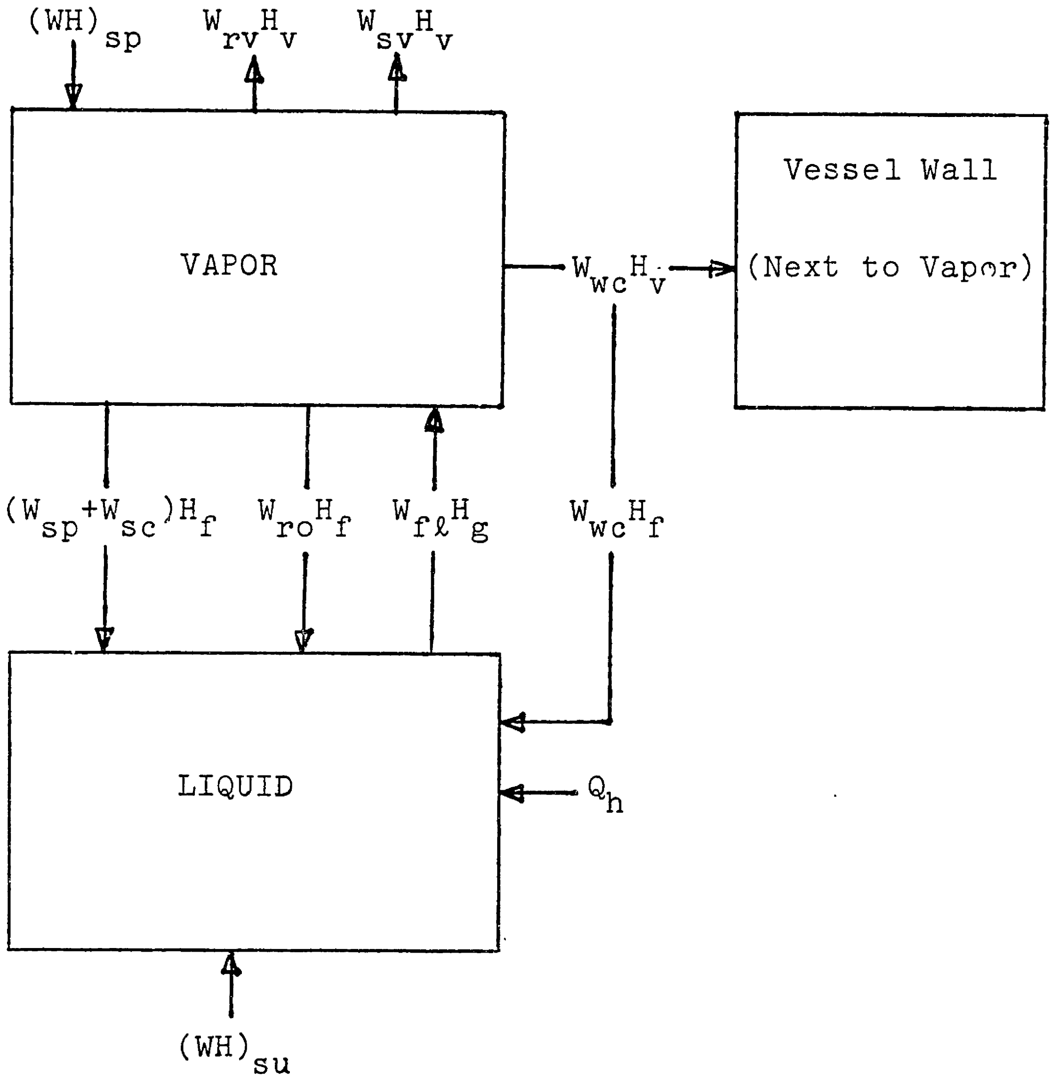


FIGURE 3.5 Schematic of A Two-Fluid Pressurizer Model.

Vapor:

$$\frac{dM_v}{dt} = W_{f\ell} - W_{ro} - W_{sc} - W_{rv} - W_{sv} - W_{wc} \tag{3.1}$$

$$\frac{d(MH)_v}{dt} = W_{sp}(H_{sp} - H_f) + W_{f\ell}H_g - (W_{ro} + W_{sc})\ H_f$$

$$- (W_{rv} + W_{sv})\ H_v - W_{wc}H_v + V_v \frac{dP}{dt} \tag{3.2}$$

Liquid:

$$\frac{dM_\ell}{dt} = W_{su} + W_{sp} + W_{sc} + W_{wc} + W_{ro} - W_{f\ell} \tag{3.3}$$

$$\frac{d(MH)_\ell}{dt} = (WH)_{su} + (W_{sp} + W_{sc} + W_{wc} + W_{ro})\ H_f$$

$$- W_{f\ell}H_g + Q_h + V_\ell \frac{dP}{dt} \tag{3.4}$$

Again as in Chapter 2, using the advantages of equations that are independent of enthalpy reference point (where H_{su} is used for the mass equation multiple), rewrite Eqs. (3.2) and (3.4), respectively, as:

$$-V_v \frac{dP}{dt} + M_v \frac{dH_v}{dt} + (H_v - H_{su}) \frac{dM_v}{dt} = W_{sp}(H_{sp} - H_f)$$

$$+ W_{f\ell}(H_g - H_{su}) - (W_{ro} + W_{sc})(H_f - H_{su})$$

$$- (W_{wc} + W_{rv} + W_{sv})(H_v - H_{su}) \tag{3.5}$$

and

$$-V_\ell \frac{dP}{dt} + M_\ell \frac{dH_\ell}{dt} + (H_\ell - H_{su}) \frac{dM_\ell}{dt}$$

$$= (W_{sp} + W_{sc} + W_{wc} + W_{ro})(H_f - H_{su})$$

$$- W_{f\ell}(H_g - H_{su}) + Q_h \qquad (3.6)$$

Equations (3.1), (3.3), (3.5), and (3.6) are the state equations of the two-fluid pressurizer model. Associated to these equations there are four independent state variables. There exist several possible choices of state variables. Here, the state variables chosen are pressure, P, liquid enthalpy, H_ℓ, vapor enthalpy, H_v, and vapor fraction, α_p. Vapor fraction is defined in terms of the vapor volume $(V_v)_p$ and the total volume $(V)_p$ in the pressurizer:

$$\alpha_p = (V_v)_p/V_p \qquad (3.7)$$

Rewrite Eqs. (3.1), (3.3), (3.5), and (3.6) in terms of the four state variables, to obtain:

$$\left(\frac{\partial M_v}{\partial P}\right) \frac{dP}{dt} + \left(\frac{\partial M_v}{\partial H_v}\right) \frac{dH_v}{dt} + \left(\frac{\partial M_v}{\partial \alpha_p}\right) \frac{d\alpha_p}{dt} = W_{f\ell} - W_{ro} - W_{sc} - W_{wc}$$

$$- W_{rv} - W_{sv} \qquad (3.8)$$

$$\left(\frac{\partial M_\ell}{\partial P}\right) \frac{dP}{dt} + \left(\frac{\partial M_\ell}{\partial H_\ell}\right) \frac{dH_\ell}{dt} + \left(\frac{\partial M_\ell}{\partial \alpha_p}\right) \frac{d\alpha_p}{dt} = W_{su} + W_{sp} + W_{sc} + W_{wc}$$

$$+ W_{ro} - W_{f\ell} \qquad (3.9)$$

$$\left[-V_v + (H_v - H_{su}) \frac{\partial M_v}{\partial P}\right] \frac{dP}{dt} + \left[M_v + (H_v - H_{su}) \frac{\partial M_v}{\partial H_v}\right] \frac{dH_v}{dt}$$

$$+ \left[(H_v - H_{su}) \frac{\partial M_v}{\partial \alpha_p}\right] \frac{d\alpha_p}{dt} = W_{sp}(H_{sp} - H_f) + W_{f\ell}(H_g - H_{su})$$

$$- (W_{ro} + W_{sc})(H_f - H_{su}) - (W_{wc} + W_{rv} + W_{sv})(H_v - H_{su}) \tag{3.10}$$

and

$$\left[-V_\ell + (H_\ell - H_{su})\left(\frac{\partial M_\ell}{\partial P}\right)\right] \frac{dP}{dt} + \left[M_\ell + (H_\ell - H_{su})\left(\frac{\partial M_\ell}{\partial H_\ell}\right)\right] \frac{dH_\ell}{dt}$$

$$+ \left[(H_\ell - H_{su})\left(\frac{\partial M_\ell}{\partial \alpha_p}\right)\right] \frac{d\alpha_p}{dt} = (W_{sp} + W_{sc} + W_{wc} + W_{ro})$$

$$(H_f - H_{su}) - W_{f\ell}(H_g - H_{su}) + Q_n \tag{3.11}$$

where the expressions of the partial derivatives are given in Table 3.1.

3.3.2 Wall Condensation Model

Wall condensation is often neglected in pressurizer modeling (as in Refs. [B1] [B4] [G1] [N1]). However, Robinson [R6] and Kim [K3] have shown that wall condensation plays an important role in absent of spray.

Condensation at the vessel wall occurs only if the wall surface temperature is lower than the vapor temperature and the saturation temperature. Robinson proposes that heat transfer from the vapor to the wall be represented according

Table 3.1. Partial derivatives in Eqs. (3.8) to (3.11).

Partial	Expression
$\left(\frac{\partial M_v}{\partial P}\right)_p$	$\left(V_v \frac{\partial \rho_v}{\partial P}\right)_p$
$\left(\frac{\partial M_v}{\partial H_v}\right)_p$	$\left(V_v \frac{\partial \rho_v}{\partial H_v}\right)_p$
$\left(\frac{\partial M_v}{\partial \alpha}\right)_p$	$\left(V_v \rho_v\right)_p$
$\left(\frac{\partial M_\ell}{\partial P}\right)_p$	$\left(V_\ell \frac{\partial \rho_\ell}{\partial P}\right)_p$
$\left(\frac{\partial M_\ell}{\partial H_\ell}\right)_p$	$\left(V_\ell \frac{\partial \rho_\ell}{\partial H_\ell}\right)_p$
$\left(\frac{\partial M_\ell}{\partial \alpha}\right)_p$	$-\left(V_\ell \rho_\ell\right)_p$

Table 3.2. Pressurizer time constants.

	Connecticut Yankee*	MIT**
T_w (°C)	350	180
ρ_w (kg/m^3)	7900	7700
$(C_p)_w$ (J/kg-°C)	600	530
k_w (W/m-°C)	46	16
h_{wc} (W/m^2-°C)	10000	4000
L_w (m)	0.143	0.0083
τ_w (s)	2160	14.5

*Carbon steel
**Stainless steel

to Newton's law of cooling as:

$$Q_{wc} = h_{wc}A_w(T_v - T_w) \quad (3.12)$$

The condensation mass flow rate is then calculated as:

$$W_{wc} = \frac{Q_{wc}}{H_{fg} + C_{pv}(T_v - T_{sat})} \quad (3.13)$$

Where it is assumed that the heat transfer coefficient is a constant, and that the wall temperature is a constant equal to the saturation temperature at the unset of condensation. With these assumptions, the model has avoided calculation of the wall surface temperature and the condensation heat transfer coefficient. A shortcoming of this model is that according to Nusselt's theory on condensation, [C3] the temperature of condensing film at the vapor interface is approximately equal to the saturation temperature, and Eq. (3.12) should have been the following:

$$Q_{wc} = h_{wc}A_w(T_{sat} - T_w) \quad (3.14)$$

where the condensation heat transfer coefficient is [R8]:

$$h_{wc} = 0.943 \left[\frac{g\rho_\ell(\rho_\ell - \rho_v)\, k_\ell^3\, H_{fg}}{\mu_\ell\, \ell(T_v - T_w)} \right]^{1/4} \quad (3.15)$$

In Kim's model, condensation heat transfer is obtained by solving the heat conduction equation in the wall assuming a slab geometry and assuming that the wall surface temperature is equal to the saturation temperature, or equivalently, the temperature drop across the condensate film is neglected. Therefore, Eq. (3.15) is not used in Kim's model.

In the present model, a method has been adopted that includes solving both the wall heat conduction equation and the film condensation heat transfer equation. Heat storage in the wall is calculated according to a one-node representation as:

$$(\rho C_p)_w \, L_c \, \frac{d\langle T_w \rangle}{dt} = q''_{wc} \tag{3.16}$$

where $\langle T_w \rangle$ is the average wall temperature over the depth L_w. The heat flux at vapor interface is given by:

$$q''_{wc} = U_{wc}(T_{sat} - \langle T_w \rangle) \tag{3.17}$$

where

$$U_{wc} = \frac{1}{1/h_{wc} + L_k/k_w} \tag{3.18}$$

The condensation mass flow rate is determined by:

$$W_{wc} = \frac{q''_{wc} A_w}{H_v - H_f} \tag{3.19}$$

The time constant of the wall is the following:

$$\tau_w = (\rho C_p)_w L_c \left(\frac{1}{h_{wc}} + \frac{L_k}{k_w}\right) \quad (3.20)$$

Eq. (3.20) suggests that the transient wall conduction is determined by choosing a proper thermal-capacity equivalent length, L_c, and the thermal-conductivity equivalent length, L_k. If the entire wall participated in the transient (i.e., $L_c = L_w$ and $L_k = L_w/3$), the time constant of Eq. (3.20) is typically on the order of two thousand seconds (Table 3.2), and condensation would be negligible. Condensation usually occurs during an insurge transient, which lasts on the order of one to several hundred seconds depending on the severity of the transient. Therefore, only a fraction of the wall actually participates in the condensation process. A method to determine of the wall equivalent lengths is given in Appendix E.

The present wall condensation model (Eqs. (3.16) to (3.20)) is better than Kim's model in on-line applications. Here, the solution procedure for a one-node conduction equation is straight forward and the equivalent lengths, L_c and L_k, are readily adopted to model wall condensation for a particular transient. Whereas in Kim's model, summation of an infinite-series solution for a transient conduction equation must be carried out every time step.

3.3.3 Rainout and Flashing Models

Spontaneous rainout and flashing processes may be illustrated by the T-s diagram shown in Figure 3.6. Consider isentropic expansion and contraction is valid, and consider liquid and vapor phases that are in thermal equilibrium at an initial pressure P_o. If depressurization occurs such that pressure drops from P_o to P_1, by following the isentropic line, vapor will be in a two-phase state at v_1. Saturated liquid droplets will form and fall out of the vapor phase. The phase separation process is, therefore, called rainout. Meanwhile, liquid will be at state ℓ_1. Bubbles will form in the liquid phase and rise toward the liquid-vapor interface. This process is called flashing. On the other hand, if pressurization occurs without heat transfer, vapor will be superheated at state v_2, and liquid subcooled at ℓ_2. No phase change takes place. Occurence of rainout and flashing depends on the final conditions of the liquid and vapor phases respectively.

From the illustration of Figure 3.6, one can show that rainout and flashing mass flow rates and the mass-energy equations depend on the rate of change of pressure. Rainout takes place when initially saturated vapor is subject to decreasing pressure. With no other heat or mass transfer processes, rainout is described by the conservation of mass and energy equations for the vapor as follows:

$$\frac{dM_v}{dt} = -W_{ro} \tag{3.21}$$

and

$$\frac{d(HM)_v}{dt} = -W_{ro}H_f + V_v \frac{dP}{dt} \quad (3.22)$$

Assuming vapor is saturated during the rainout process, i.e., $H_v = H_g(P)$, combine Eqs. (3.21) and (3.22), and solve for W_{ro}, obtain the following relation:

$$W_{ro} = \frac{-M_v\left(\frac{1}{\rho_v} - \frac{dH_g}{dP}\right)\frac{dP}{dt}}{H_{fg}} \quad (3.23)$$

Similarly, flashing takes place when initially saturated liquid experiences decreasing pressure. The conservation equations of liquid mass and energy for flashing alone are:

$$\frac{dM_\ell}{dt} = -W_{f\ell} \quad (3.24)$$

and

$$\frac{d(HM)_\ell}{dt} = -W_{f\ell}H_g + V_\ell \frac{dP}{dt} \quad (3.25)$$

Assuming liquid stays at saturation, i.e., $H_\ell = H_f(P)$, flashing mass flow rate is obtained by combining Eqs. (3.24)

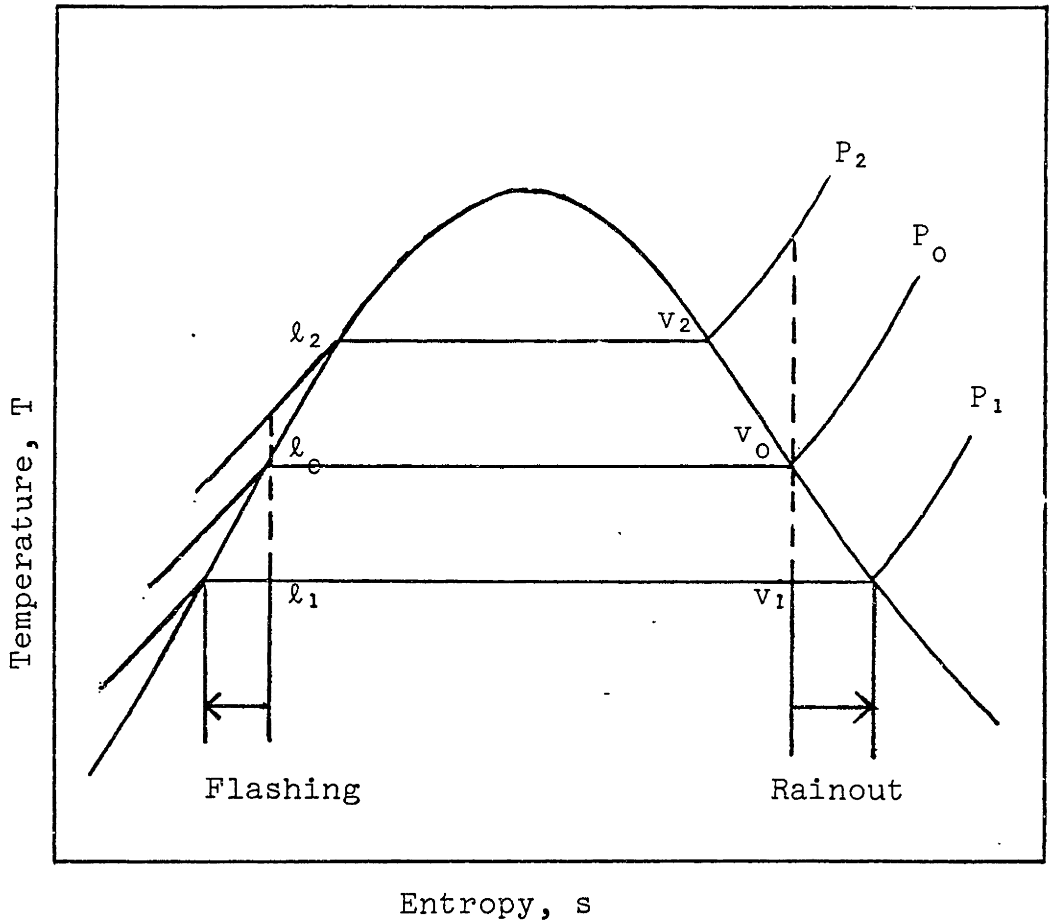


FIGURE 3.6 T-s Diagram for Isentropic Rainout and Flashing.

and (3.25) as:

$$W_{f\ell} = \frac{-M_{\ell}\left(\frac{dH_f}{dP} - \frac{1}{\rho_v}\right)\frac{dP}{dt}}{H_{fg}} \qquad (3.26)$$

Eqs. (3.23) and (3.26) show that rainout and flashing mass flow rates are proportional to the time rate of change of pressure. However, when other heat transfer processes exist in the pressurizer such as spray and heating, Eqs.(3.23) and (3.26) do not properly represent W_{ro} and $W_{f\ell}$, respectively. Generalizations that properly account both for extra processes and for finite time steps are given in Section 3.4.2.

In a condition in which stratification of saturated and subcooled liquid exists in the pressurizer, Eq. (3.26) is used to estimate the flashing mass flow rate of the saturated liquid (using for M_{ℓ} only the mass of saturated liquid).

3.3.4 Spray Condensation Model

In deriving a relation for the mass flow rate of condensation caused by spray, the following assumptions are made:

1) Spray reaches a saturated liquid condition before leaving the vapor region.
2) The amount of spray condensate is determined entirely from the energy required to bring

spray to saturation.

3) Spray condensation occurs instantaneously.

Therefore, the energy balance between the spray and its condensate before and after condensation is the following:

$$(WH)_{sp} + W_{sc}H_v = (W_{sp} + W_{sc})\ H_f \tag{3.27}$$

Solve for W in Eq. (3.27), to obtain:

$$W_{sc} = \left(\frac{H_f - H_{sp}}{H_v - H_f}\right) W_{sp} \tag{3.28}$$

3.3.5 Relief and Safety Valve Models

The mass flow rates through the relief and safety valves are determined by the critical mass flow rate equation for an ideal gas, as [L1]:

$$W_{rv} = A_{rv} \left[Kg\ P\ \rho_v \left(\frac{2}{K + 1}\right)^{\frac{K+1}{K-1}}\right]^{1/2} \tag{3.29}$$

where K is the thermal capacity ratio: C_p/C_v. Eq. (3.29) is solved explicitly in the present model. The valve action is approximated by a step function (i.e., the valve is either closed or fully open).

3.3.6 Spray Model

The spray mass flow rate, W_{sp}, can be calculated using the following relation:

$$\Delta P_{sp} = \frac{K_{sp} W_{sp}^2}{A_{sp}^2} \tag{3.30}$$

where

A_{sp} = spray valve area,

K_{sp} = spray valve loss coefficient, and

ΔP_{sp} = pressure drop across spray valve.

Pressure drop across spray valve is equal to the sum of pressure differences (in a loop starting at spray line inlet, through spray line, pressurizer, surge line, hot leg, steam generator primary side, and pump suction leg, ending at pump discharge leg) as:

$$\Delta P_{sp} = (\Delta P_{HL} + \Delta P_{SG} + \Delta P_{CL}) + g(\rho_{\ell})_p(L_p + Z_p) - g\ \rho_{CL} Z_{sp} \tag{3.31}$$

where

ΔP_{HL} = pressure drop in hot leg,

ΔP_{SG} = pressure drop in steam generator primary side,

ΔP_{CL} = pressure drop in cold leg,

L_p = pressurizer liquid level,

Z_p = elevation of pressurizer inlet nozzle from hot leg, and

Z_{sp} = elevation of spray nozzle from cold leg.

Where friction losses in the spray and surge lines are neglected. Hence,

$$W_{sp} = A_{sp} \left[\frac{\Delta P_{sp}}{K_{sp}}\right]^{1/2} \tag{3.32}$$

where the valve loss coefficient, K_{sp}, is assumed constant and equal to the value at the specified maximum spray mass flow rate.

3.4 Numerical Solution

3.4.1 Mass-Energy Equations in Finite-Difference Form

Use semi-implicit methods to obtain finite-difference counterparts of Eqs. (3.8) to (3.11). Express the results as a state vector equation:

$$\underline{\underline{A}}_p^n \, \Delta\underline{x}_p^{n+1} = \underline{\underline{C}}_p^n \, \underline{u}_p^{n+1} + \underline{\underline{D}}_p^n \, \underline{v}^{n+1} \tag{3.33}$$

where the vector $\underline{x}_p$ is the state vector:

$$\underline{x}_p = [P, H_\ell, H_v, \alpha_p]^T$$

Vector $\underline{u}_p$ consists of the following boundary conditions:

$$\underline{u}_p = [W_{su}, W_{sp}, W_{sc}, W_{wc}, W_{rv}, W_{sv}, Q_h]^T$$

Vector $\underline{v}$ consists of the following interfacial mass exchange terms:

$$\underline{v} = [W_{ro}, W_{f\ell}]^T$$

and Matrices $\underline{\underline{A}}_p$, $\underline{\underline{C}}_p$, and $\underline{\underline{D}}_p$ have the following structures:

$\underline{\underline{A}}_p$ =

a_{11}	0	a_{12}	a_{13}
a_{21}	a_{22}	0	a_{23}
a_{31}	0	a_{32}	a_{33}
a_{41}	a_{42}	0	a_{43}

$\underline{\underline{C}}_p$ =

0	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	0
0	c_{21}	c_{22}	c_{23}	0	0	c_{24}
0	0	c_{31}	c_{32}	c_{33}	c_{34}	0
c_{41}	c_{42}	c_{43}	c_{44}	0	0	0

and

$\underline{\underline{D}}_p$ =

d_{11}	d_{12}
d_{21}	d_{22}
d_{31}	d_{32}
d_{41}	d_{42}

where the elements of $\underline{\underline{A}}_p$, $\underline{\underline{C}}_p$, and $\underline{\underline{D}}_p$ are given in Tables 3.3, 3.4, and 3.5, respectively.

3.4.2 Closure by Rainout/Flshing Relations

In Eq. (3.33), the boundary conditions left unresolved are W_{ro}, $W_{f\ell}$, and W_{su}. To determine W_{ro} and $W_{f\ell}$, apply Assumptions 3 and 4 in Section 3.3.1. Consider that a

Table 3.3. Elements of matrix $\underline{\underline{A}}_p$.

Element	Expression	Element	Element
a_{11}	$-V_v + (H_v - H_{su}) \frac{\partial M_v}{\partial P}$	a_{31}	$\frac{\partial M_v}{\partial P}$
a_{12}	$M_v + (H_v - H_{su}) \frac{\partial M_v}{\partial H_v}$	a_{32}	$\frac{\partial M_v}{\partial H_v}$
a_{13}	$(H_v - H_{su}) \frac{\partial M_v}{\partial \alpha_p}$	a_{34}	$\frac{\partial M_v}{\partial \alpha_p}$
a_{21}	$-V_\ell + (H_\ell - H_{su}) \frac{\partial M_\ell}{\partial P}$	a_{41}	$\frac{\partial M_\ell}{\partial P}$
a_{22}	$M_\ell + (H_\ell - H_{su}) \frac{\partial M_\ell}{\partial H_\ell}$	a_{42}	$\frac{\partial M_\ell}{\partial H_\ell}$
a_{23}	$(H_\ell - H_{su}) \frac{\partial M_\ell}{\partial \alpha_p}$	a_{43}	$\frac{\partial M_\ell}{\partial \alpha_p}$

Table 3.4. Elements of matrix $\underline{\underline{C}}_p$

Element	Expression	Element	Expression
c_{11}	$\Delta t(H_{sp} - H_f)$	c_{31}	$-\Delta t$
c_{12}	$-\Delta t(H_f - H_{su})$	c_{32}	$-\Delta t$
c_{13}	$-\Delta t(H_v - H_{su})$	c_{33}	$-\Delta t$
c_{14}	$-\Delta t(H_v - H_{su})$	c_{34}	$-\Delta t$
c_{15}	$-\Delta t(H_v - H_{su})$	c_{41}	Δt
c_{21}	$\Delta t(H_f - H_{su})$	c_{42}	Δt
c_{22}	$\Delta t(H_f - H_{su})$	c_{43}	Δt
c_{23}	$\Delta t(H_f - H_{su})$	c_{44}	Δt
c_{24}	Δt		

Table 3.5. Elements of matrix $\underline{\underline{D}}_p$.

Element	Expression	Element	Expression
d_{11}	$-\Delta t(H_f - H_{su})$	d_{31}	$-\Delta t$
d_{12}	$\Delta t(H_g - H_{su})$	d_{32}	Δt
d_{21}	$\Delta t(H_f - H_{su})$	d_{41}	Δt
d_{22}	$-\Delta t(H_g - H_{su})$	d_{42}	$-\Delta t$

provisional set of pressurizer conditions is known at an advanced time. In each of four condition categories, two additional constraints are applied, and the number of unknowns is reduced to five. The constraints and unknowns for the four pressurizer condition categories are given in Table 3.6.

In condition 1, liquid is subcooled and vapor is superheated, both W_{ro} and $W_{f\ell}$ are set equal to zero, since no rainout and flashing are possible. In Condition 2, both phases are saturated, hence liquid and vapor enthalpies are functions of pressure, no longer independent variables. H_ℓ and H_v at time n+1 can be approximated respectively by the following equations:

$$H_\ell^{n+1} = H_f^{n+1} \approx H_f^n + \left(\frac{dH_f}{dp}\right)^n (P^{n+1} - P^n) \tag{3.34}$$

and

$$H_v^{n+1} = H_g^{n+1} \approx H_g^n + \left(\frac{dH_g}{dP}\right)^n (P^{n+1} - P^n) \tag{3.35}$$

In Condition 3, liquid is saturated and vapor is superheated. The constraints are that there is no rainout, and liquid enthalpy is related to pressure by Eq. (3.34). In Condition 4, liquid is subcooled and vapor is saturated, the constraints are that there is no flashing, and vapor enthalpy is related to pressure by Eq. (3.35). Finally, the surge mass flow rate, W_{su}, is left as an input quantity, or, in the present model, obtained by simultaneously solving the

Table 3.6. Constraints and unknowns in the two-fluid pressurizer model (these quantities are for evaluation at time t^{n+1}).

Condition	Liquid	Vapor	Constraints	Unknowns
1	Subcooled	Superheated	$w_{ro} = 0$ $W_{f\ell} = 0$	P, H_ℓ, H_v, α_p, W_{su}
2	Saturated	Saturated	$H_\ell = H_f(P)$ $H_v = H_g(P)$	P, α_p, $W_{f\ell}$, W_{ro}, W_{su}
3	Saturated	Superheated	$H_\ell = H_f(P)$ $W_{ro} = 0$	P, H_v, α_p, $W_{f\ell}$, W_{su}
4	Subcooled	Saturated	$H_v = H_g(P)$ $W_{f\ell} = 0$	P, H_ℓ, α_p, W_{ro}, W_{su}

pressurizer state equations, Eq. (3.33), and those of the RCS flow path components, Eq. (2.63).

3.4.3 Solution Technique

The computation procedure is an iterative one, because the advance time condition of the pressurizer is not known apriori. The procedure begins by assuming that the pressurizer is in Condition 1 at the beginning of each time step, that is, by letting $\underline{v}=\underline{0}$ in Eq. (3.33):

$$\underline{\underline{A}}_p^n \left(\underline{x}_1^{n+1} - \underline{x}^n\right)_p = \underline{\underline{C}}_p^n \, \underline{u}_{p,1}^{n+1} \tag{3.36}$$

where 1 indicates first guess. With four equations and five unknowns, Eq. (3.33) can be reduced to an equation in terms of P_1^{n+1} and $W_{su,1}^{n+1}$:

$$P_1^{n+1} - P^n = K_{21} W_{su,1}^{n+1} + K_{22} \tag{3.37}$$

where K_{21} and K_{22} are constants. Eq. (3.37) is solved simultaneously with Eq. (2.63) to yield solutions for P_1^{n+1} and $W_{su,1}^{n+1}$. The resulting liquid and vapor enthalpies after the first guess are compared with the new saturation enthalpies to identify the correct pressurizer condition. This identification procedure is illustrated in Figure 3.7.

The logic behinds the procedure is that if meta-stable condition exists in the liquid, or the vapor, or both, phase separation by flashing, or rainout, or both must take place.

If the results indicate conditions other than that of

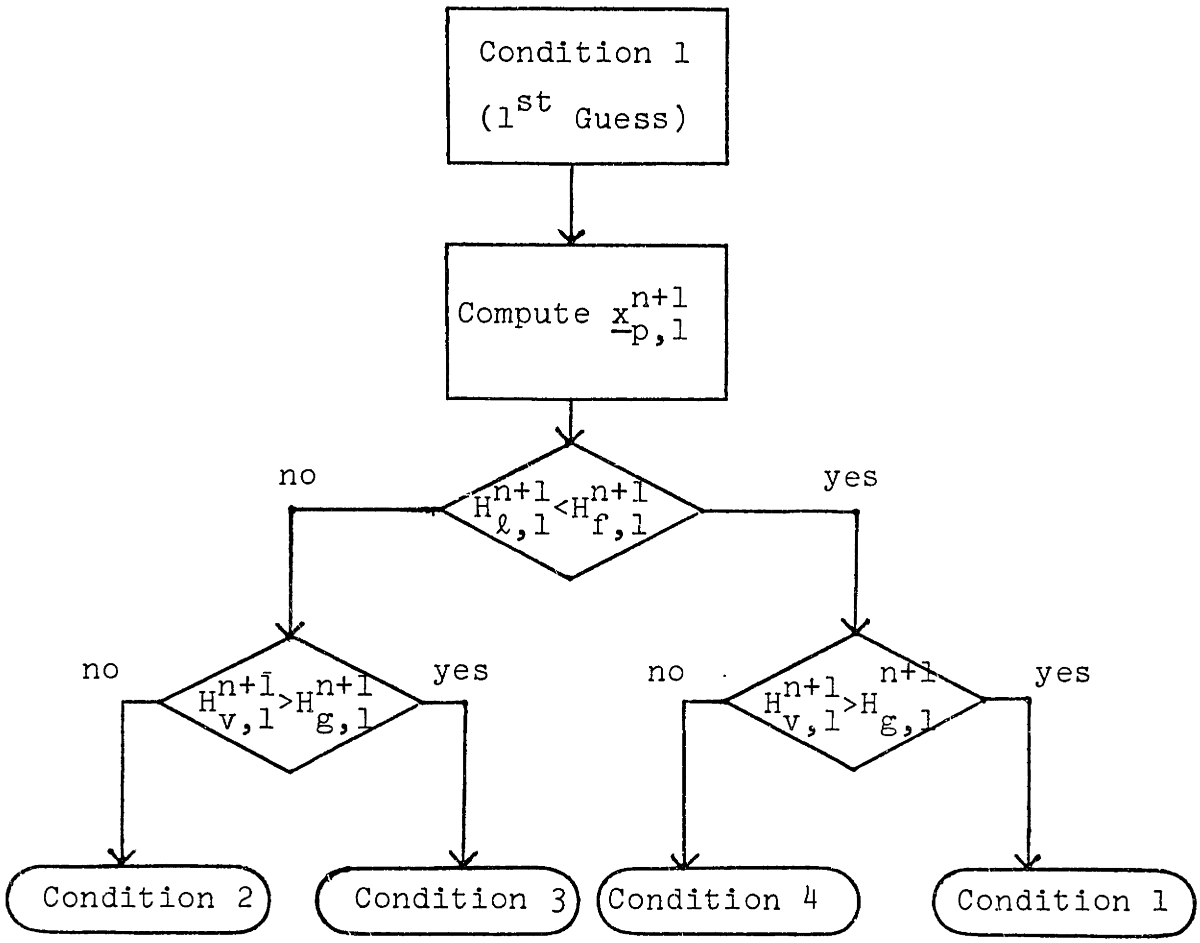


FIGURE 3.7 Pressurizer Condition Selection Flow Chart

the first category (Condition 1), correction is obtained by subtracting Eq. (3.36) from Eq. (3.33) to yield:

$$\underline{\underline{A}}_p^n \, (\underline{x} - \underline{x}_1)^{n+1} = \underline{\underline{C}}_p^n \, (\underline{u} - \underline{u}_1)_p^{n+1} + \underline{\underline{D}}_p^n \, \underline{v}^{n+1} \qquad (3.38)$$

where

$$(\underline{u} - \underline{u}_1)_p^{n+1} = [(W_{su} - W_{su,1})^{n+1}, 0, 0, 0, 0, 0]^T$$

Eq. (3.38) can be reduced to an equation in terms of P and W as:

$$P^{n+1} - P_1^{n+1} = K'_{21} \, (W_{su} - W_{su,1})^{n+1} + K'_{22} \qquad (3.39)$$

where K'_{21} and K'_{22} are constants. A simultaneous solution of Eqs. (3.39) and (2.63) yields P^{n+1} and W_{su}^{n+1}.

3.5 Model Validation

A computer program containing only the pressurizer algorithm was written. In the program, surge mass flow rate is provided throught input. Two sets of empirical data are chosen for benchmark validation of the pressurizer model. They are obtained from the small-scale pressurizer experiments conducted at MIT and an in-outsurge transient that occurred during the operation of Connecticut Yankee power plant.

3.5.1 MIT Pressurizer Experiment

The pressurizer experiment facility at MIT [K3] was built to study heat transfer mechanisms in a pressurizer during insurge and outsurge transients. The test section is a 1/10 scale model of a Westinghouse PWR pressurizer, and is made of stainless steel. The experimental data used here are those taken from the outsurge-after-insurge experiment No. IO1.

The experiment was conducted by injecting cold liquid into partially filled test section with both phases initially in thermal equilibrium, then draining the liquid in the pressurizer. In this experiment, mixing of cold and hot liquid was measured to be three inches (75 mm) in depth. Stable thermal stratification was maintained until all the injected cold liquid was drained out. During the outsurge phase, flashing of hot water occurred when the pressure decreased below the initial pressure. However, the flashing was less than that of simple outsurge of hot water. Flashing suppression was due to heat losses to the cold water by mixing and to the cold wall by conduction.

Since the two-fluid pressurizer model assumes perfect mixing of hot and cold liquids, flashing of hot liquid during outsurge can not occur. Here, provisions are made to allow flashing in hot liquid when pressure drops below the initial pressure at onset of insurge. To account for the "flashing suppression" effect, a factor of 0.2 is multiplied to reduce the amount of flashing in hot liquid during the outsurge phase of the transient.

The equivalent lengths for wall conduction are $L_c = L_w$ and $L_k = L_w/3$. A final adjustment of results uses a factor of 1.5 to multiply to the wall condensation mass flow rate (this indicates that this single-lump wall conduction model gives too little condensation, which is expected since it gives a too slow temperature response approximation to the time and space dependent model).

The surge mass flow history for this test is shown in Figure 3.8. The pressure calculation is shown in Figure 3.9.

The calculation speed is 30 ms per time step on the Honeywell DPS 8/70 M computer.

3.5.2 Connecticut Yankee In-Outsurge Transients

This event [R6] was initiated by a spurious SIS signal at Connecticut Yankee, which resulted in reactor scram. Outsurge occurred in the pressurizer due to contraction of the primary fluid. Then, initiation of ECCS caused an insurge in the pressurizer. There was no spray flow due to tripping of reactor coolant pumps.

The surge mass flow rate is calculated from the water level data recorded on strip chart (Figure 3.10). In this calculation, the equivalent lengths are $L_c = L_w/3$ and $L_k = L_w/12$. A wall-condensation multiplier of 1.65 and a flashing suppression factor of 0.5 are used. Figure 3.11 shows the pressure calculation results. As shown, agreement between results and data are excellent.

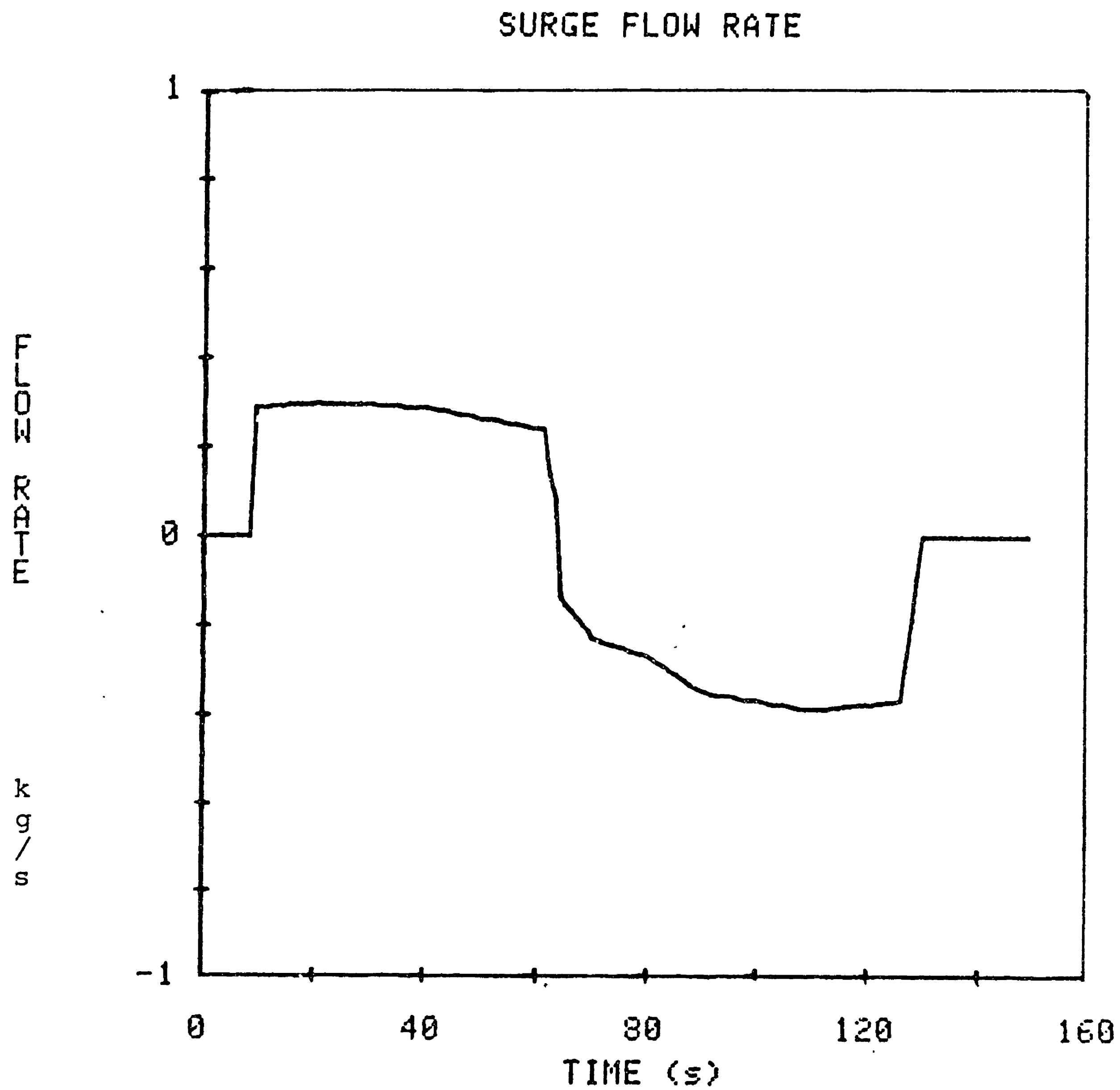


FIGURE 3.8 Input Surge Mass Flow Rate History for MIT Pressurizer Experiment IO1.

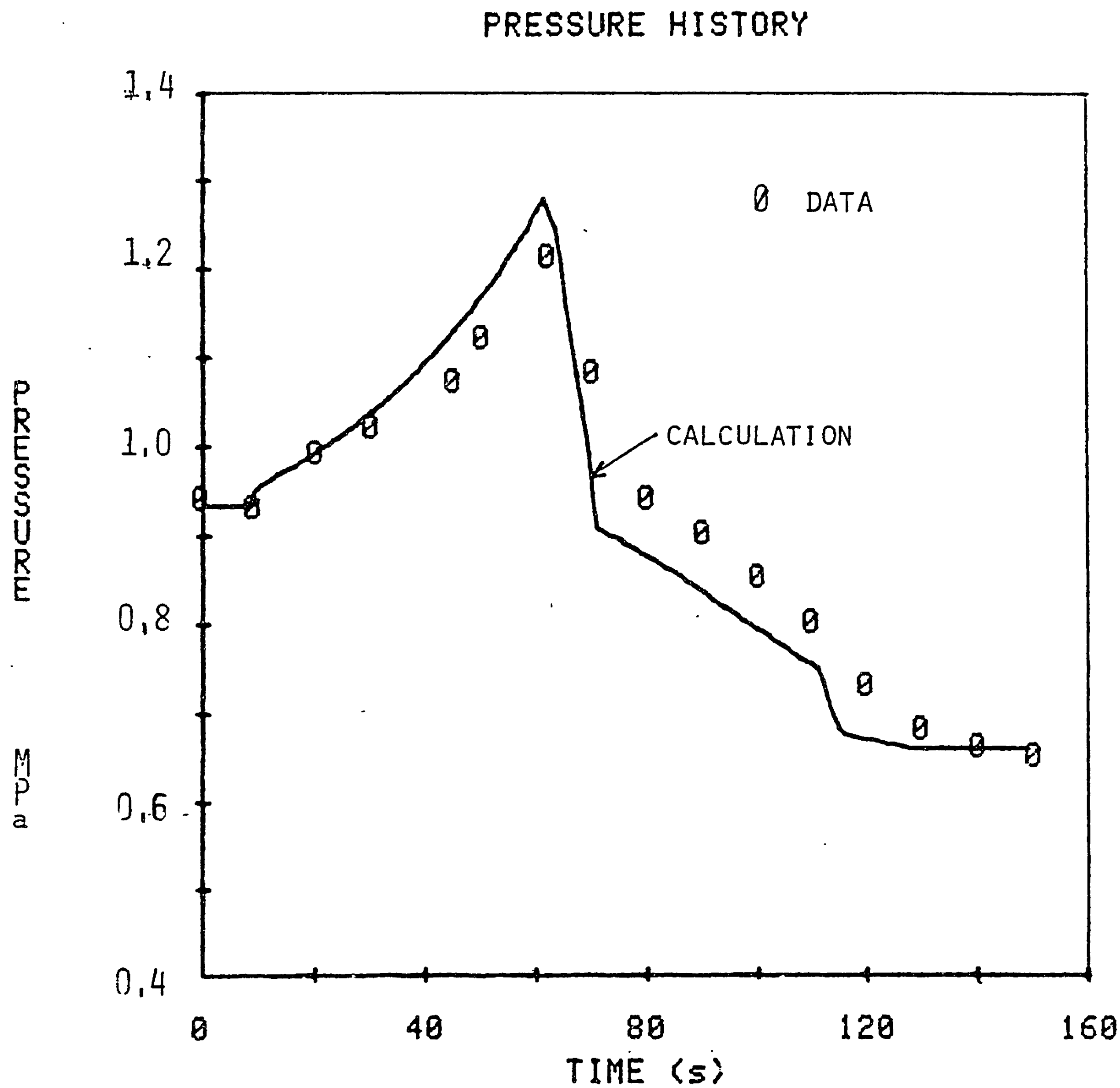


FIGURE 3.9 Pressure Calculation Results for MIT Pressurizer Experiment I01.

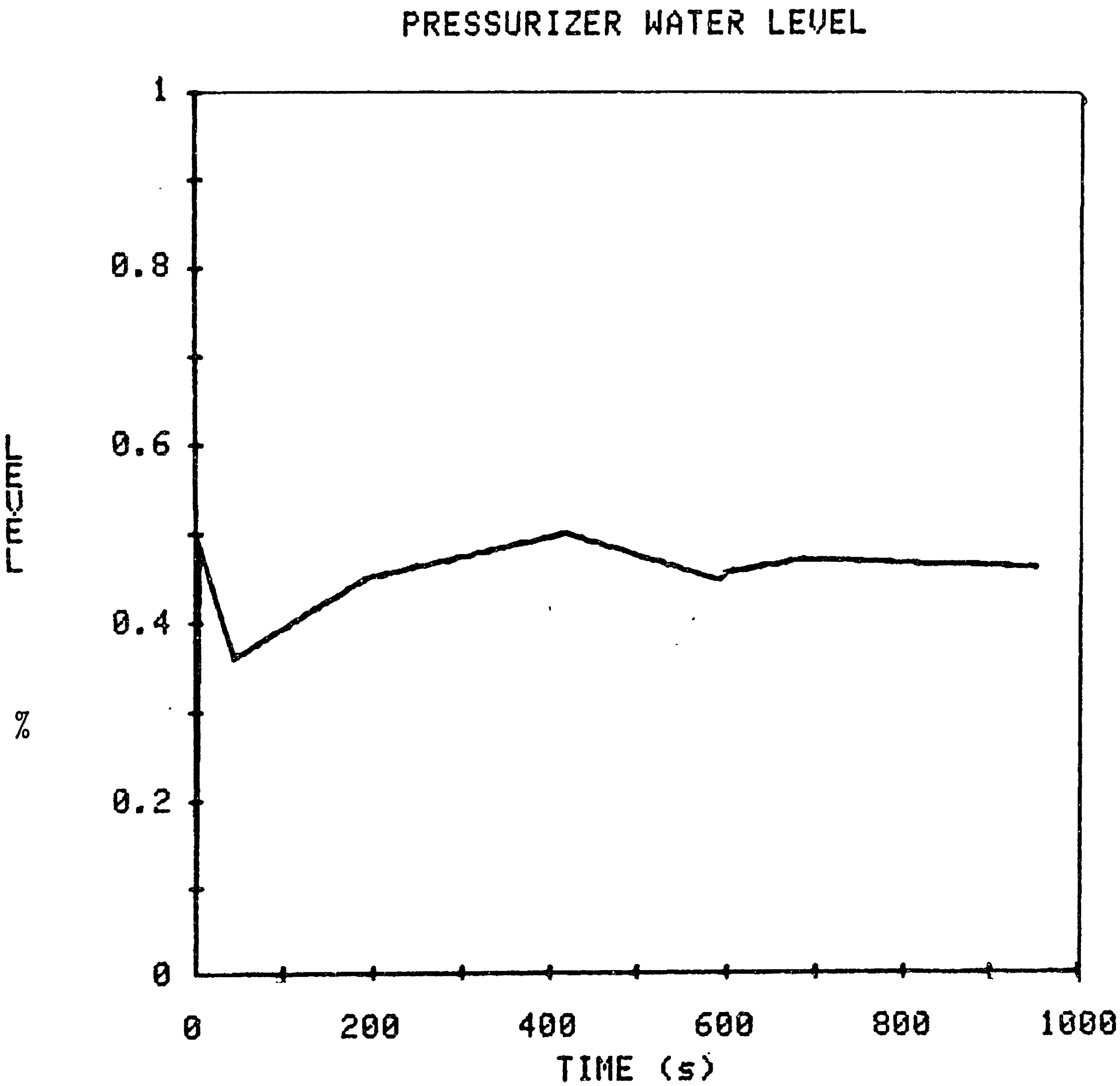


FIGURE 3.10 Input Water Level Histroy for Connecticut Yankee Transient.

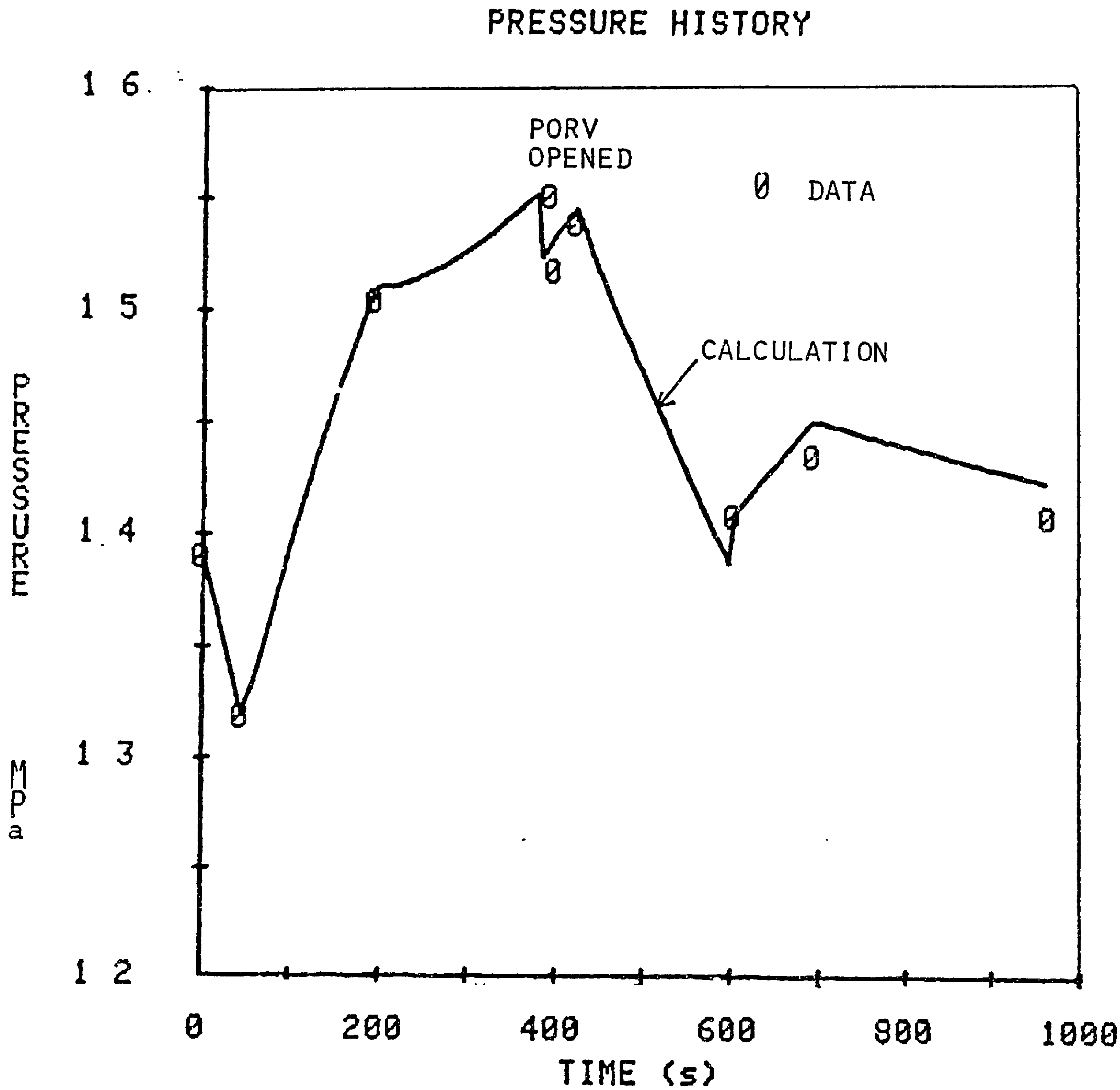


FIGURE 3.11 Pressure Calculation Results for Connecticut Yankee Transient.

3.6 Summary

A two-fluid pressurizer model is developed that can be readily adopted for on-line applications. The summary of this model are the following:

1) Properties are uniform within each phase;
2) No subcooled vapor and superheated liquid are allowed;
3) System equation are represented by a state space model with state variables P, H_ℓ, H_v, and α_p;
4) The pressurizer state equations are solved in two steps. The first calculation determines the condition in each phase assuming subcooled liquid and superheated vapor to exist in the final state. The second calculation is performed with adjustments for rainout or flashing or both if so required.
5) Wall condensation mass flow rate is determined by a single-lump conduction model; and
6) Flashing suppression in an in-outsurge is accounted for by including an isentropic flashing model and a "flashing suppression multiplier".

Validation tests show that the adopted wall condensation model gives too little condensation. By multiplying a factor around 1.5 to the calculated wall condensation mass flow rate, excellent pressure calculation results are obtained as compared with plant data.

It is seen that four empirically determined quantities (adjustable parameters) must be employed:

- the equivalent lengths for wall conduction (L_c and L_k);
- the wall-condensation multiplier; and
- the flashing suppression factor.

It is assumed that for a new application suitable values for these quantities can be determined by a trial-and-error process or, more formally, by a process of parameter adaptation.

CHAPTER 4

RCS MOMENTUM MODEL

This chapter discusses the development of a momentum model which is compatible with the energy and mass state-space equations derived in Chapter 2. As stated there, the properties are considered to be functions of the system pressure and the mixture enthalpies. In addition, the convection terms are described by an average mass flow rate in the each loop. These assumptions imply that the effects of sonic waves are neglected in that pressure or velocity disturbances are propagated at an infinite velocity (since the compressibility of fluid due to local pressure changes are not considered). A model that encompasses these characteristics is a single mass flowrate model described by Meyer [M5]. This model is felt to be appropriate for analyzing intermediate-speed transients. The loop momentum equation developed in this research is similar to previous developments with a few additional approximations. The constitutive relations used in the pump-rotor speed equation are based on the work of Wong [W3].

4.1 Flow Path Momentum Equation

4.1.1 Equation Derivation

This section summarizes the derivation of a momentum equation based on integration over a flow path with uniform mass flow rate. This flow path equation can be applied to each loop outside of the rector vessel and it can be applied to flow through the reactor vessel. The conservation of momentum equation for a point in a flow path is written as [P2] [S1]:

$$\frac{\partial}{\partial t} \frac{W}{A} + \frac{1}{A} \frac{\partial}{\partial s} \left(\frac{v'W^2}{A}\right) = - \frac{\partial P}{\partial s} - \frac{fW|W|}{2\rho D_h A^2} - \rho g \sin \theta \qquad (4.1)$$

where

$$v' = \frac{(1 - X)^2}{\rho_f (1 - \alpha)} + \frac{X^2}{\rho_g \alpha}$$

Eq. (4.1) can be integrated over the flow path to give:

$$\frac{\partial}{\partial t} \int \frac{W}{A} ds + \int \frac{v'W^2}{A^2} ds = -\Delta P - \int \frac{fW|W|}{2\rho D_h A^2} ds$$

$$- \int \rho g \sin \theta \, ds \qquad (4.2)$$

The single mass flowrate approximation is applied here to simplify Eq. (4.2). That is, the equation can be evaluated in terms of the single mass flowrate, <W>, or:

$$\int \frac{ds}{A} \frac{d\langle W\rangle}{dt} = -\Delta P - \int \frac{f\, ds}{2\rho D_h A^2} \langle W\rangle \, |\langle W\rangle|$$

$$- \langle W\rangle^2 \int \frac{v'ds}{2} - \int \rho g \sin \theta \, ds \qquad (4.3)$$

The integrals in Eq. (4.3) become summations of the average quantities. For example, for an RCS loop outside the reactor vessel:

$$\int \frac{dS}{A} = \left(\frac{L}{A}\right)_{HL} + \left(\frac{L}{A}\right)_{SG} + \left(\frac{L}{A}\right)_{CL}$$

$$= \sum_i \left(\frac{L}{A}\right)_i \qquad (4.4)$$

A similar application to flow through the reactor vessel (or core), W_C, supplies the final flow path momentum equation. Therefore, the momentum equation integrated over the closed path including loop j and the reactor vessel is the following:

$$\sum_i \left(\frac{L}{A}\right)_i \frac{d\langle W\rangle_j}{dt} + \left(\frac{L}{A}\right)_{RV} \frac{dW_C}{dt} = \Delta P_{pump,j} - \sum_i F_i(\langle W\rangle_j) - F_{RV}(W_{RV}) + S_j \qquad (4.5)$$

where

$$F_i = \left(\frac{fL}{2\rho D_n A}\right)_i \langle W\rangle_j \left|\langle W\rangle\right|_j + K_i \frac{\langle W\rangle_j^2}{A_i^2} \qquad (4.6)$$

and

$$S_j = \oint \rho \underline{g} \cdot d\underline{s} \qquad (4.7)$$

where F_i represents the loop friction losses and form drag in Component i, S_i is the hydrostatic head for Loop j, and ΔP is the head delivered by the loop j pump. The expressions used to evaluate these terms are discussed in the following sections.

4.1.2 Pressure Losses from Friction and Form Drag

It has been found satisfactory (and convenient) to solve the momentum equation independent of the energy and mass equations. The momentum mesh scheme can be different and finer than that of energy and mass. The flow path of a PWR RCS loop is divided into ten sections for the momentum model (Figure 4.1). The loop is divided into the core, reactor vessel exit plenum and R/V exit nozzle, hot leg, steam generator inlet nozzle and plenum, steam generator tube, steam generator exit plenum and nozzle, cold leg, reactor vessel inlet nozzle and plenum, down-comer, and reactor vessel lower plenum. Component numbers are assigned as shown on the figure.

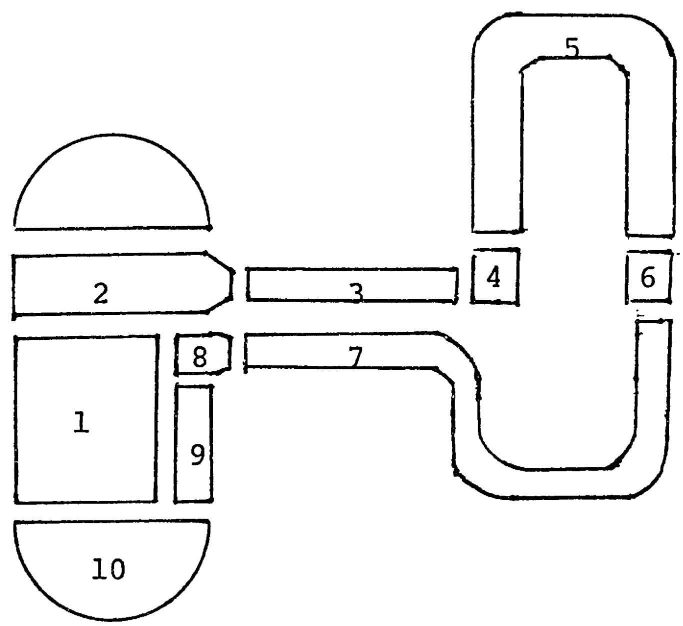
The friction pressure loss in each component is evaluated by the following equation:

$$\Delta P_{fr} = \frac{f\langle W\rangle \left|\langle W\rangle\right| L}{2 \rho D_h A^2} \tag{4.8}$$

Where the single-phase friction factor is calculated by the Colebrook-White equation, and where the Martinelli-Nelson-Jones two-phase multiplier is used for two-phase flow (App. B).

For calculating pressure changes due to expansion, contraction, and bends the following equation is used:

$$\Delta P_{loss} = \frac{K\langle W\rangle \left|\langle W\rangle\right|}{2 \rho A^2} \tag{4.9}$$



1. Core
2. R/V Upper Plenum
3. Hot-Leg
4. S/G Inlet Plenum
5. S/G Tubes
6. S/G Exit Plenum
7. Cold-Leg
8. R/V Inlet Plenum
9. Downcomer
10. R/V Lower Plenum

FIGURE 4.1 Momentum Mesh Scheme

Where the loss coefficient is treated as an input constant (it is a parameter that can be changed to fit available steady-state flow pressure-drop data from the plant being modeled).

The spacers supporting the fuel rods in the core give a loss that can be calculated by an equation given in Rust [R7] [W3], as:

$$\Delta P_{sp} = \frac{C_v \varepsilon^2 \langle W \rangle \left| \langle W \rangle \right|}{2 \rho A^2} \tag{4.10}$$

where

ε = ratio of projected grid cross section to undisturbed flow cross section;

and

C_v = modified drag coefficient
$54.91 \left| \langle W \rangle \right|^{-0.0245}$

4.1.2 Hydrostatic Pressure Head

The hydrostatic force is a body force due to gravity. When the RCS is operated in a forced convection mode, the hydrostatic force is negligible compared to other pressure forces in Eq. (4.5). However, after the pumps have coasted down, it becomes the driving head in natural circulation. The requirement for the hydrostatic head being positive is that the thermal center of the steam generator be higher than that of the core.

For a closed primary recirculation loop, the integral in Eq. (4.7) is the sum of five integrals as the following:

$$\int \rho \underline{g} \cdot d\underline{s} = -\int_{C} \rho g ds - \int_{UP} \rho g ds - \int_{SG} \rho g ds + \int_{CL} \rho g ds + \int_{LP} \rho g ds \qquad (4.11)$$

The hydrostatic forces in the core, R/V upper and lower plenum are approximated by using the average density, as given by:

$$\int_0^L \rho \underline{g} \cdot d\underline{s} \approx \frac{1}{2} (\rho_{in} + \rho_{out}) gZ = \rho_m gZ \qquad (4.12)$$

where Z is the vertical height of the component. The equation is exact if the density profile is linear. In the core region, it can be assumed that the heat flux is symmetric so that Eq. (4.12) is a valid approximation.

Since there is a U-bend in the cold leg just before the pump, a "cold trap" can be caused by a sudden cooling of the primary coolant in the steam generator; hence, cold fluid can be trapped in the U-bend blocking the flow. The effects of "cold trap" can be approximated by:

$$\int_{CL} \rho \underline{g} \cdot d\underline{s} = (\rho_m)_{CL} gZ_{CL} - (\rho_m)_{SG} gZ_{CL} \qquad (4.13)$$

where Z_{CL} = the height of U-bend.

In steam generator tubes, the fluid density assumes an exponential decay profile. In this case, the hydrostatic force is determined by evaluating the integral analytically using the log-mean temperature profile. The solution, derived in Appendix D, is represented in terms of the thermal center of the steam generator Z_{SG} as:

$$\int_{SG} \rho \underline{g} \cdot d\underline{s} = (\rho_{HL} - \rho_{SG})\, g Z_{SG} \tag{4.14}$$

Substituting Eqs.(4.12), (4.13), and (4.14) into Eq. (4.11), obtain an expression for the overall hydrostatic head in the loop:

$$\begin{aligned}\int \rho \underline{g} \cdot d\underline{s} = g \{ & -2(\rho_m)_C Z_C - (\rho_m)_{UP} Z_{UP} - (\rho_m)_{HL}(Z_0 - Z_{UP} - 2Z_C) \\ & + [(\rho_m)_{CL} - (\rho_m)_{SG}] Z_{CL} + (\rho_m)_{SG}\,(Z_0 - Z_{EP} - 2Z_C) \\ & + \int_{SG} (\rho_m)_{SG}\, \underline{g} \cdot d\underline{s} + (\rho_m)_{LP}\,(Z_{UP} + 2Z_C) \}\end{aligned} \tag{4.15}$$

where

Z_C = distance from core inlet to core mid-plane,

Z_0 = elevation difference between core inlet and steam generator tube sheet, and

Z_{UP} = elevation difference between core exit and reactor vessel exit nozzle.

4.1.4 Pump Head

The pumping head delivered by a Reactor Coolant Pump is represented by the characteristic head-capacity relations given in Appendix G. The head-capacity relation is a function of mass flow rate (or volumetric flow rate), pump rotor speed, and fluid density ratios:

$$\Delta P_{pump} = F(W/W_R,\ \omega/\omega_R,\ \rho/\rho_R) \tag{4.16}$$

As indicated by the functional relationship, we need an equation to determine the pump rotor speed as a function of time. The formulation of the pump-rotor-speed differential equation is obtained by applying the conservation of angular momentum to the pump rotor and associated rotating components [R7] [W3]:

$$I \frac{d\omega}{dt} = T_e - T_B - T_W \tag{4.17}$$

where

ω = pump rotor speed;

I = rotational moment of inertia;

T_e = electric torque;

T_B = break-horse-power torque; and

T_W = windage and bearing loss torque.

The electric torque is the torque provided by the electric motor which is operated on a constant current. When the power is cut off, the electric torque decays exponentially to zero according to the equation [W3]:

$$(T_e/T_{e_R}) = (\omega/\omega_R) \exp(-2t/\tau_e) \tag{4.18}$$

where T_{eR} and ω_R are rated electric torque and rotor speed respectively, and τ_e is a time decay constant (here set equal to 0.1 μs, essentially instantaneous loss of applied torque).

The brake-horse-power torque is obtained from the pump torque-capacity characteristic performance relations given in Appendix G. It is the torque imparted by the fluid to the pump impeller at a given flow and rotor speed.

The windage and bearing loss torque is due to the named friction losses. The expression for the windage and bearing torque is taken to be the following [W3]:

$$
\begin{aligned}
(T_W/T_{WR}) &\simeq (\omega/\omega_R)^2 \quad ; \quad \omega > 0.19\ \omega_R \\
&= 0.035 \quad ; \quad 0 < \omega < 0.19\ \omega_R \qquad (4.19) \\
&= 0.1 \quad ; \quad \omega = 0
\end{aligned}
$$

where T_{WR} is the rated windage and bearing torque.

4.2 State-Space Momentum Representation

4.2.1 Differential Equations

In the previous section, we have shown that for each primary loop, there are two state equations, Eqs. (4.5) and (4.17), corresponding to the two state variables: $\langle W \rangle$ and ω. The loop momentum equation, Eq. (4.5), can be represented in terms of the function F as:

$$\sum_i \left(\frac{L}{A}\right)_i \frac{d\langle W \rangle_j}{dt} + \left(\frac{L}{A}\right)_{RV} \frac{dW_C}{dt} = F(\langle W \rangle_j, \omega_j, W_C) \qquad (4.20)$$

In addition, the pump rotor speed equation, Eq. (4.17), can be represented in terms of a function G as:

$$\frac{d\omega_j}{dt} = G(\langle W \rangle_j, \omega_j) \qquad (4.21)$$

where Matrix $\underline{\underline{B}}^+$ has the following structure:

$$
\underline{\underline{B}}^+ = \left[\begin{array}{cc|cc|c|c}
b_{11}(1) & b_{12}(1) & 0 & 0 & \cdots & b_{13}(1) \\
b_{21}(1) & b_{22}(1) & 0 & 0 & & 0 \\
\hline
0 & 0 & b_{11}(2) & b_{12}(2) & & b_{13}(2) \\
0 & 0 & b_{21}(2) & b_{22}(2) & & 0 \\
\hline
& \vdots & & & \ddots & \vdots \\
\hline
-1 & 0 & -1 & 0 & & 1
\end{array}\right]
$$

and where the elements of $\underline{\underline{B}}^+$ are given in Table 4.1.

By including W_C as a state variable and Eq. (4.22) as a state equation, we obtain the coefficient matrix, $\underline{\underline{B}}^+$, which consists of a diagonal block matrix augmented by one row and one column. The calculation steps required to invert $\underline{\underline{B}}^+$ are much simpler than inverting the matrix obtained by substituting Eq. (4.22) into (4.20). Although Matrix $\underline{\underline{B}}^+$ has a zero eigenvalue corresponding to W_C, error propagation would not occur if loop flows are determined first and Eq. (4.22) is used to determine W_C^{n+1}; i.e.:

$$W_C^{n+1} = \sum_j \langle W \rangle_j^{n+1} \tag{4.27}$$

4.3 Model Validation

The momentum model is an independent model if the fluid densities are known. Therefore, it is possible to perform an

The mass flow rate through the core is obtained by the following equation:

$$W_C = \sum_{j=1}^{m} \langle W \rangle_j \quad ; \quad m = \text{no. of loops} \tag{4.22}$$

For a multiple loop RCS, the following system of differential equations is obtained as the state space model for conservation of momentum:

$$\underline{\underline{B}}\dot{\underline{y}} = \underline{h}(\underline{y}) \tag{4.23}$$

where

$$\underline{y} = [W_1, \omega_1, \ldots, W_m, \omega_m, W_C]^T$$

$$\underline{h} = [F_1, G_1, \ldots, F_m, G_m, 0]^T$$

Matrix $\underline{\underline{B}}$ is a square matrix with dimension (2m+1) by (2m+1).

4.2.2 Finite-Difference Equations

Use a semi-implicit finite-difference representation of Eq. (4.23) as follows:

$$\underline{\underline{B}}^n \Delta\underline{y}^{n+1} = \Delta t \, [\underline{h}(\underline{y}^n) + \underline{\underline{J}}^n \Delta\underline{y}^n] \tag{4.24}$$

where $\underline{\underline{J}}$ is the Jacobian of $\underline{h}$, i.e., $(\partial\underline{h}/\partial\underline{y})$. Re-arrange Eq. (4.24) to give:

$$[\underline{\underline{B}} - \Delta t \, \underline{\underline{J}}]^n \, \Delta\underline{y}^{n+1} = \Delta t \, \underline{h}^n \tag{4.25}$$

or

$$(\underline{\underline{B}}^+) \, \Delta\underline{y}^{n+1} = \Delta t \, \underline{h}^n \tag{4.26}$$

Table 4.1. Elements of matrix $\underline{\underline{B}}^T$.

Element	Expression
$b_{11}(j)$	$\sum_i (\frac{L}{A})_i - \Delta t \ (\frac{\partial F}{\partial W})_j^n$
$b_{12}(j)$	$-\Delta t \ (\frac{\partial F}{\partial \omega})_j^n$
$b_{21}(j)$	$-\Delta t \ (\frac{\partial G}{\partial W})_j^n$
$b_{22}(j)$	$1 - \Delta t \ (\frac{\partial G}{\partial \omega})_j^n$
$b_{13}(j)$	$(\frac{L}{A})_{RV} - \Delta t \ (\frac{\partial F}{\partial W_C})_j^n$

independent validation of the model with respect to a pump coastdown transient. The first benchmark test is a total loss-of-flow test conducted at Maine Yankee. The second validation test is based on one-out-of-three pump trip flow calculation given in the Maine Yankee FSAR [F1]. Maine Yankee is a three-loop Combustion Engineering PWR unit (Figure 4.2).

4.3.1 Maine Yankee Total Loss-Of-Flow Transient

The test data are obtained from the Maine Yankee Start-Up Report [Y1]. The computer input file for this case is given in Appendix H. The results are shown in Figure 4.3. The results from this model follow the data closely for the most of the transient. Toward the end where the data level off while the calculation shows a continuing decrease. Since it is unlikely that natural circulation would establish at such high mass flow rate, the discrepancy is thought due to sensor saturation.

4.3.2 Maine Yankee One-Pump Trip Transient

The calculation of the present model is compared with calculated results obtained from the Maine Yankee FSAR as shown in Figure 4.4. The present model yields results slightly less than given in the FSAR. The calculation is carried out to one hundred seconds as shown in Figure 4.5. Although there are no data to compare with, the calculation shows that flow reversal occurs at 46 seconds into the

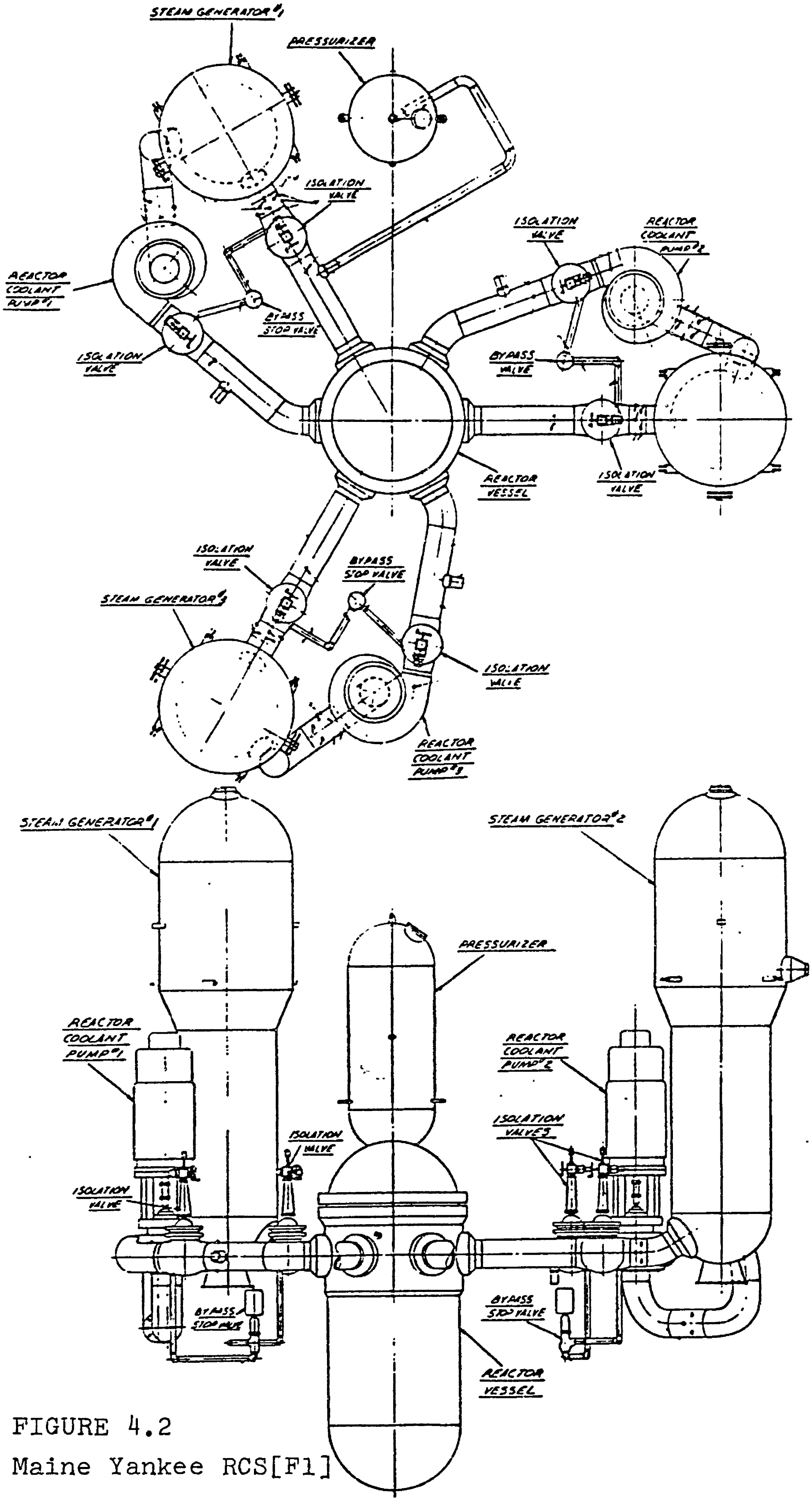


FIGURE 4.2
Maine Yankee RCS[F1]

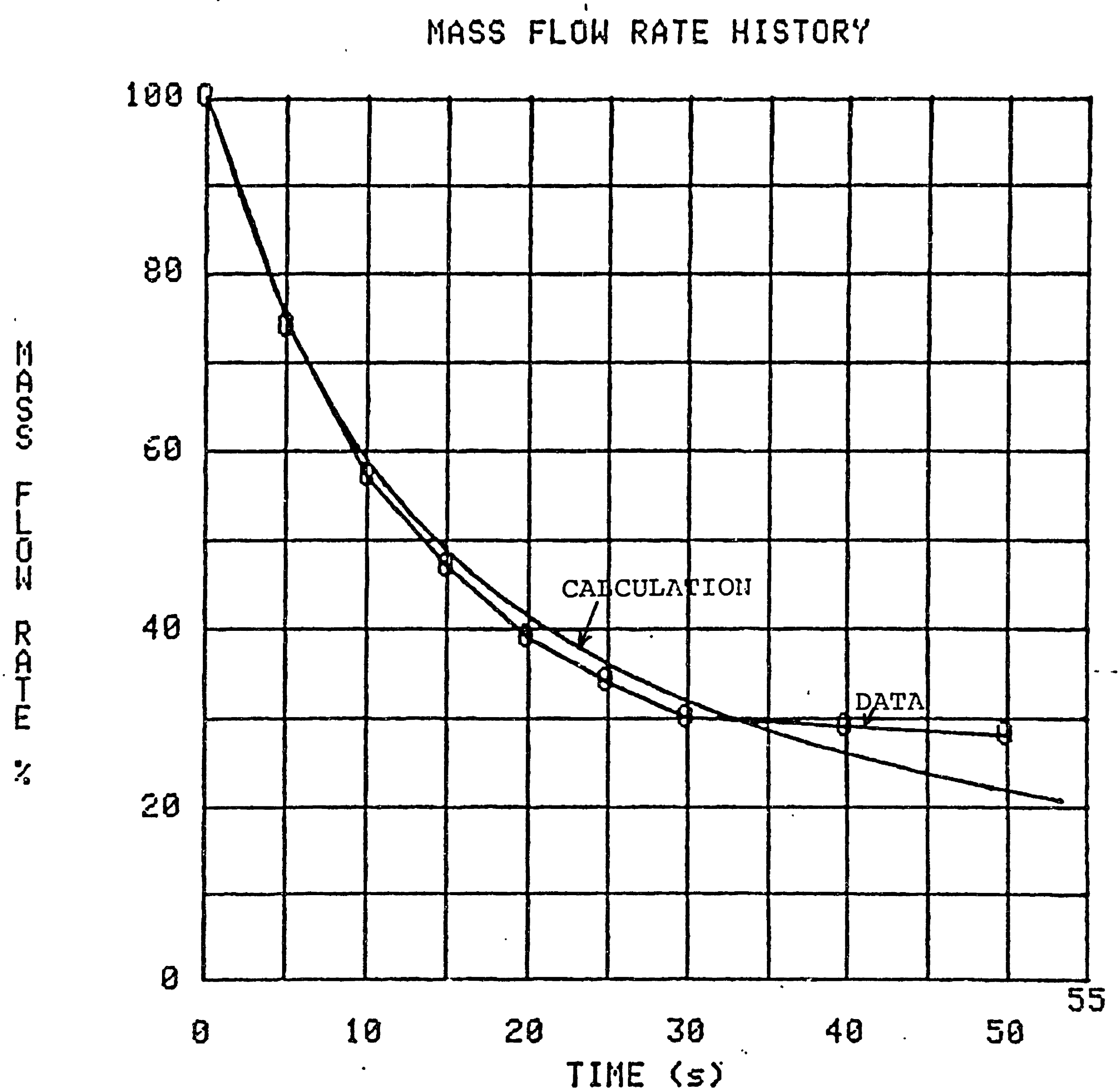
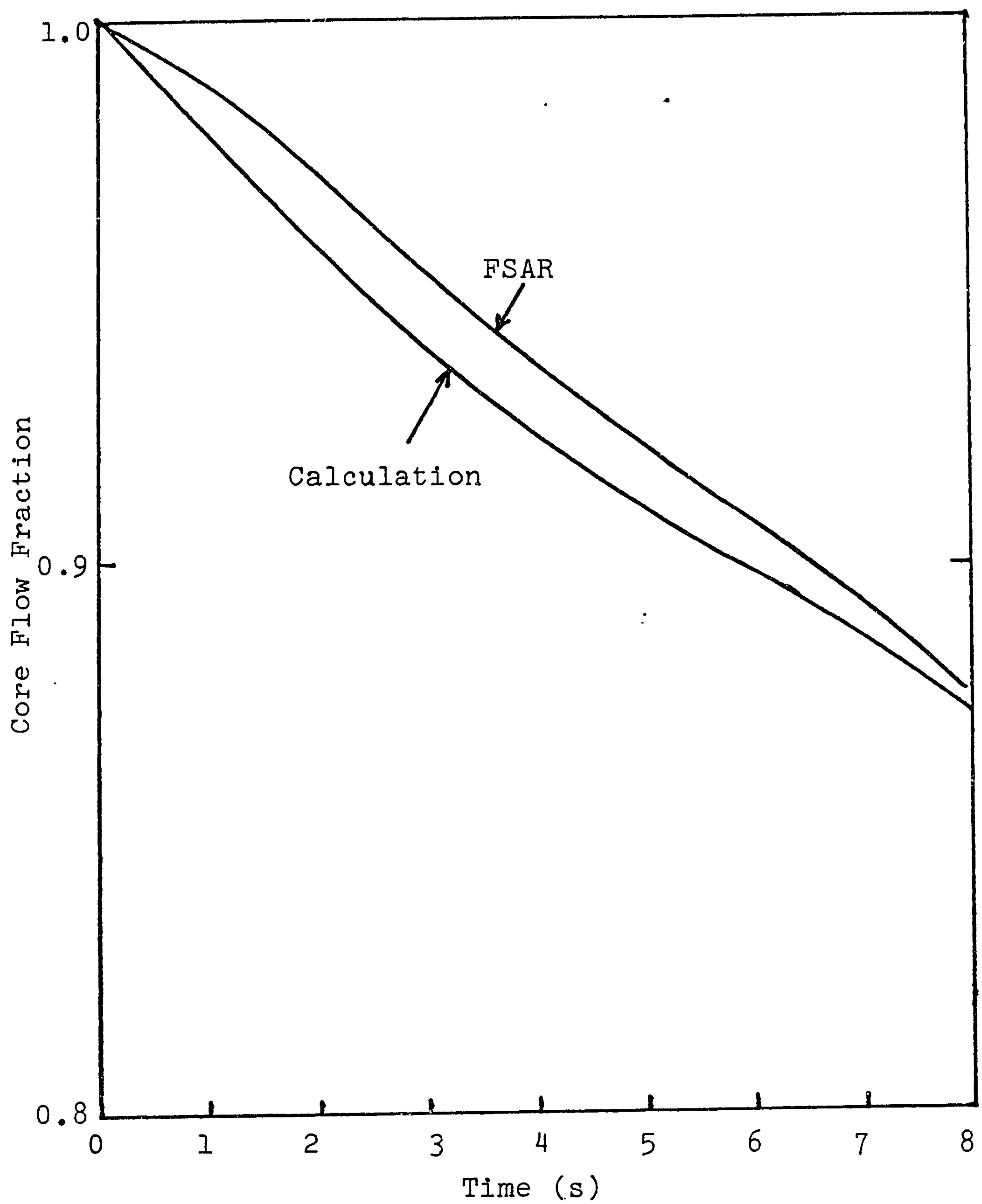


FIGURE 4.3 Maine Yankee Total-Loss-of-Flow Transient Mass Flow Rate History.



FIUGRE 4.4 Comparison of Calculated Maine Yankee One-Pump-Trip Core Flow Fraction with FSAR Results.

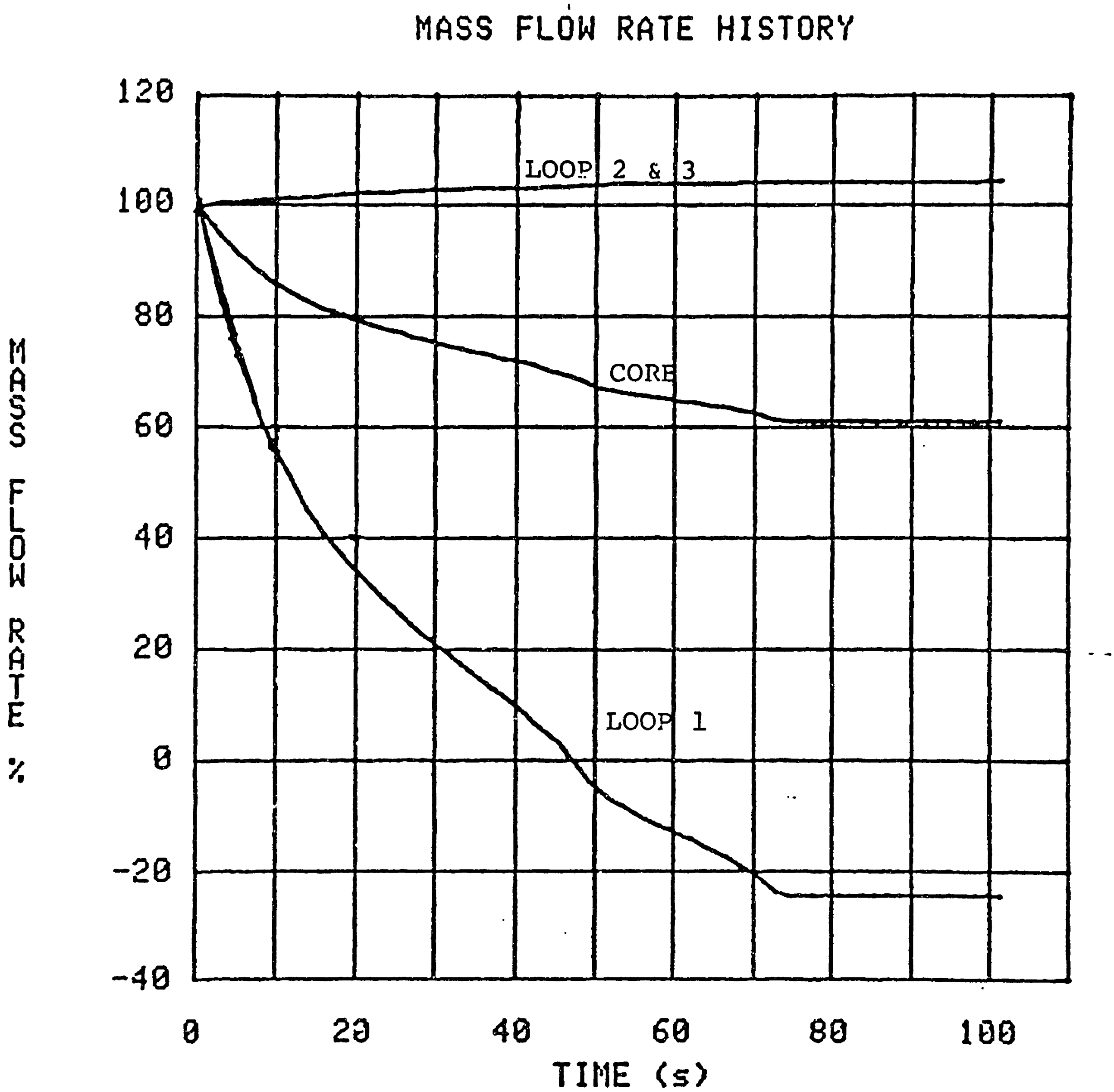


FIGURE 4.5 Maine Yankee One-Pump-Trip Mass Flow Rate History.

transient in the loop with failed pump. A steady reverse flow established at 73 seconds, when the rotor speed becomes zero and the anti-reverse rotation command is executed. The flows in the other two loops increase by about five percent. The net effect on the mass flow rate through the core is that about sixty percent of full flow is achieved with two-pump operation.

4.4 Summary

A multiple-loop momentum model is developed to calculate the average mass flow rate for each loop. The model consists of the loop integral momentum equations and the pump rotor speed differential equations. In deriving the momentum equation, we have assumed that a single mass flowrate can be used to approximate the flow in each loop. Therefore, only this single mass flowrate is used to evaluate the friction and form pressure losses. A constitutive equation is included to determine the hydrostatic head and the steam generator thermal center. Complete pump characteristic relations describing the pump head and torque are also included. The model is validated against plant data and FSAR calculation of Maine Yankee pump coastdown transients. The results are satisfactory. The model is also shown to work satisfactorily from a calculational standpoint in reverse flow.

CHAPTER 5

DIGITAL COMPUTER SIMULATION MODEL

In this chapter, the structure of a computer simulation model based on the numerical models developed in the previous chapters is presented. Included are discussions of the method for initial steady-state condition determination and the computation procedure for transient calculations.

5.1 Initial Steady-State Solutions

For the pressurizer, the initial pressure and water level are given. The liquid and vapor enthalpies are corresponding saturation values. All the boundary conditions, such as the spray mass flow rate and the heater power, are set equal to zero.

For the flow path components, the initial thermal power and the cold-leg temperature are given. The steam generator exit temperature and the reactor vessel lower plenum temperature are taken equal to that of cold leg. The system initially is in symmetric operation. That is, the system variables and boundary conditions (secondary pressures) are identical in all loops. After the initial mass flow rate is obtained from the momentum and pump speed equations, the hot leg temperature is determined by the following equation:

$$H_{HL} = H_{CL} + Q_{th}/W_C \tag{5.1}$$

The core exit enthalpy is then set equal to the hot leg enthalpy.

The initial average fuel temperature is obtained by solving the steady-state fuel pin conduction equation using the false-position iteration [H2]. That is, the iterative formula for the steady-state conduction equation, $f(T_F) = 0$, is given by:

$$z_{k+1} = z_k - \frac{z_k - z_o}{f_k - f_o} \quad (k \geq 1) \tag{5.2}$$

where z_0 is an initial approximation to the desired root, which is taken to be the hot leg temperature.

The initial overall heat transfer coefficient in the steam generator is obtained on the basis of a specified initial secondary pressure by determining the initial fouling factor:

$$r_{ff} = \frac{A_o \Delta T_{\ell m}}{Q_{SG}} - \frac{A_o}{A_i h_i} - \frac{r_o \ell n(r_o/r_i)}{k_t} - \frac{1}{h_o} \tag{5.3}$$

Eq. (5.3) is based on solving Eq. (2.51) for r_{ff}.

In the initial steady-state calculations, the hydrostatic pressure term is neglected in the momentum equation. All loops are assumed identical with the same mass flow rate and pump rotor speed. The initial loop mass flow rate is determined by solving the steady-state momentum equation using a Newton-Raphson iteration [H2], for a given initial pump speed. The method is as follows. For the steady-state momentum equation $f(W) = 0$, the iterative formula is given by:

$$W_{k+1} = W_k + \left(\frac{\partial f}{\partial W}\right)_k f_k \tag{5.4}$$

The initial guess is the rated mass flow rate.

5.2 Transient Calculation Procedure

After the initial states are determined, the transient calculation steps are illustrated by the flow chart given in Figure 5.1. The computation procedure consists of three basic algorithms: the pressurizer, the flow path component mass energy, and the momentum algorithms. The computation procedure is as follows. For each time step, the boundary conditions--the nuclear power and the secondary pressures, are taken from user specified input. The mass flow rate and pump rotor speed are solved first. Then the pressurizer and flow path component state variables are determined simultaneously. Next the properties are updated. Then the hydrostatic head based on the new fluid densities is determined. At the end of each time step, a new time-step size is calculated. It is equal to the fluid transport time through the core, according to the equation:

$$\Delta t = \frac{(\rho V)_C}{W_C} \tag{5.5}$$

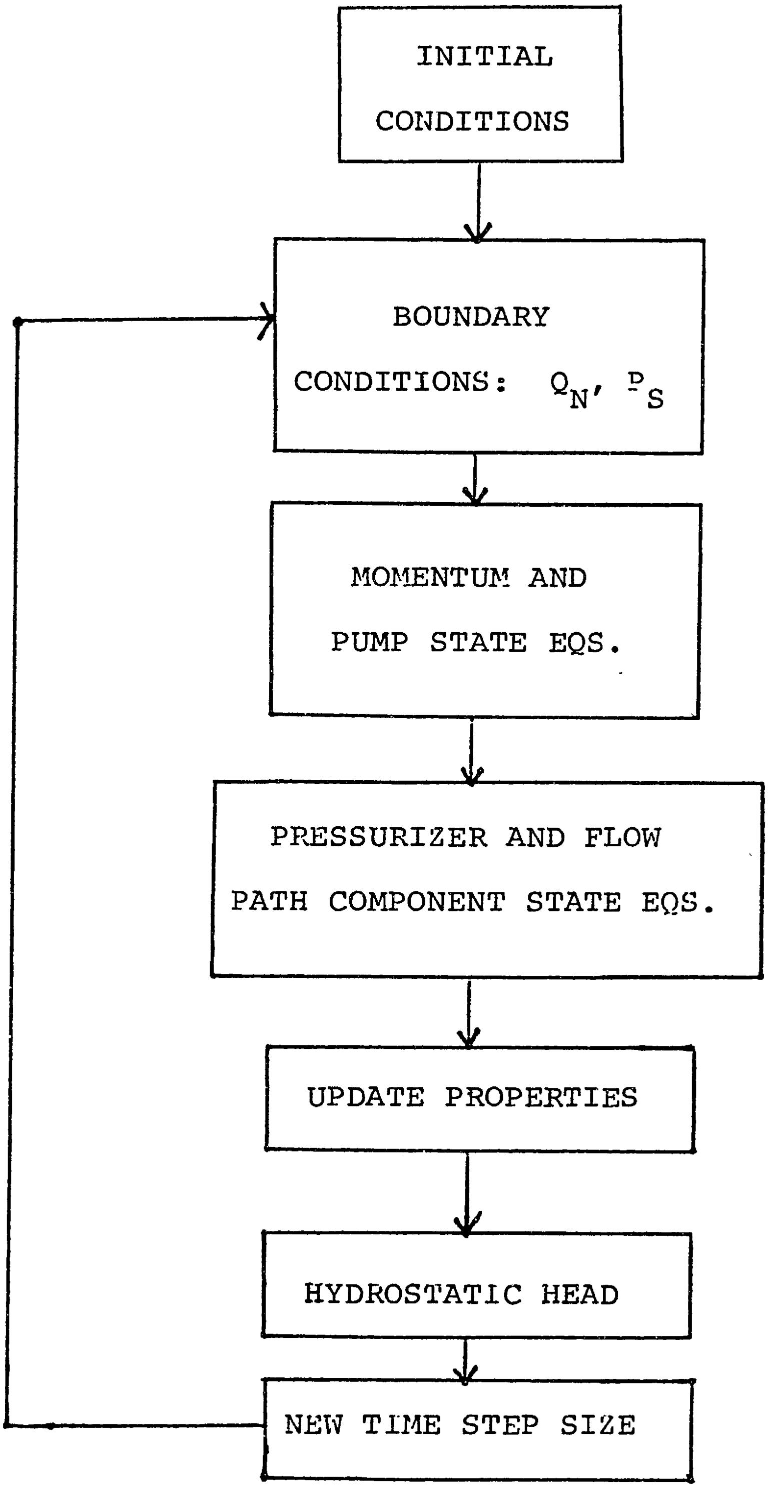


FIGURE 5.1 Computational Flow Chart

The steps are repeated for the next time step, until the end of simulation.

The computer program is named the SPK code (System and Pressurizer Kode). The computer program is written in the FORTRAN language. The input description is given in Appendix I. The code listing is provided in Appendix J.

CHAPTER 6

NATURAL CIRCULATION VALIDATION

Natural circulation is an efficient, passive heat transfer mechanism for long-term decay-heat removal operation in a Pressurized Water Reactor. In this chapter, simulations of natural circulation experiments conducted at Maine Yankee, San Onofre Unit 2, and the Loss-of-Fluid Test (LOFT) facility are discussed. These experiments range from single-phase natural circulation to two-phase natural circulation initiated by a small break in the primary system. In all cases, it is assumed that a symmetrical condition is maintained throughout the transient, hence a single-loop RCS representation is used. The system parameters and initial conditions for these tests are given in Appendix H.

6.1 Maine Yankee Natural Circulation Test

The test at Maine Yankee [Y1] was to demonstrate the plant's decay-heat removal capability in a loss-of-power event. The reactor was operated at 35% power prior to the initiation of the test. The test was initiated by simultaneously tripping all three Reactor Coolant Pumps. The reactor was tripped automatically on a low flow signal 2.2 seconds after the pump trip. The indication of natural circulation was verified by measuring the temperature rise across the vessel (or the temperature difference between the hot and cold legs). The temperature rise across the vessel indicates the extent that heat generated in the core is transfered to the steam generator by natural circulation.

The simulation is a continuation of the Maine Yankee total-loss-of-flow transient discussed in Chapter 5. The steam generator secondary pressure was 4.96 MPa initially, then increased to 5.2 MPa at 10 seconds after the pump trip. It was assumed in the calculation that the steam generator pressure was maintained at 5.2 MPa for the duration of the transient.

The calculated vessel temperature rise in percent of rated temperature rise is given in Figure 6.1. The calculation shows a lower temperature rise than the data for the first 400 seconds, then converges closely to the data. The oscillation in the data may be due to pressure variation in the steam generator (the steam generator pressure is given for the first fifty seconds only).

The calculated mass flow rate history is shown in Figure 6.2. The calculated natural circulation mass flow rate is about 3% of full flow.

6.2 San Onofre Unit 2 Low-Power Natural Circulation Test

San Onofre Unit 2 is a Combustion Engineering System 80 unit, whose Reactor Coolant System is characterized by two steam generators and four Reactor Coolant Pumps (or two hot legs and four cold legs (Figure 6.3)). The geometry and operation parameters of the RCS are given in Appendix H.

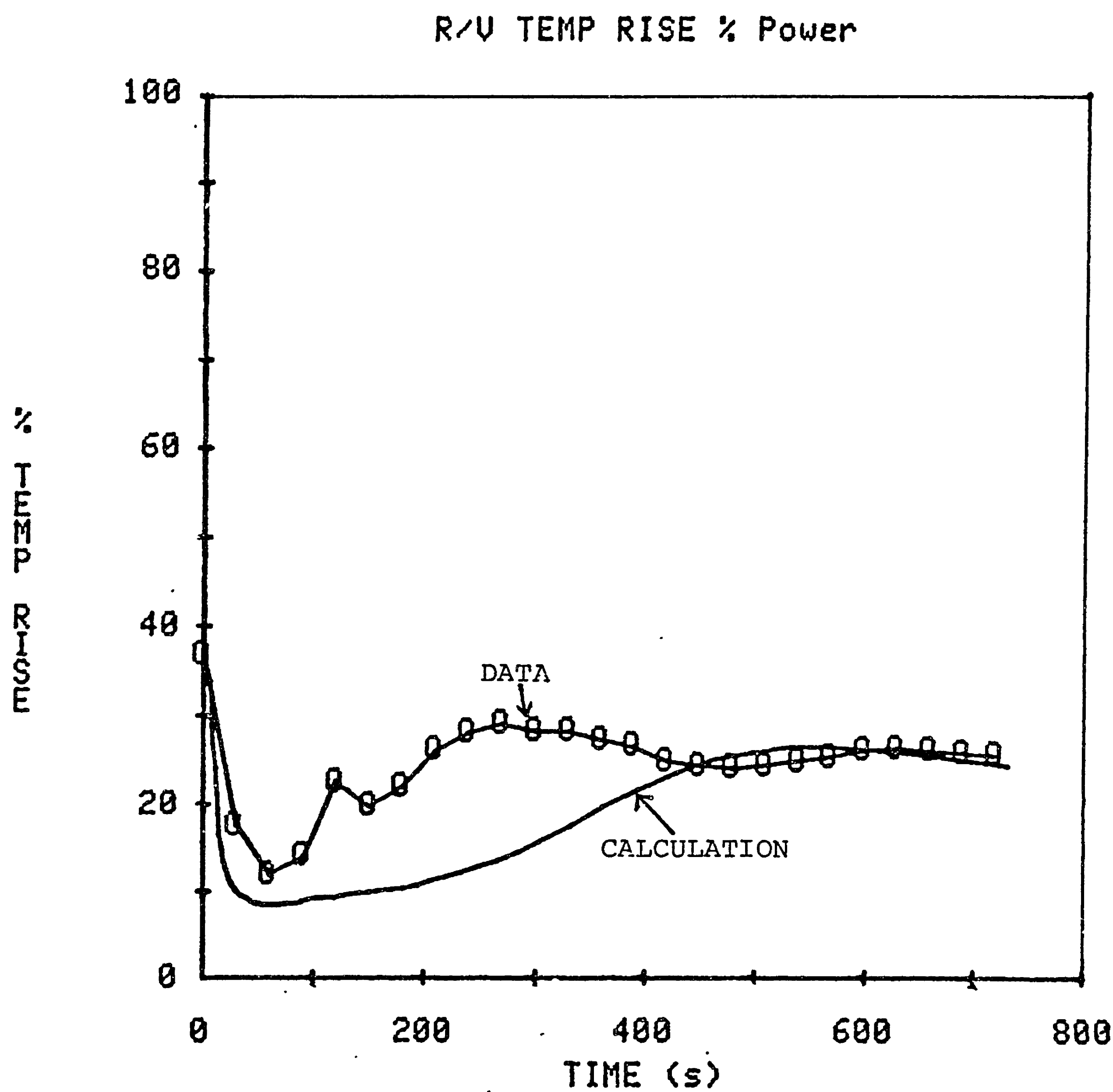


FIGURE 6.1 Maine Yankee Natural Circulation Reactor Vessel Temperature Rise History.

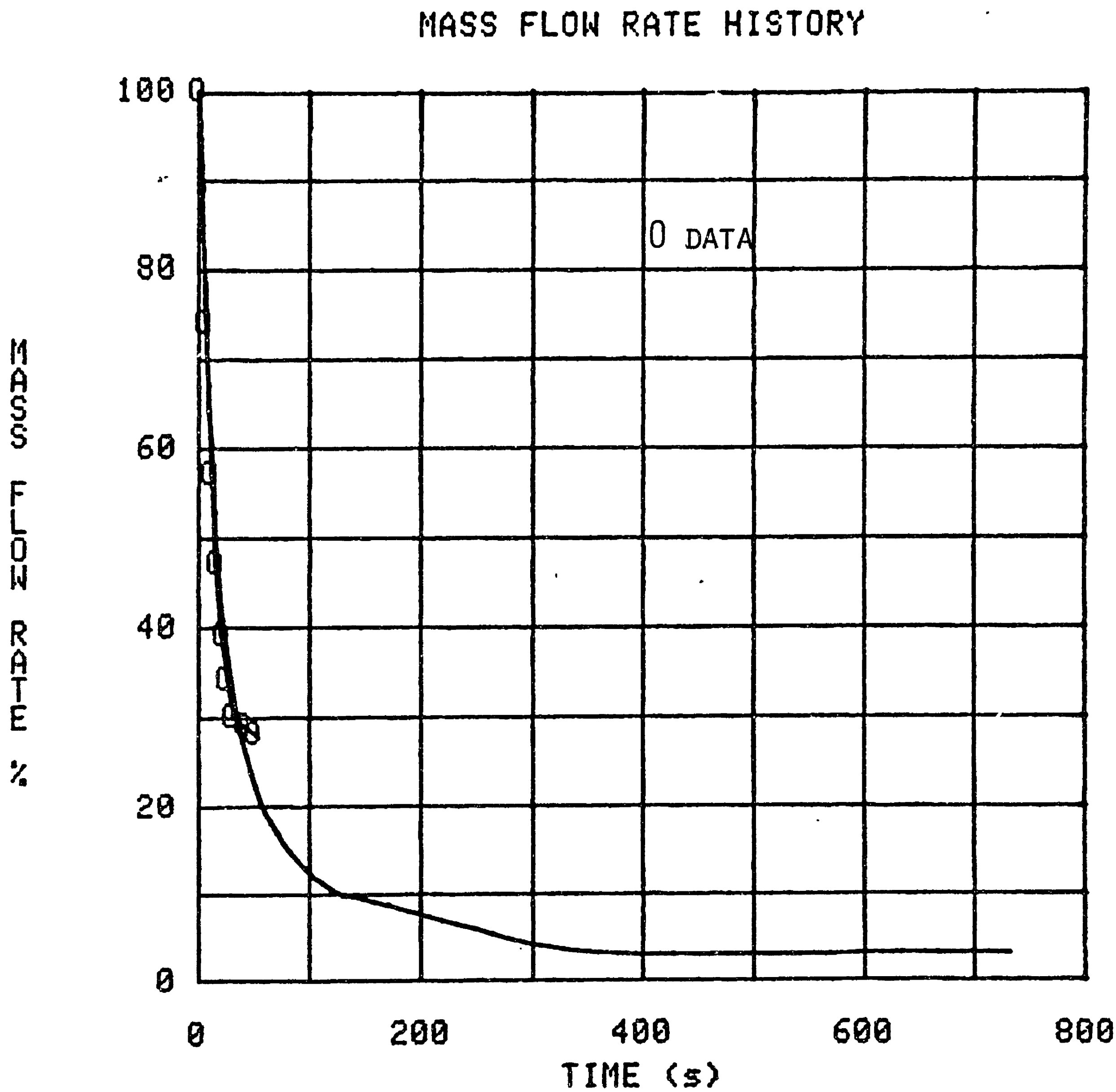


FIGURE 6.2 Calculated Maine Yankee Natural Circulation Mass Flow Rate.

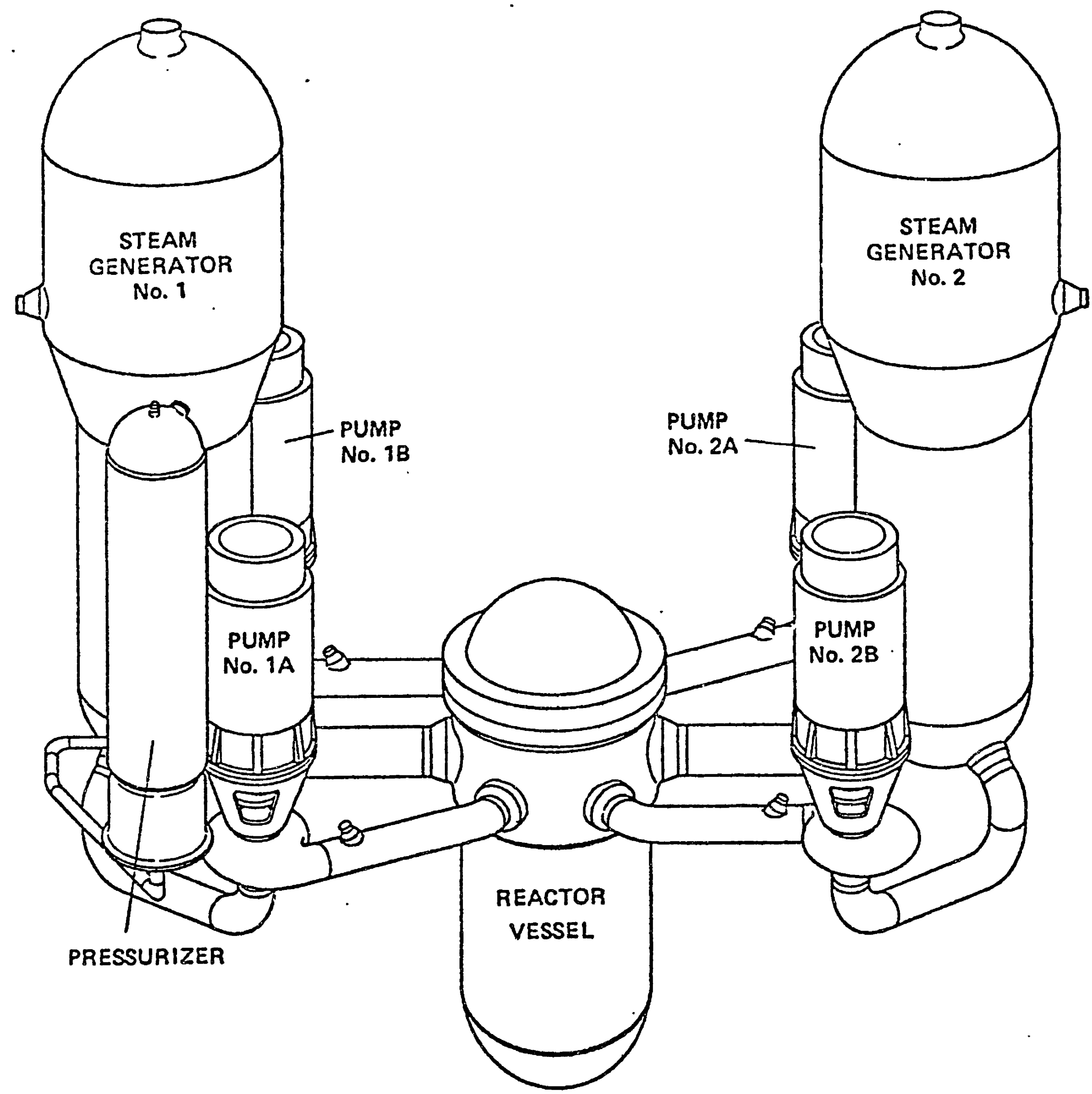


FIGURE 6.3 Isometric View of San Onofre 2 Reactor Coolant System [S2].

The test was conducted using a critical reactor maintained between 1% and 3% of rated thermal power to simulate decay heat to verify natural circulation in the Reactor Coolant System [C5]. The test was initiated by simultaneously tripping all four Reactor Coolant Pumps. The steam generator secondary pressure was maintained at 6.9 MPa.

The calculated reactor vessel temperature rises at ten minutes after the pump trip are compared with the data at power levels of 0.85%, 1.5%, 2.2%, and 3.2% (Figure 6.4). The calculations are based on adjusting the pump loss coefficient to match the 1.5% power datum, then using the same loss coefficient in the other cases. The calculated temperature rises agree well with the data in all cases.

6.3 LOFT Natural Circulation Experiments

The Loss-of-Fluid Test facility is a 50 MW thermal Pressurized Water Reactor system designed to simulate the major components and system responses of a commercial PWR during anticipated transients and loss-of-coolant accidents (LOCA).

The primary system of LOFT facility consists of an intact loop and a broken loop, as illustrated in Figure 6.5. The intact loop consists of a steam generator, a pressurizer, and two primary coolant pumps. The broken loop consists of a steam generator simulator, a pump simulator, two isolation valves, two quick opening valves, a supression

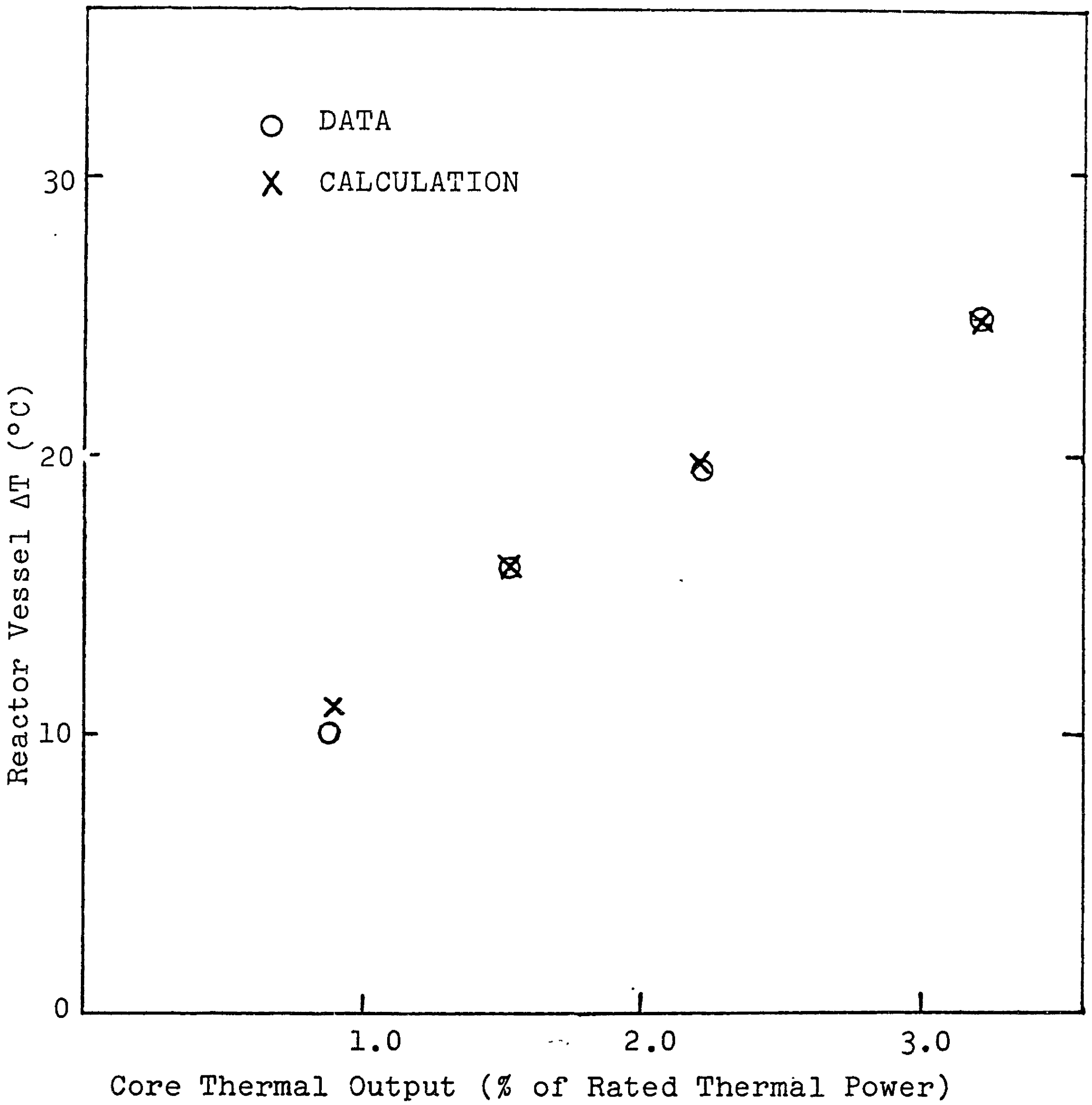


FIGURE 6.4 Comparison of San Onofre 2 Data with Calculated Reactor Vessel ΔT vs Core Thermal Output

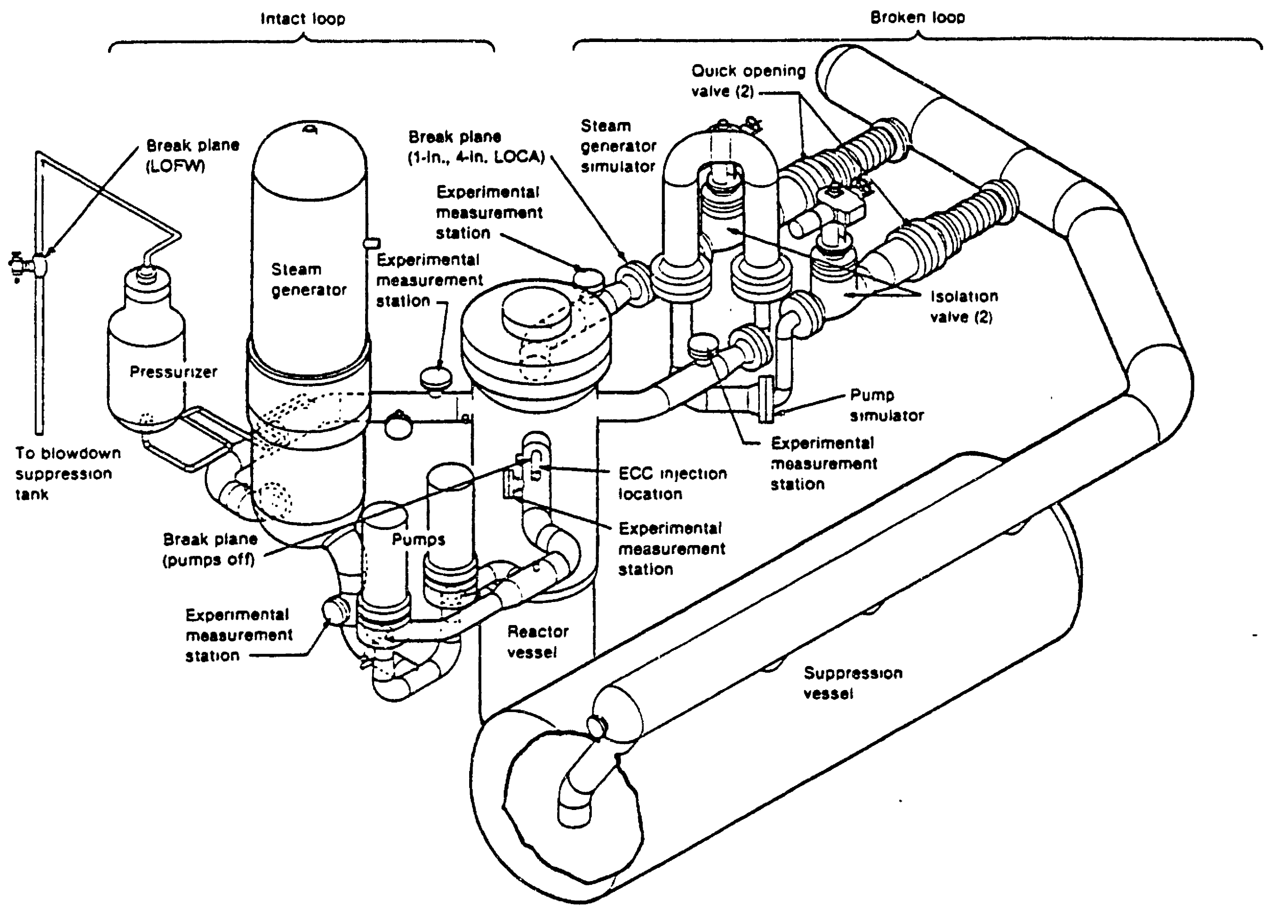


FIGURE 6.5 LOFT Primary System Configuration [A1].

vessel header, and a pressure suppression vessel. Figure 6.6 depicts component elevations of the intact loop relevant to natural circulation.

The experiments conducted at LOFT facility range from anticipated transients (such as loss-of-flow) to large break LOCA [N3]. From these experiments, two are chosen to study the capability of SPK code in single- and two-phase natural circulation. The first test is a loss-of-flow experiment (Experiment L6-2 [B5]). The second test is a small break experiment (Experiment L3-7 [G2]), in which a stable two-phase natural circulation was observed.

6.3.1 Loss-of-Flow Experiment

Experiment L6-2 simulated a loss-of-flow transient in a large scale PWR. The test was conducted at a initial power of 37.2 MW thermal. The transient was initiated by tripping the power to the pump motors. Two seconds after the pump trip, the reactor was scrammed by the plant protection system upon indication of a low primary coolant flow. At 18 seconds, the pump rotors were disengaged from the flywheel. The experiment was terminated by re-starting the primary coolant pumps.

The first 200 seconds of the experiment is simulated using the SPK code. The calculated pressure and press' 'izer water level results are shown in Figure 6.7 and 6.8, respectively. Also shown in the figure are RETRAN calculation results (Ref. [N3]). The results of SPK

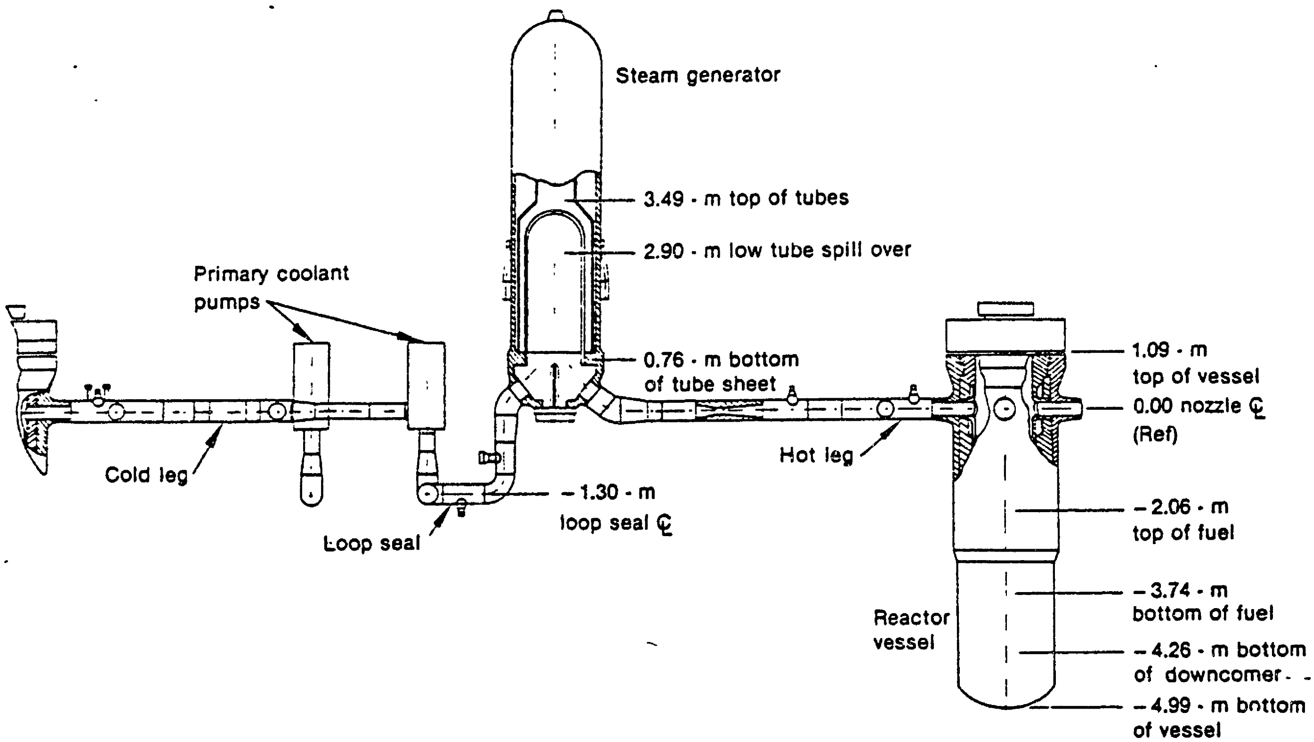
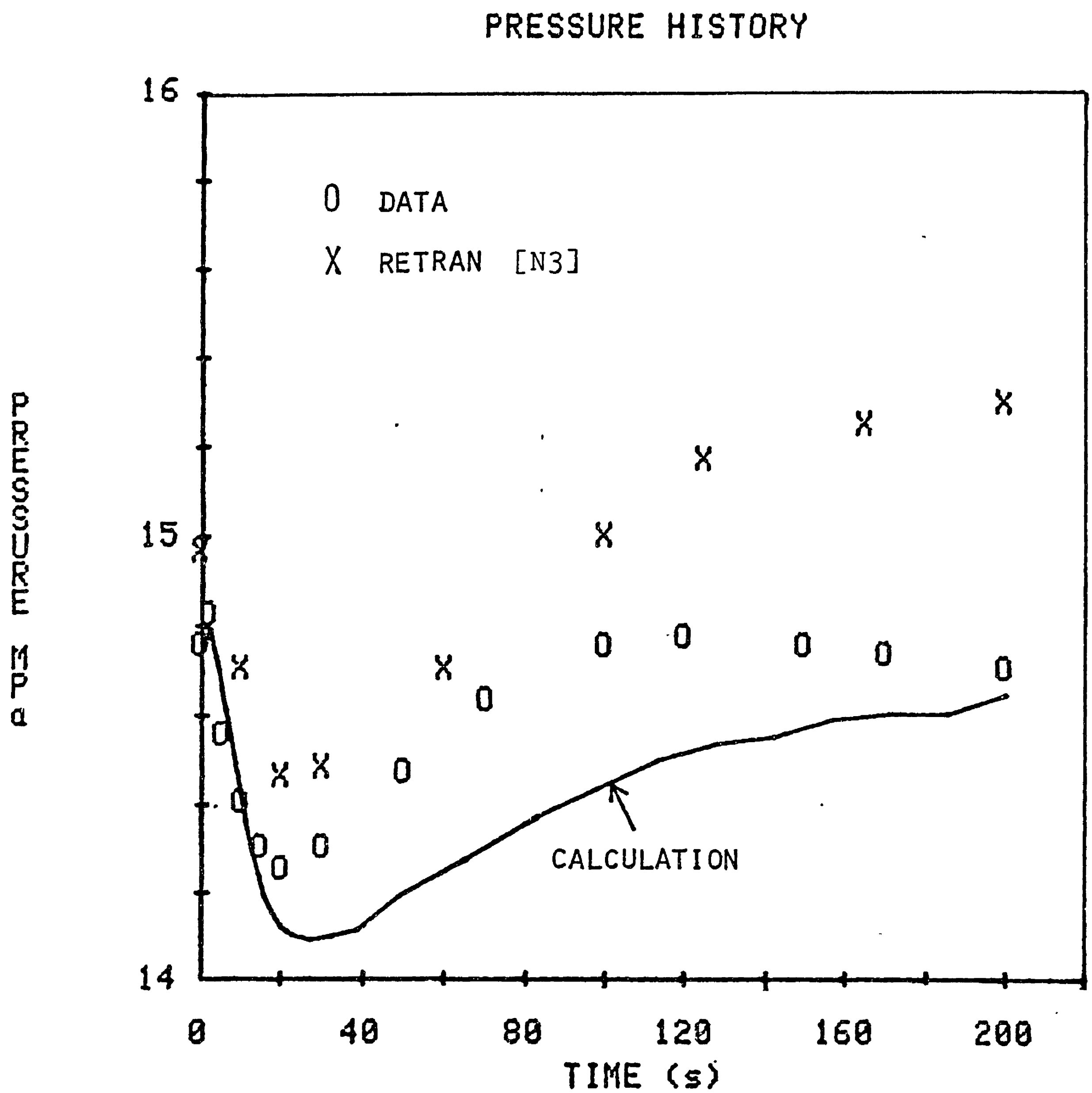


FIGURE 6.6 LOFT Intact Loop Depicting Elevations Relevant to Natural Circulation [A1].



IFUGRE 6.7 LOFT Experiment L6-2 Pressure History.

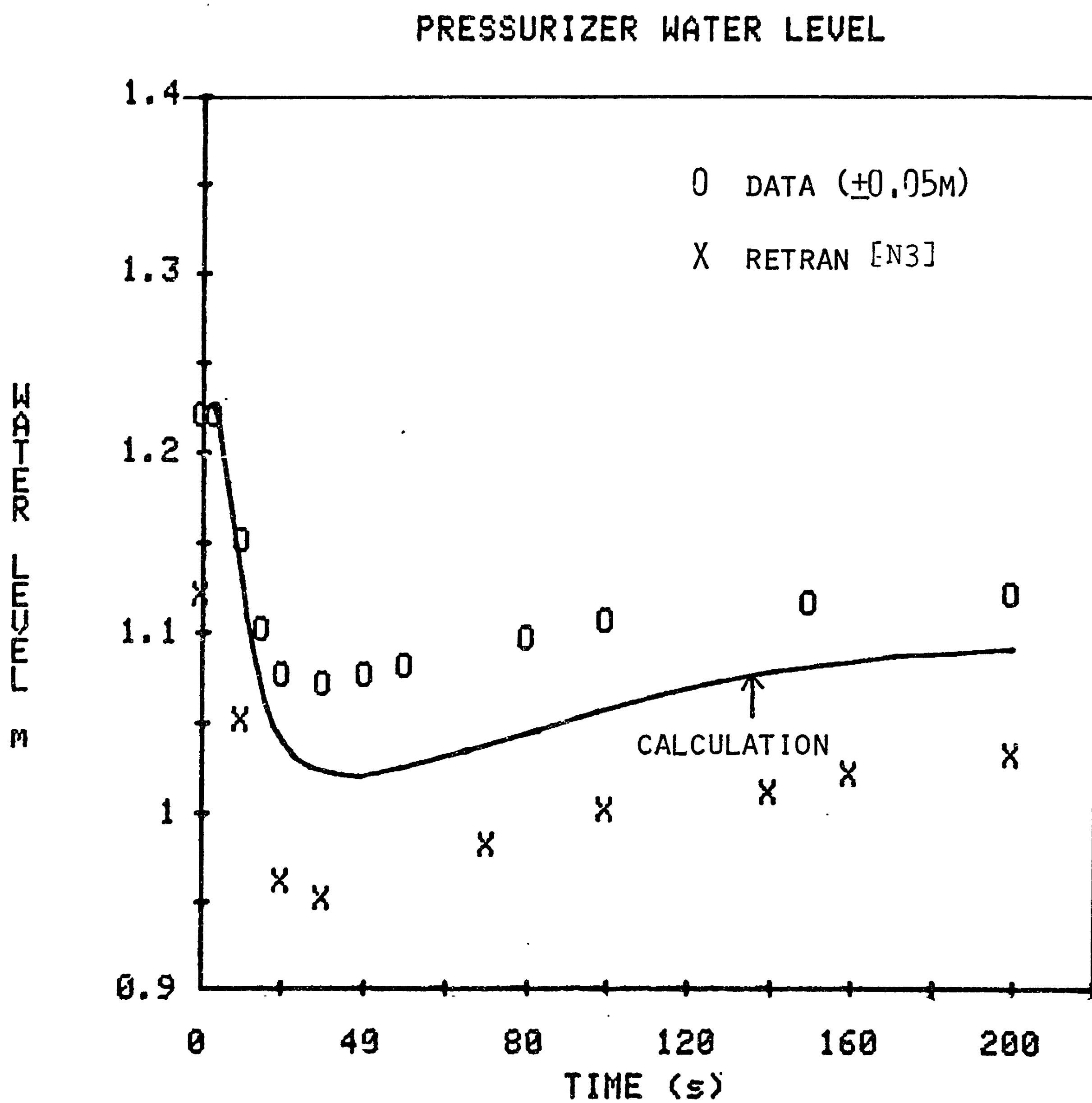


FIGURE 6.8 LOFT Experiment L6-2 Pressurizer Water Level History.

calculations are around the uncertainty limits of the data and better than that of RETRAN calculations.

The calculated mass flow rate history in percent of full flow are illustrated in Figure 6.9. Both SPK and RETRAN calculations are in close agreement. The data indicate that the mass flow rate reaches zero at 27 seconds after pump trip. However, calculations show a flow coastdown to steady natural circulation. The discrepancy suggests that the mass flow rate may be beyond the sensitive range of the flow measurement sensor.

The calculated pump speed history is shown in Figure 6.10. The results of hot and cold leg temperature calculations are shown in Figure 6.11. The calculated hot leg temperature lags behind the data initially, then converges to the data. The calculated cold leg temperature is within the uncertainty of the data.

6.3.2 Small Break Experiment

Experiment L3-7 simulated a small break equivalent to a break 25 mm in diameter in a cold leg of a four-loop commercial PWR. The sequence of events for the first 1000 seconds of the experiment is described below. Prior to the initiation of the break, the broken loop was isolated and the reactor was brought to 49 MW thermal power. The break was initiated by opening the isolation valve in the broken-loop cold leg. The primary coolant flowed through an orifice (13.2 mm diameter), then through the isolation valve

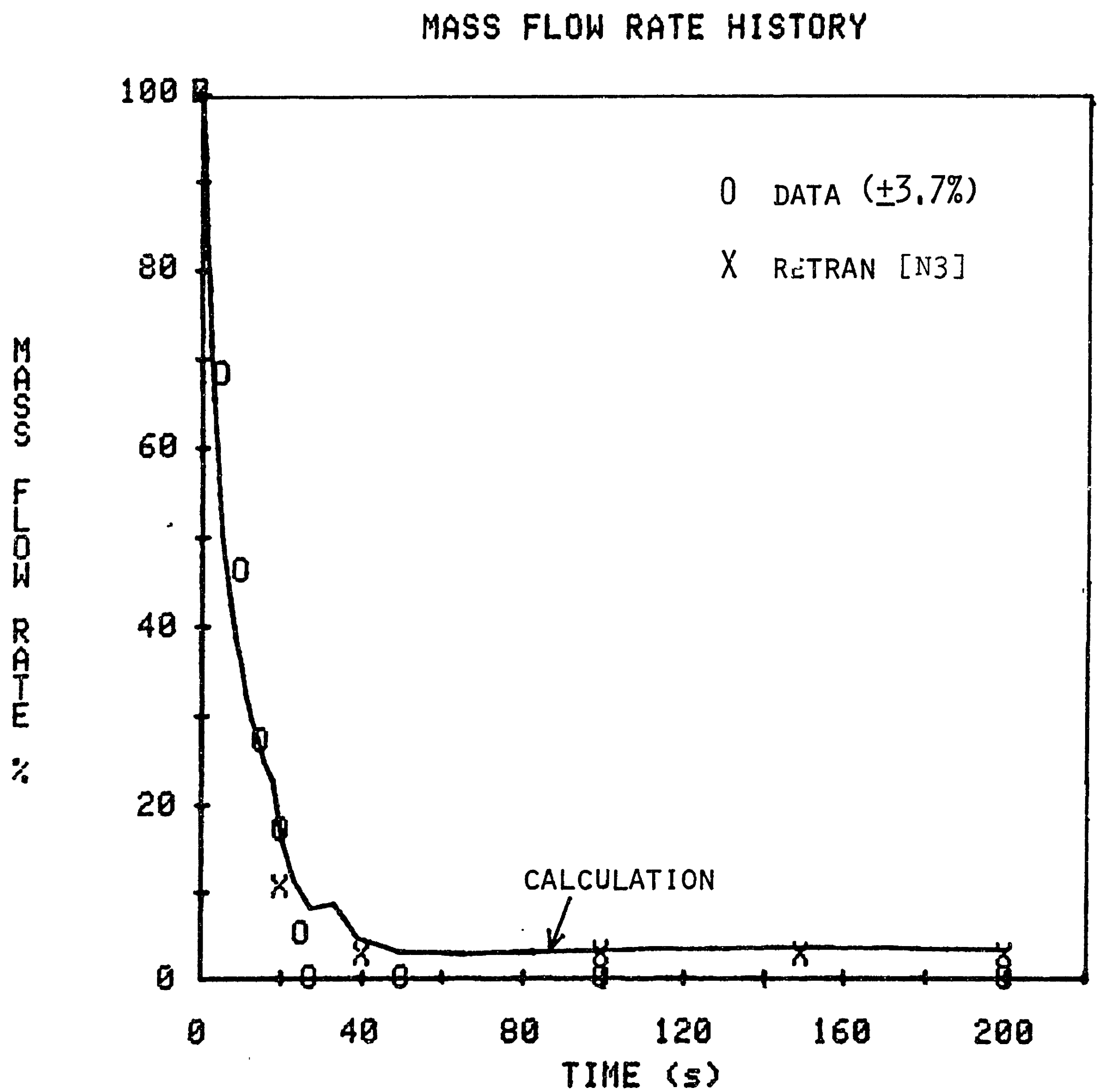


FIGURE 6.9 LOFT Experiment L6-2 Hot-Leg Mass Flow Rate History.

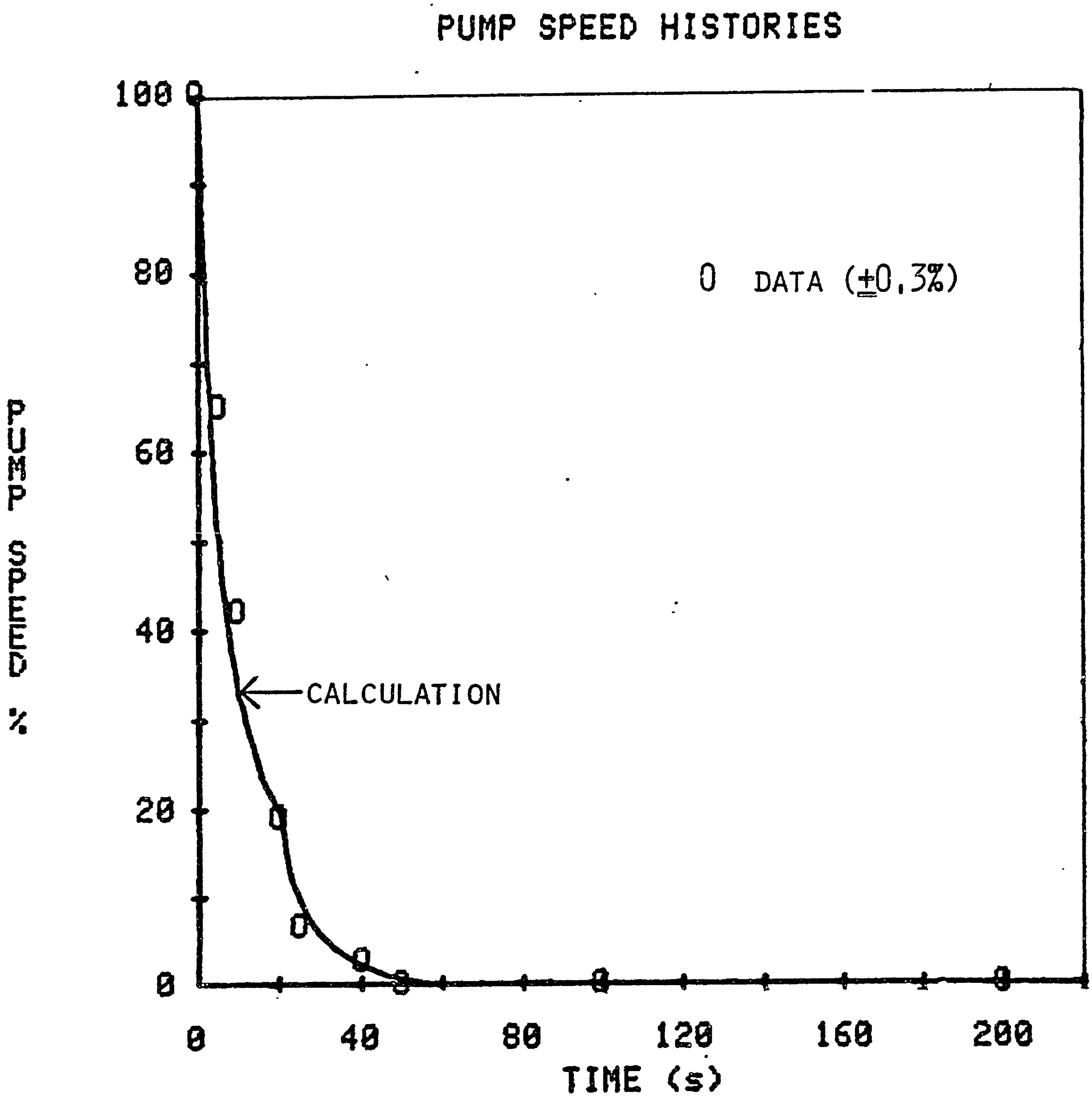


FIGURE 6.10 LOFT Experiment L6-2
Pump Speed History.

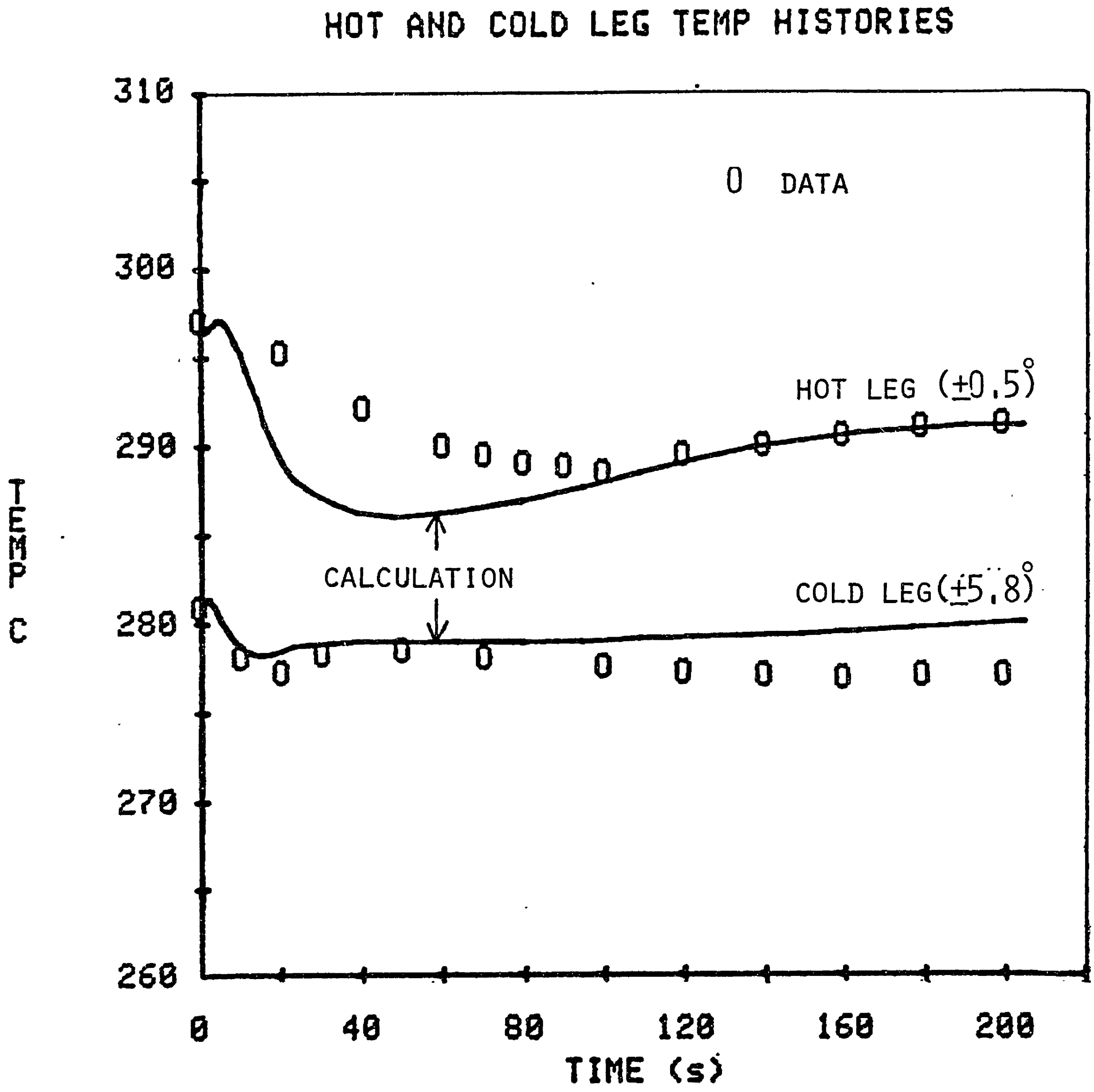


FIGURE 6.11 LOFT Experiment L6-2 Hot and Cold Leg Temperature Histories.

into the pressure supression vessel header. The reactor was scrammed at 36 seconds after break initiation. The primary coolant pumps were manually tripped at 39.3 seconds and the flywheels were disengaged from the pump rotors at 56.2 seconds. The High Pressure Injection System (HPIS) was activated when the primary system pressure decreased to 13.2 MPa at 65.6 seconds. The pressurizer was emptied at 264 seconds. Saturation condition in the primary system was detected at 382 seconds. Two-phase blowdown commenced at 1037 seconds.

The calculated pressure history is shown in Figure 6.12. As shown, the results are in good agreement with the reported data. In the calculation, a time step size of two seconds is used when the pressurizer vapor fraction is above 95%, in order to avoid instability. Also shown in the figure are results of a RELAP5/MODG calculation [N3].

The calculated mass flow rate history is shown in Figure 6.13. The initial drop in mass flow rate is caused by the decreasing fluid density caused by depressurization. The calculated single-phase and two-phase natural circulation mass flow rates are significantly less than indicated by data. The difference may lie in the fact that the broken loop is not included in the calculation (the difficulty is that the integral-loop momentum model of the SPK code can not treat a broken loop). Instead, the break is included in the cold leg of the intact loop. The calculation assumes a symmetrical condition, but in reality,

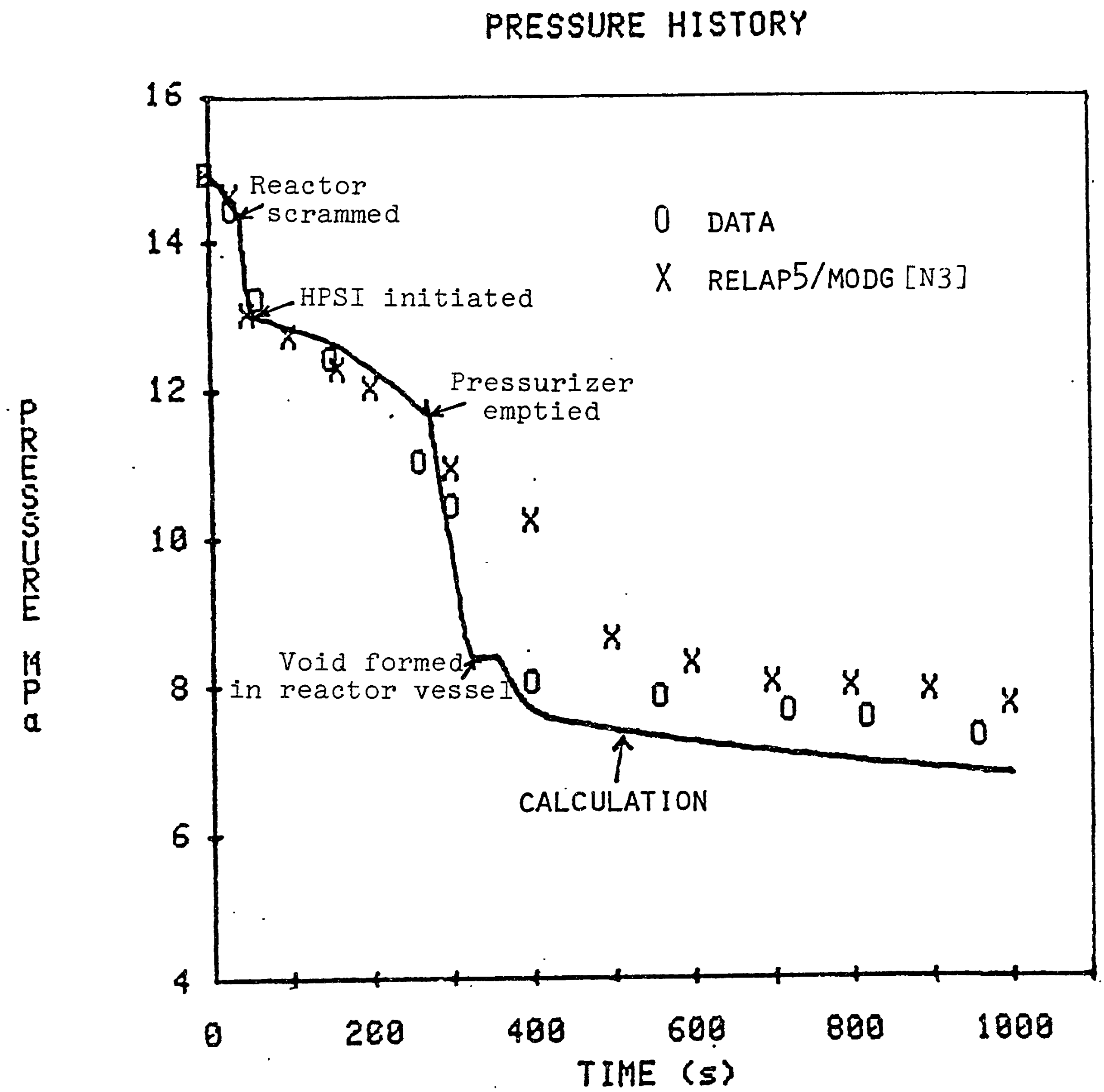


FIGURE 6.12 LOFT Small Break Experiment L3-7 Pressure History.

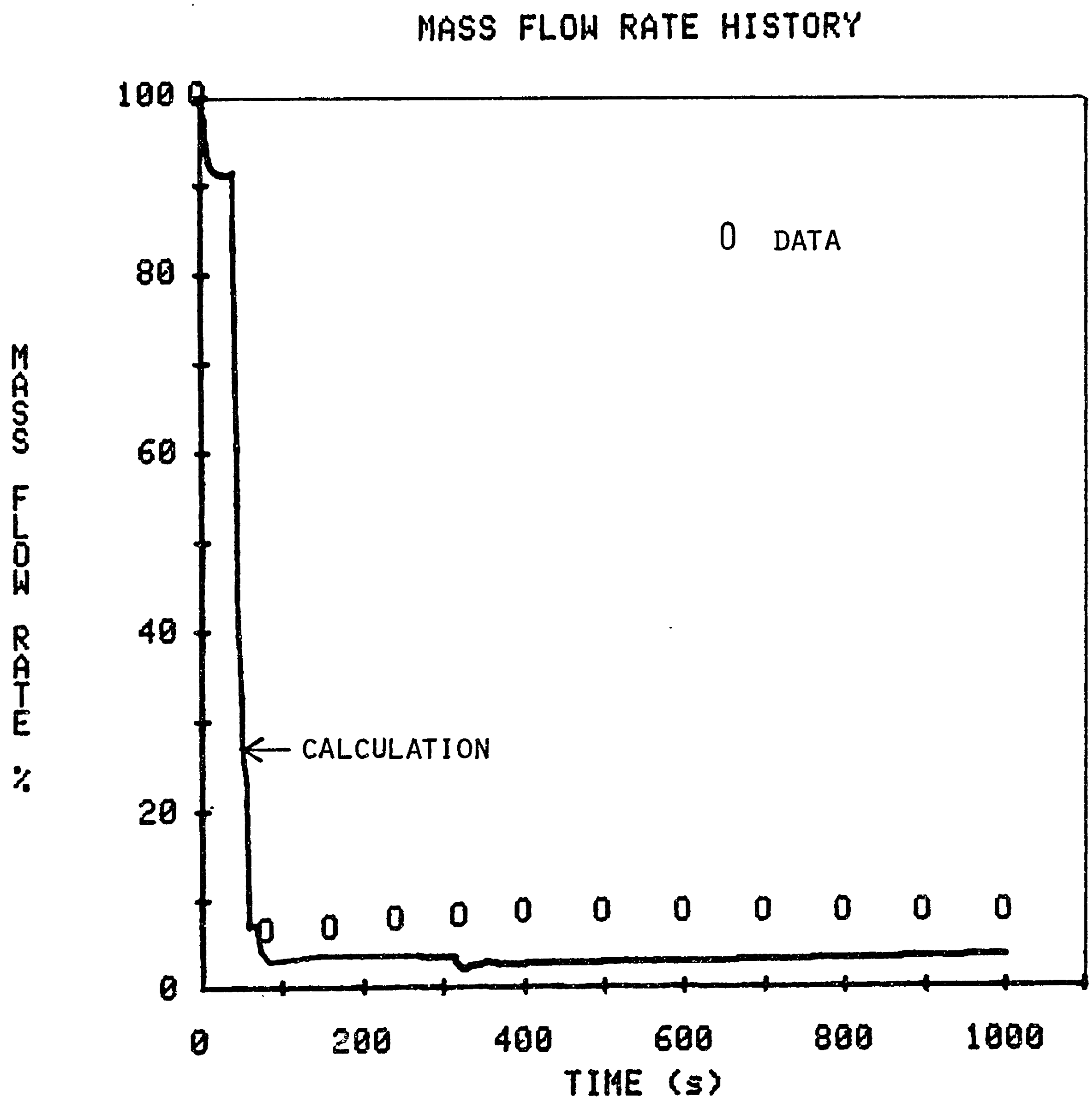


FIGURE 6.13 LOFT Small Break Experiment L3-7
Hot-Leg Mass Flow Rate History.

the system is asymmetric.

6.4 Computational Speed

The computational speed of the SPK code on the MULTICS system (Honeywell DPS 11/80 M computer) is summarized in Table 6.1 (excluding the CPU time for printing output). The calculation requires less than 50 ms of CPU time per calculation step. Under natural circulation in a commercial PWR, required CPU time is over several hundred times less than real time. Whereas, under 100% forced circulation, CPU time is more than 20 times less than real time.

6.5 Summary

The results of natural circulation calculations using the SPK code are compared with plant data of Maine Yankee, San Onofre Unit 2, and the LOFT Facility in this chapter. The following conclusions are made regarding to the code's capability:

1) A good single-phase natural circulation capability is indicated by the close agreement between the calculated temperature-rise across the reactor vessel and the plant test data.
2) A good two-phase natural circulation capability is indicated by obtaining a stable two-phase natural circulation calculation.
3) The code is able to treat a small break LOCA as shown by the successful simulation of the LOFT

TABLE 6.1 Computational Speed of SPK Code in Natural Circulation Calculations

Experiment	Simulation Time (s)	CPU Time (s)	CPU/Time Step (ms)	Real/CPU Time
San Onofre 2	1500	2.77	25	540
Maine Yankee	700	1.96	34	360
LOFT (L6-2)	210	1.49	43	140
LOFT (L3-7)	1000	21.1	45	50

Experiment L3-7.

4) Comparisons with RETRAN and RELAP5/MODG results show that SPK code performs as good and sometimes better than these major system codes.

A MULTIPLE-LOOP PRIMARY SYSTEM MODEL FOR PRESSURIZED WATER REACTOR PLANT SENSOR VALIDATION

Vol. 2

by

SHIH-PING KAO

B. S. Georgia Institute of Technology
(June 1979)

S. M. Massachusetts Institute of Technology
(June 1981)

Submitted to the Department of
Nuclear Engineering in partial
fulfillment of the requirements
for the Degree of

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

July 1984

Signature of Author ______________________
Department of Nuclear Engineering
July 1984

Certified by ______________________
Professor John E. Meyer
Thesis Supervisor

Certified by ______________________
Doctor John H. Hopps, Jr.
Co-Thesis Supervisor

Accepted by ______________________
Professor Allan F. Henry, Chairman
Departmental Graduate Committee

CHAPTER 7

COMPATIBILITY WITH AN OPERATOR ASSIST SYSTEM

In this chapter, two example signal validation approaches for a PWR Reactor Coolant System are proposed that use the analytic model developed in this thesis. These approaches use the signal validation concept being developed by the Charles Stark Draper Laboratory (CSDL).

7.1 Background

The CSDL signal validation technique uses a parity-space representation and analytic redundancy. The technique has been successfully used in three nuclear power applications.

First, the technique was applied to a steam generator feedwater system using Combustion Engineering's ZAMBO3 code as the plant [M2]. It was able to detect and isolate failures of the steam generator pressure sensors, the main feed-valve differential pressure sensors, and the feedwater temperature sensors.

In the second case, the technique was applied to the measurement of primary sodium flowrate in the EBR-II reactor [D1]. The signal validation program was implemented on the EBR-II data acquisition computer (Xerox Sigma-5), and executed with data sampling rate of one per second. The technique successfully detected and isolated any sensor failure when each of the 15 different sensor inputs was deliberately failed.

Finally, the technique was applied to the automatic power-level control system of the MIT Research Reactor II (MITR-II) [R2]. The power level instrumentation consists of three neutron flux sensors and a gamma-ray sensor that correlates neutron power with the radioactivity (Nitrogen-16) of the primary coolant. The digital control system replaced the original analog controller.

The technique has also been proposed for use on a Safety Parameter Display System [H3], to achieve high information reliability even under multiple failures of sensor inputs. The structure of the SPDS based on this concept is modular.

The RCS model developed in this thesis may be used to provide analytic measurements to supplement validate plant sensor signals. These measurements are especially valuable in places where sensor signals are less than three, since a reliable estimate of a system variable can be obtained from at least three independent measurements. In places where more than three sensor measurements are present, addition of analytic model augments the reliability of the estimate.

In this chapter, application of the analytic model to signal validation is examined. Two applications are proposed. The first one deals with the validation of the pressurizer water level. The second one deals with the validation of mass flow rate through the core under natural circulation.

7.2 Validation of Pressurizer Water Level

In many older PWRs there are only two water level sensors in the pressurizers. These sensors provide information to the level controller which adjusts the Chemical and Volume Control System (CVCS). In order to isolate any sensor failures, an additional measurement is required. In this case, the computer code, SPK code, can provide an analytic measurement of the pressurizer water level. Figure 7.1 illustrates the signal validation flow diagram. It shoud be noted that the computed value is a collapsed water level. The pressurizer model discussed in Chapter 3 alone cannot complete the calculation, since the surge mass flow rate is an unknown in the model and is not a directly measured variable.

7.3 Validation of Natural Circulation Mass Flow Rate

In operation of a PWR, natural circulation is often employed in long-term decay-heat removal operations. However, unlike in forced circulation, mass flow rate in natural circulation can neither be measured directly nor be inferred from pump information. This loss of signal occurs since the pressure drops across the sensors are outside their sensitive range. The remaining sensor indications related to the mass flow rate are from the temperature measurements in the hot and cold legs.

The steady-state relationship between the mass flow rate and the temperature difference in each loop is given

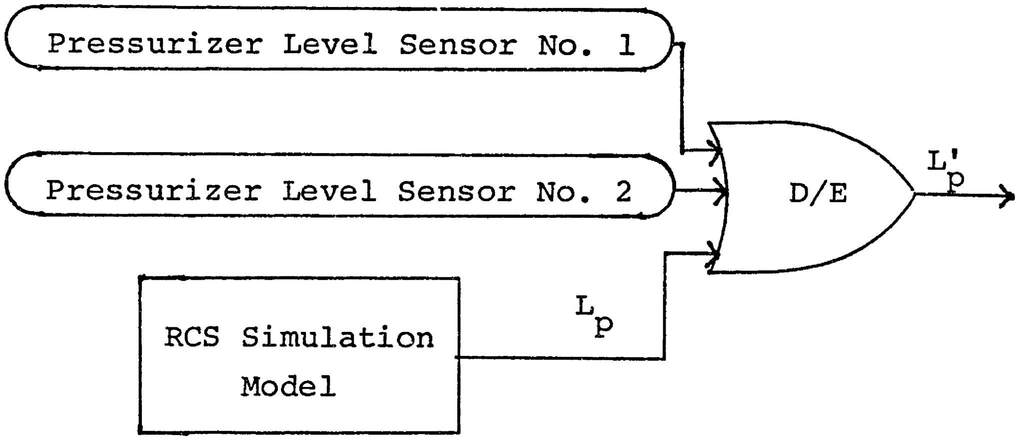


FIGURE 7.1

Pressurizer Level Sensor Validation Diagram by:

$$W\langle C_p\rangle \ \Delta T = Q_{SG} \tag{7.1}$$

The steam generator power can be calculate by using the log-mean temperature difference assumption; that is:

$$Q_{SG} = A_o U_o \ \Delta T_{\ell m} \tag{7.2}$$

or in a functional form:

$$Q_{SG} = f(T_{HL}, T_{CL}, T_S, W) \tag{7.3}$$

Combining Eqs. (7.1) and (7.3), we obtain an implicit relation for W in terms of the hot leg, cold leg, and secondary side temperatures:

$$W = f'(T_{HL}, T_{CL}, T_S, W) \tag{7.4}$$

Eq. (7.4) must be solved iteratively to yield solutions for W. A referred mass flow rate through the core is equal to the sum of the individual loop mass flow rate obtained in Eq. (7.4), i.e.:

$$W_C = \sum_{i=1}^{m} W_i \tag{7.5}$$

where m = number of cold legs.

An additional analytic measurement for W_C can be obtained by utilizing the core-exit thermocouple temperature measurements. In natural circulation under decay heat, the steady-state energy balance across the core is:

$$Q_D = W_C(T_{Co} - T_{Ci}) \tag{7.6}$$

where

T_{Co} = average core exit temperature,

T_{Ci} = average core inlet temperature, and

Q_D = decay power.

The decay power can be calculated by Eq. (2.40). The average core exit temperature is determined by taking the mean of the thermocouple measurements, as:

$$T_{Co} = \frac{1}{n} \sum_{k=1}^{n} (T_{tc})_k \tag{7.7}$$

where

n = number of thermocouples, and

tc = core exit thermocouple.

Similarly, the core inlet temperature is approximately equal to the flow weighted average of cold-leg temperature measurements:

$$T_{Ci} = \frac{1}{W_C} \sum_{i=1}^{m} (WT_{CL})_i \tag{7.8}$$

Combining Eqs. (7.6) to (7.8) obtain:

$$W_C = \frac{Q_D - \sum_{i=1}^{m} (WT_{CL})_i}{\frac{1}{n} \sum_{k=1}^{n} (T_{tc})_k} \tag{7.9}$$

A third analytic measurement for W_C can be provided by the present RCS model. The validation diagram is given in Figure 7.2.

However, note that these thermocouples have a typical operating range of 268 to 324 C. Therefore in a cold-shutdown heat removal operation, the RCS model becomes the only source of information on the natural circulation mass flow rate.

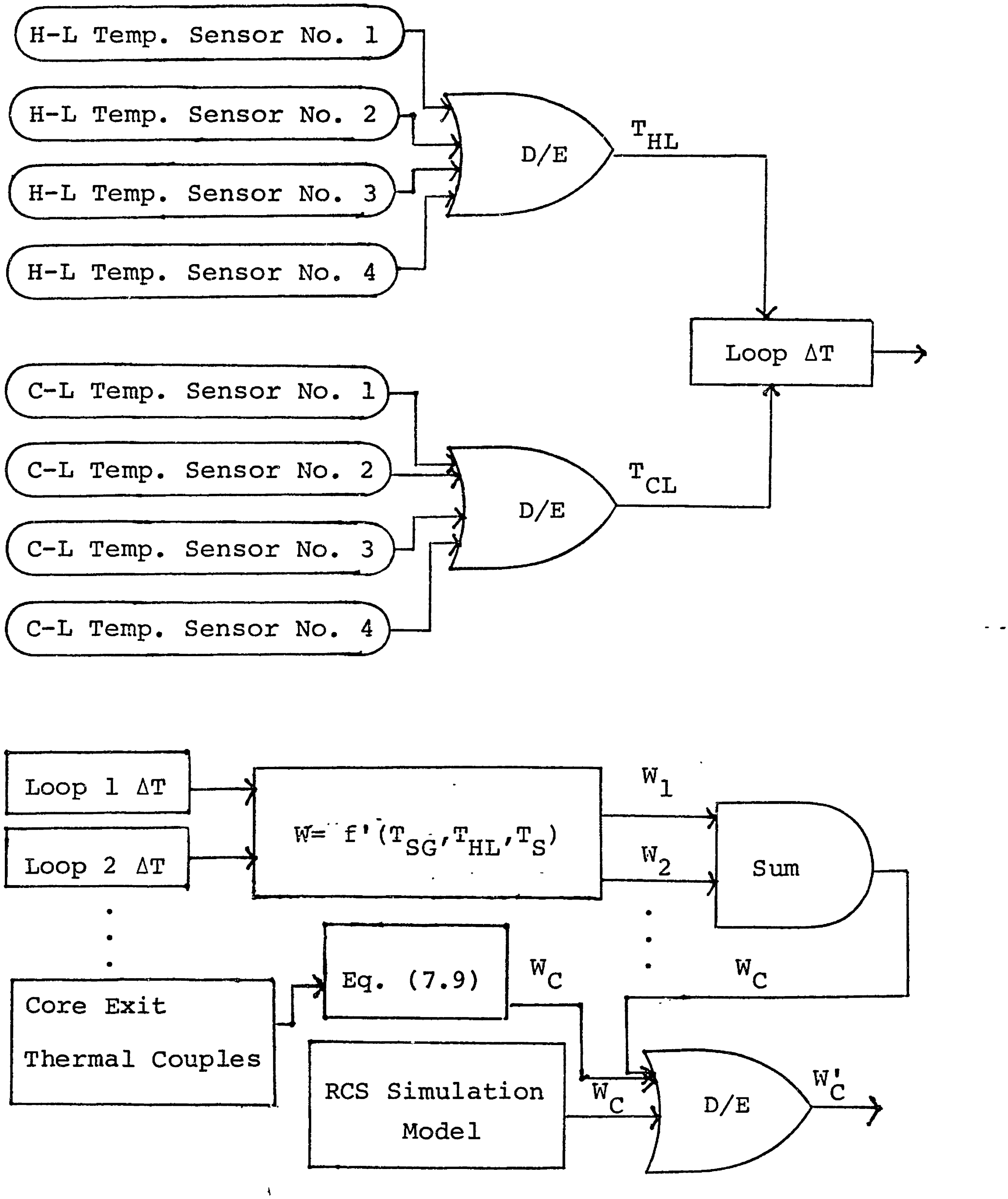


FIGURE 7.2 <u>Natural Circulation Mass Flow Rate Sensor Validation Model</u>

CHAPTER 8

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

8.1 Summary

The motivation for this research is the belief that an on-line information system is essential to enhance reliable operation of a commercial Pressurized Water Reactor. It is further believed that the system should include a physically based analytical model that adequately simulates the dynamics of the plant. Major system codes (such as RETRAN and RELAP) have not been developed for real-time online simulation. The available real-time simulation codes are very limited in application. Therefore, the task of this research is to develop a system code that includes both adequate physical models and faster-than-real-time computation capability.

The major physical features of this code are summarized below.

1) The Reactor Coolant System is divided into the following components: reactor core, RV upper plenum, RV lower plenum, hot legs, cold legs, steam generator U-tubes, and pressurizer.

2) Except for reactor vessel plena and pressurizer, the mass and energy conservation equations are integrated over the control volume assuming a linear enthalpy profile.

3) In the reactor vessel plena, a complete mixing assumption is applied.

4) A drift-flux model is used to describe the degree of phase separation.

5) The pressurizer model is based on a two-fluid assumption that properties are uniform in each phase. Flashing and rainout are included to avoid meta-stability in the liquid and vapor phase, respectively. Condensation on the vessel wall is determined by including a one-lump conduction model with a film condensation model as a boundary condition.

6) Heat transfer in the core is characterized by a single-node fuel-rod transient conduction model. Decay power is calculated by summing the powers produced by seven decay-product groups, which are adequate to simulate decay power to twelve days after scram.

7) Steam generator heat transfer rate is characterized by a steady-state, zero-dimensional analysis, defined in terms of a log-mean temperature difference and an overall heat transfer coefficient.

8) The momentum model consists of the loop integral momentum equations and the pump-rotor speed differential equations. A constitutive equation is included to determine the hydrostatic head in each recirculation loop. Complete four quadrant pump characteristic relations for pump

head-capacity and torque-capacity are also included.

The salient numerical features adopted in the model are enumerated as follow:

1) A donor-cell differencing scheme is adopted in the energy finite difference equations; hence, flow reversal is allowed.

2) The finite-difference equations are fully implicit and developed by using a linearization at each time step.

3) The computer model consists of three basic models: flow path mass-energy, pressurizer, and momentum. Each model is represented by a system of state equations defined in terms of state variables. The system of state equations are solved simultaneously using a Gauss reduction method.

4) The pressurizer state equations are solved using a two-step calculation. The first step determines the condition in each phase assuming subcooled liquid and superheated vapor to exist in the final state. The second calculation is performed with adjustments for rainout, or flashing, or both, if so required.

5) Multiple-loop configurations up to four independent loops are provided.

6) Variable time-step size scheme is adopted, in which the time-step size is equal to the fluid transport time throught

the core. When an empty pressurizer is encountered during the calculation, a fixed time-step size (two seconds) is used.

8.2 Conclusions

The following conclusions are made based on the comparisons of calculated results to experimental data:

1) Pressurizer Response--The pressurizer model is shown to follow closely the dynamics of both PWR and small-scale pressurizers. The transients studied include outsurge, in-outsurge, and empty pressurizer conditions.

2) Pump Coastdown--Calculations show smooth coastdown of pump impeller after a pump trip. Reslts are in good agreement with plant data.

3) Flow Reversal--The code is demonstrated to work satisfactorily from a calculational standpoint in reverse flow (in a multiple-loop configuration).

4) Single-Phase Natural Circulation--The model is shown to work well in single-phase natural circulation by comparing with plant temperature measurements.

5) Two-Phase Natural Circulation--A good two-phase natural circulation capability is indicated by comparison with LOFT data.

6) Small Break LOCA--The code is capable of providing good small-break LOCA information as shown by the LOFT simulation. The calculated pressure results compare well with those of RELAP5/MODG.

7) Faster-Then-Real-Time Simulation--Computation is less than 50 ms per time step on the Honeywell DPS 8/70M computer. This gives a real-to-CPU time ratio of better than twenty when simulating a commercial PWR RCS (with a single-loop configuration) operated in forced circulation, and the ratio is more then one hundred when the RCS is operating in single-phase natural circulation.

8.3 Recommendations

The work in this research produces an analytic tool that can be applied directly to operation of a commercial PWR. The following is a list of recommendations for further refinement and application of the model developed in this thesis.

1) Signal Validation--Using the approach of Chapter 7, we suggest that the computer model be applied directly to the validation of pressurizer level and the validation of nautral circulation mass flow rate.

2) Pressurizer Model Upgrade--The present model cannot treat a pressurizer that is filled with liquid (i.e., "solid" pressurizer). Some modifications can be made to

include such capability.

3) Incorporation of A Neutronic Model--A point kinetics or few-group diffusion neutronic model can be incorporated into the core representation.

4) Validation--A fully examination of the model's applicability to two phase condition should be completed. Application to transients such as stuck-open relief valve and steam-generator tube rupture transients should be included.

5) PWR Plant Simulator--The model can be coupled with a secondary system model to obtain a complete plant simulator model.

6) Dynamic Graphic Display--Interactive graphic display software can be incorporated so that the user can visualize simulated plant dynamics on a faster-than-real-time basis.

APPENDICES

APPENDIX A

Derivation of Conservation Equations

This appendix discusses a derivation of one-dimensional conservation equations for a fixed boundary control volume. For a rigorous derivation, the reader is referred to the work in Meyer [M6].

A.1 Conservation of Mass

The conservation principle of mass is:

$$\begin{bmatrix}\text{Rate of Change} \\ \\ \text{of Mass}\end{bmatrix} + \begin{bmatrix}\text{Rate of Efflux} \\ \\ \text{of Mass}\end{bmatrix} = 0 \qquad \text{(A.1)}$$

For a control volume ΔV with surface S and fluid velocity $\underline{u}$, Eq. (A.1) becomes:

$$\frac{\partial}{\partial t} \int_{\Delta V} \rho dV + \int_{S} (\rho \underline{u}) \cdot d\underline{S} = 0 \qquad \text{(A.2)}$$

Consider a control volume with a constant flow area A, i.e., $\Delta V = A \, \Delta z$, as shown in Figure A.1. Define a volume averaged quantity $\langle \psi \rangle$ as:

$$\langle \psi \rangle = \frac{1}{A} \int \psi dA \qquad \text{(A.3)}$$

Carryout the integrations in Eq. (A.2), divide by $A \, \Delta z$, and

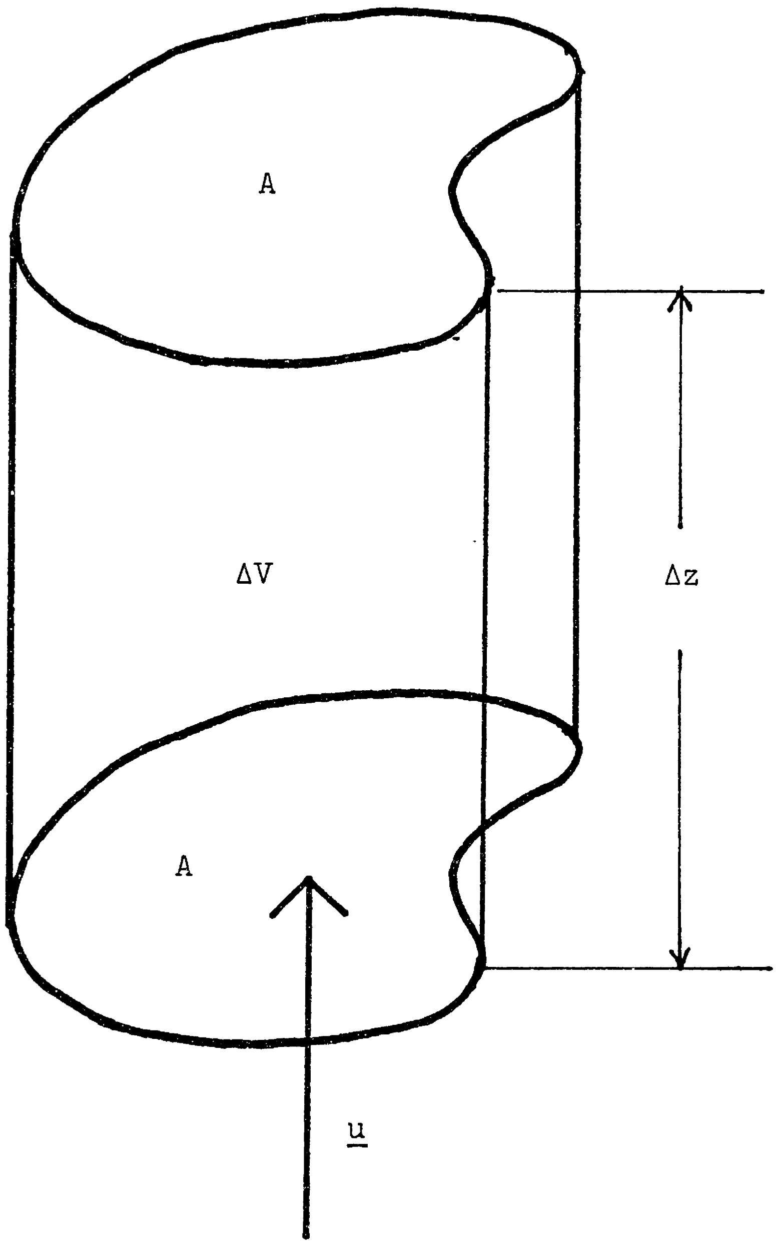


FIGURE A.1 Control Volume for Conservation Laws.

then let Δz approach to zero to obtain:

$$\frac{\partial}{\partial t} \langle \rho_v \alpha + \rho_\ell (1 - \alpha) \rangle + \frac{\partial}{\partial Z} \langle \rho_v u_v \alpha + \rho_\ell u_\ell (1 - \alpha) \rangle = 0 \quad (A.4)$$

Denote:

$$\rho_m = \langle \rho_v \alpha + \rho_\ell (1 - \alpha) \rangle \quad (A.5)$$

= mixture density, and

$$G = \langle \rho_v u_v \alpha + \rho_\ell u_\ell (1 - \alpha) \rangle \quad (A.6)$$

= mass velocity.

Eq. (A.4) becomes:

$$\frac{\partial \rho_m}{\Delta t} + \frac{\partial G}{\partial z} = 0 \quad (A.7)$$

Integrate Eq. (A.7) over a macroscopic control volume V = A z (with a constant flow area A), to give:

$$\frac{dM}{dt} = W_{in} - W_{out} \quad (A.8)$$

where

$$M = \int_V \rho_m \, dV \quad (A.9)$$

and

$$W = \int_A G \, dA \quad (A.10)$$

A.2 Conservation of Momentum

The conservation principle of momentum is:

$$\begin{bmatrix}\text{Rate of Change} \\ \\ \text{of Momentum}\end{bmatrix} = \begin{bmatrix}\text{Sum of External} \\ \\ \text{Forces}\end{bmatrix} \qquad \text{(A.11)}$$

For a control volume ΔV with surface S, Eq. (A.11) becomes:

$$\frac{\partial}{\partial t}\int_{\Delta V} \rho \underline{u}\, dV + \int_S (\rho \underline{u})(\underline{u} \cdot d\underline{S}) = \int_S \underline{\underline{\sigma}} \cdot d\underline{S} + \int_{\Delta V} \rho \underline{g}\, dV \qquad \text{(A.12)}$$

where $\underline{\underline{\sigma}}$ is the shear stress tensor. Assume flow is uniform in the z-direction and flow area is constant, Eq. (A.12) becomes:

$$\frac{\partial G}{\partial t} + \frac{\partial}{\partial z}\langle v' G^2 \rangle = -\frac{\partial p}{\partial z} + \frac{p_w \tau_w}{A} - \rho_m g \sin\theta \qquad \text{(A.13)}$$

where

$$v' G^2 = \langle \rho_v u_v^2 \alpha + \rho_\ell u_\ell^2 (1 - \alpha) \rangle \qquad \text{(A.14)}$$

$$\frac{p_w \tau_w}{A} = \frac{f v |G| G}{2 D_h} \qquad \text{(A.15)}$$

p_w = wetted perimeter, and

τ_w = shear stress form wetted surface acting on the fluid.

Integrate Eq. (A.13) over a macroscopic control volume V = A z, to give:

$$\frac{dW}{dt} + \Delta \left(\frac{v'W^2}{A^2}\right) = -\Delta P + \frac{fv|W|W}{2A^2} \frac{z}{D_h} - \rho_m V g \sin \theta \qquad (A.16)$$

A.3 Conservation of Energy

The conservation principle of energy is:

$$\left[\begin{array}{l}\text{Rate of Change in}\\ \\ \text{Intrinsic Energy}\end{array}\right] = \left[\begin{array}{l}\text{Rate of Work done}\\ \\ \text{by External Forces}\end{array}\right] + \left[\begin{array}{l}\text{Rate of Heat}\\ \\ \text{Addition}\end{array}\right] \qquad (A.17)$$

For a control volume ΔV with surface S, Eq. (A.18) becomes:

$$\frac{\partial}{\partial t}\int_{\Delta V} (\rho e)\, dV + \int_S (\rho e)\underline{u} \cdot d\underline{S} = \int_S (\underline{\underline{\sigma}} \cdot \underline{u}) \cdot d\underline{S} + \int_{\Delta V} \rho \underline{g} \cdot \underline{u}\, dV + \int_S - \underline{q}'' \cdot d\underline{S} + \int_{\Delta V} q''' \, dV \qquad (A.18)$$

where

$$e = H - P/\rho + \frac{1}{2} \underline{u} \cdot \underline{u} \qquad (A.19)$$

= specific internal energy

If we neglect the following:

1) kinetic energy ($1/2\ u^2$),

2) work done by gravity ($\rho \underline{g} \cdot \underline{u}$),

3) work done at bottom and top surfaces (except pressure forces), and

4) work done by strsses at wall,

for a constant for area A, Eq. (A.19) becomes:

$$\frac{\partial}{\partial t} (\rho_m H_m) + \frac{\partial}{\partial z} (GH') = \frac{\partial P}{\partial t} + \frac{q'' p_h}{A} + q''' \qquad (A.20)$$

where

$$H_m = \frac{1}{\rho_m} \langle \rho_v H_v \alpha + \rho_\ell H_\ell (1 - \alpha) \rangle \qquad (A.21)$$

= mixture enthalpy,

$$H' = \frac{1}{G} \langle \rho_v u_v H_v \alpha + \rho_\ell u_\ell H_\ell (1 - \alpha) \rangle \qquad (A.22)$$

= mixing cup enthalpy,

p_h = heated perimeter,

q'' = heat flux averaged over p_h, and

q'''= volumetric heat generation rate.

Integrate Eq. (A.20) over a macroscopic control volume V (with a constant flow area A), to give:

$$\frac{\partial}{\partial t}(\rho_m H_m)V + \Delta(WH') = V\frac{\partial P}{\partial t} + q' p_h z + q''V \tag{A.23}$$

A practical use of Eq. (A.23) is to evaluate the fluid density with a single reference pressure P^* and neglect local pressure effect. That is [M5]:

$$\rho_m = \rho_m(H_m, P^*) \tag{A.24}$$

and

$$\frac{\partial P}{\partial t} \Rightarrow \frac{dP^*}{dt} \tag{A.25}$$

the result is that the sonic effect is eliminated from Eq. (A.23) and since that local compressibility is not included any pressure and velocity perturbations propagate at an infinite velocity.

APPENDIX B

Empirical Correlations

All correlations presented in this appendix are expressed in SI units.

B.1 Steam/Water Properties

The properties for water and steam are evaluated as functions of enthalpy and pressure. The property correlations are taken from the SSC-L computer code developed by the Brookhaven National Laboratory [G3]. They are not repeated here except for the density of superheated steam where mis-prints are found in the original report. The density of superheated steam is given by:

$$\rho_v(H_v, P) = 1/[g_0 + g_1 H_v] \qquad (B.1)$$

where

$$g_0 = g_{00} + g_{01}P + g_{02}/P$$

$$g_1 = g_{10} + g_{11}P + g_{12}/P$$

and

$$g_{00} = -5.1026 \times 10^{-5}$$

$$g_{01} = 1.1208 \times 10^{-10}$$

$$g_{02} = -4.4506 \times 10^{5}$$

$$g_{10} = -1.6893 \times 10^{-10}$$

$g_{11} = -3.3980 \times 10^{-17}$

$g_{12} = 2.3058 \times 10^{-1}$

B.2 UO2 and Zircaloy Properties

The property correlations for the fuel (UO2) and cladding (Zircaloy) materials are taken from the THERMIT [K1] code developed at MIT. Please refer to the original report for their expressions.

B.3 Inconel 600 Properties

The expressions for the thermal properties of Inconel 600, which is the material used in the steam generator tubes, are taken from Reference [S1].

B.4 Heat Transfer Correlations

B.4.1 Convective Heat Transfer Correlations

The single-Phase convective heat transfer coefficient is given by the Dittus-Boelter Correlation, as:

$$Nu = 0.023 \, (Re)^{2.8} \, (Pr)^{0.3} \tag{B.2}$$

where

Nu = Nusselt number = $h \, D_h / k$;

Re = Reynolds number = $G \, D_h / \mu$; and

Pr = Prandlt number = $C_p \, \mu / k$.

The two-phase convective heat transfer coefficient is given by the Thom Correlation as [S1]:

$$h = \frac{(T_w - T_{sat}) \exp(2P/8.7 \times 10^6)}{22.65 \times 10^{-3}} \qquad (B.3)$$

B.4.2 Film Condensation Heat Transfer Correlation

The film condensation heat transfer coefficient is given by [R8] [C3]:

$$h = 2.442 \times 10^2 \left[\frac{g \rho_f (\rho_f - \rho_g) k_f^3 H_{fg}}{\mu_f \ell_w (T_v - T_w)} \right]^{1/4} \qquad (B.4)$$

This correlation was developed by Nusselt with the following assumptions:

1) flow in the film is laminar,
2) fluid properties are constant,
3) subcooling of the condensate is neglected,
4) momentum change in the film is neglected,
5) vapor is stationary, and
6) heat transfer is by conduction only.

B.5 Drift-Flux Correlations

The vapor void fraction defined by the drift flux model

[Z1] is given by:

$$\langle \alpha \rangle = \frac{\langle j_g \rangle}{C_o \langle j \rangle + U_{gj}} \tag{B.5}$$

where

$\langle\ \rangle$ = averaged over the cross-sectional area,

α = void fraction,

j = mixture velocity,

j_g = superficial vapor velocity,

U_{gj} = drift velocity, where

$$U_{gj} = \frac{\langle \alpha (u_g - j) \rangle}{\langle \alpha \rangle} \tag{B.5.a}$$

u_g = vapor velocity, and

C_o = void distribution parameters, where

$$C_o = \frac{\langle \alpha j \rangle}{\langle \alpha \rangle \langle j \rangle} \tag{B.5.b}$$

Since the mixture velocity is given by:

$$\langle j \rangle = \langle \alpha \rangle \langle j_g \rangle + (1 - \langle \alpha \rangle) \langle j_f \rangle \tag{B.6}$$

where

$$\langle j_g \rangle = G\, X/\rho_g, \text{ and} \tag{B.7}$$

$$\langle j_f \rangle = G(1 - X)/\rho_f \tag{B.8}$$

Therefore, the relation between the void fraction the flow quality is obtained by substituting Eqs. (B.6), (B.7), and

(B.8) into Eq. (B.5), to give:

$$\langle\alpha\rangle = \frac{X}{C_o[X + (\rho_g/\rho_f)(1 - X)] + \rho_g U_{gj} A/W} \quad (B.9)$$

The drift velocity and the void distribution parameter must be determined empirically, since the void and velocity profiles are not known analytically. It is observed experimentally that U_{gj} is flow-regime dependent and C_o is geometry and pressure dependent. In the churn-turbulent flow regime, the Zuber Correlation [Z1] is recommended to determined the drift velocity:

$$U_{gj} = 1.41 \left[\frac{\sigma g(\rho_f - \rho_g)}{\rho_f^2}\right]^{1/4} \quad (B.10)$$

A fit to the above equation as a function of pressure only is given by [S1]:

$$U_{gj} = (6.41 \times 10^{-17})P - (5.7794 \times 10^{-9})P + 0.20957 \quad (B.11)$$

In virtical flows, the distribution parameter is a function of hydraulic diameter and pressure, given by [C3]:

$$\begin{aligned} C_o &= 1.5 - 0.5\ Pr \quad &; D > 5\ cm \\ &= 1.2 \quad &; D < 5\ cm,\ Pr < 0.5 \\ &= 1.2 - 0.4(Pr - 0.5) \quad &; D < 5\ cm,\ Pr > 0.5 \end{aligned} \quad (B.12)$$

where

$$Pr = P/(22.09 \times 10^6)$$

In inverted U-tubes, Nagumo and co-workers [N4] have concluded that Eq. (B.10) adequately correlates the drift velocity, and that the distribution parameter is adequately determined by Ishii's Correlation [I1]:

$$C_o = 1.2 - 0.6(\rho_g/\rho_f)^{1/2} \tag{B.13}$$

B.6 Friction Loss Correlations

Pressure loss due to friction is given by:

$$\Delta P = \frac{fLW^2}{2\rho_\ell D_h A^2} \tag{B.14}$$

where

A = flow cross-sectional area,

D_h = hydraulic diameter,

f = friction factor,

L = flow passage length,

W = mass flow rate, and

ρ_ℓ = fluid density.

In single-phase turbulent flow, the friction factor can be determined from the Moody Diagram, which may be approximated by the following equation [M10]:

$$f = \frac{0.25}{[\log(\frac{\varepsilon}{3.7D_h} + \frac{5.74}{Re^{0.9}})]^2} \tag{B.15}$$

where is the roughness parameter, for a smooth steel pipe it has the value of 0.025 mm.

In case of two-phase flow, Eq. (B.13) is multiplied by a two-phase multiplier. The correlation of Martinelli-Nelson-Jones [L1] is used in claculating two-phase friction loss. That is:

$$\phi^2 = \Omega\, 1.2\, (\rho_f/\rho_g - 1)\, X^{0.824} + 1 \qquad (B.16)$$

where

$$\Omega = 1.43 + \left[(g - G_o)/G\right](0.07 - 7.35 \times 10^{-8} P); \quad G \geq G_o$$

$$\Omega = 1.43 + (G_o/G - 1)\,(0.17 - 6 \times 10^{-8} P); \quad G < G_o$$

$$G_o = 950 \text{ kg/m-s}$$

For pressure losses due to form changes, the homogeneous two-phase multiplier is used, which is given by:

$$\phi^2 = 1 + X(\rho_f/\rho_g - 1) \qquad (B.17)$$

APPENDIX C

Steam Generator Heat Transfer Rate Calculations

The calculations are based on the results given in Reference [S1]. The procedure is extended here to include two-phase flow cases on the primary side.

As discussed in Chapter 2, the heat transfer rate in the steam generator is given by:

$$Q_{SG} = A_o U_o \Delta T_{\ell m} \tag{C.1}$$

where

$$\frac{1}{U_o} = \left[\frac{A_o}{A_i h_i} + \frac{r_o \ell n(r_o/r_i)}{k_f} + \frac{1}{h_o} + r_{ff}\right] \tag{C.2}$$

where subscript i indicates tube-side and o indicates shell-side. The heat transfer coefficient on the shell side is given by the Thom Correlation as:

$$h_o = \frac{Q_{SG}/A_o}{(T_w - T_s)} = \frac{\sqrt{Q_{SG}/A_o}}{22.65 \times 10^6} \exp(P_o/8.7 \times 10^6) \tag{C.3}$$

The assumption here is that the fluid on the shell side is always in the bubbly flow regime. Since h_o is a function of Q_{SG}, Eq. (C.1) must be solved implicitly. Let

$$z_2 = 22.65(A_o/10^6)^{1/2} \exp(-P_s/8.7 \times 10^6) \tag{C.4}$$

Substituting Eq. (C.4) into (C.3) obtain:

$$h_o = \sqrt{Q_{SG}}/Z_2 \tag{C.5}$$

Let

$$Z_1 = \frac{A_o}{A_i h_i} + \frac{r_o \ell n(r_o/r_i)}{B_t} + r_{ff} \tag{C.6}$$

Substituting Eqs.(C.6) and (C.5) into (C.2) obtain:

$$\frac{1}{U_o} = Z_1 + Z_2/\sqrt{Q_{SG}} \tag{C.7}$$

Combine Eqs. (C.7) and (C.1), yield:

$$Q_{SG} = \frac{Z_3}{Z_1 + Z_2/\sqrt{Q_{SG}}} \tag{C.8}$$

where

$$Z_3 = A_o \ \Delta T_{\ell m} \tag{C.9}$$

Express Eq. (C.8) as:

$$Z_1 Q_{SG} + Z_2 \sqrt{Q_{SG}} - Z_3 = 0 \tag{C.10}$$

Solving for Q_{SG} using the quadratic formula, obtain:

$$Q_{SG} = \left[\frac{-Z_2 + \sqrt{(Z_2)^2 + 4Z_1 Z_3}}{2Z_1}\right]^2 \tag{C.11}$$

In the case which $\Delta T_{\ell m}$ is negative, i.e., heat is transferring from the secondary to the primary side, Eq. (C.11) becomes:

$$Q_{SG} = -\left[\frac{-Z_2 - \sqrt{(Z_2)^2 + 4Z_1 Z_3}}{2Z_1}\right]^2 \tag{C.12}$$

If two-phase condition exists on the tube side, it is assumed that the Thom correlation can be applied to the two-phase section of the tube side as well. In this case the steam generator heat transfer rate is the sum of the heat transfer rates from the single- and two-phase section:

$$\begin{aligned} Q_{SG} &= (Q_{SG})_{1p} + (Q_{SG})_{2p} \\ &= (A_o U_o \Delta T_{\ell m})_{1p} + (A_o U_o)_{2p}(T_i - T_o)_{sat} \end{aligned} \tag{C.13}$$

where subscript 1p indicates sigle-phase section and 2p indicates two-phase section. The areas are also divided into two sections as:

$$A = A_{1p} + A_{2p} \tag{C.14}$$

where

$$A_{1p} = n_t \pi D(L - L_b) \tag{C.15}$$

$$A_{2p} = n_t \pi D L_b \tag{C.16}$$

n = number of steam generator tubes, and

L_b = two-phase length.

Assuming linear enthalpy distribution, the two-phase length,

L_b, is approximately given by:

$$L_b = \left(\frac{H_{in} - H_f}{H_{in} - H_{out}}\right) L \qquad (C.17)$$

for $H_{out} \leq H_f$. The heat transfer coefficient on the tube side is given by:

$$(h_i)_{2p} = \frac{\Delta T_w \exp(2P_i/8.7 \times 10^6)}{22.65 \times 10^{-3}} \qquad (C.18)$$

where $\Delta T_w = (T_i - T_o)$. The heat transfer coefficient on the shell side is given by:

$$(h_o)_{2p} = \frac{\Delta T_w \exp(2P_o/8.7 \times 10^6)}{22.65 \times 10^{-3}} \qquad (C.19)$$

The overall heat transfer coefficient in the two phase section is, therefore:

$$\frac{1}{(U_o)_{2p}} = \left[\left(\frac{A_o}{A_i h_i}\right)_{2p} + \frac{r_o \ln(r_o/r_i)}{k_t} + \frac{1}{(h_o)_{2p}} + r_{ff}\right] \qquad (C.20)$$

Solution for $(Q_{SG})_{1p}$ is given by Eq. (C.11) where the heat transfer area is defined by Eq. (C.15).

APPENDIX D

U-Tube Steam Generator Thermal Center and Hydrostatic Head

The thermal center concept is based on a zero-dimensional analysis in which the heat transfer process can be thought of taking place at a point inside the steam generator. A U-tube steam generator is shown schematically in Figure D.1, where a single tube with average length L and average height representative of the entire tube is shown. If the fluid inside the tube is single phase, thermal center (Z_{SG}) as measured from the bottom tube sheet is defined by the following equation:

$$\int_0^L (T - T_{out})\, ds = (T_{in} - T_{out})\, Z_{SG} \qquad \text{(D.1)}$$

If the temperature difference between the inlet and outlet is not large, we can assume that the coefficient of thermal expansion is constant; i. e.:

$$\beta = -\frac{1}{\rho}\frac{\Delta\rho}{\Delta T} = \text{constant} \qquad \text{(D.2)}$$

Using the assumption of Eq. (D.2), a relation for thermal center, as illustrated in Figure D.2, in terms of the density differences can be obtained as:

$$\int_0^L (\rho - \rho_{out})\, \underline{g} \cdot d\underline{s} = (\rho_{in} - \rho_{out})\, Z_{SG} g_o \qquad \text{(D.3)}$$

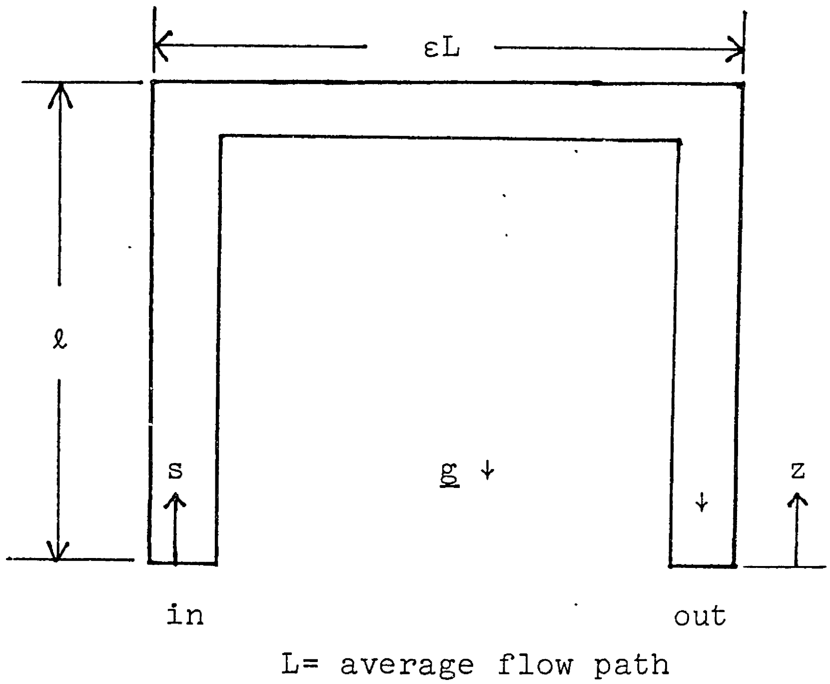


FIGURE D.1 Schematic of A U-Tube Steam Generator Tube.

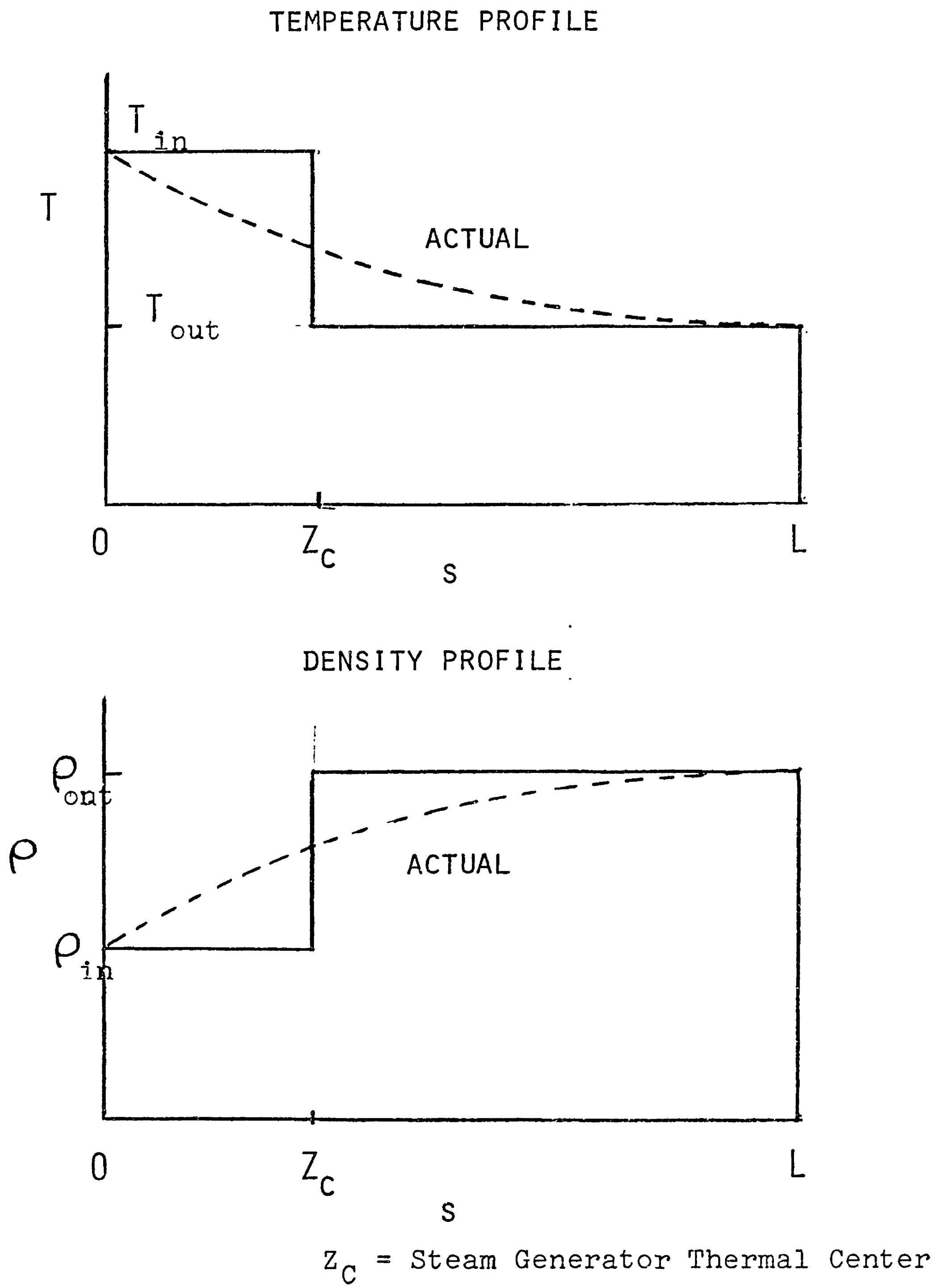


FIGURE D.2 Temperature and Density Profiles in a Steam Generator Tube.

Eq. (D.3) is the hydrostatic head inside the steam generator tubes. The right hand side of Eq. (D.3) can be expanded into two integrals as:

$$\int_0^L (\rho - \rho_{out})\, \underline{g} \cdot d\underline{s} = - \int_o^{\ell} (\rho - \rho_{out}) g \, ds + \int_{\ell+\varepsilon L}^{L} (\rho - \rho_{out}) g \, ds \qquad \text{(D.4)}$$

since

$$\underline{g} \cdot d\underline{s} = \begin{cases} g\,ds & ; \; 0 < z < \ell \\ 0 & ; \; \ell < z < \ell + \varepsilon L \\ -g\,ds & ; \; \ell + \varepsilon L < z < L \end{cases}$$

Eq. (D.4) is determined if the density profile is known. The density profile can be determined using an one-dimensional heat exchanger analysis. The average temperature gradient of the primary fluid is the following [W2]:

$$\frac{dT}{ds} = \frac{-n_t \pi D_o U_o}{WC_p} (T - T_S) \qquad \text{(D.5)}$$

where U_o = the overal heat transfer coefficient;

T_S = secondary-side saturation temperature;

n_t = number of tubes.

Rewrite Eq. (D.5) as the following:

$$\frac{d(T - T_S)}{(T - T_s)} = -ds/\ell^* \qquad \text{(D.6)}$$

where

$$\ell^* = \frac{WC_p}{n_t \pi D_o U_o} \tag{D.7}$$

Integrate Eq. (D.6) from s=0 to s, obtain:

$$T(s) - T_{in} = -(T_{in} - T_S) + (T_{in} - T_S) \exp(-s/\ell^*) \tag{D.8}$$

Substitute Eq. (D.2) into (D.7), obtain:

$$(\rho(s) - \rho_{in}) = \beta\rho_{in}(T_{in} - T_S)[1 - \exp(-s/\ell^*] \tag{D.9}$$

Substitute Eq. (D.9) into (D.4) and integrate, obtain an expression for the hydrostatic head in the steam generator tube side:

$$\int_0^L (\rho - \rho_{in}) \underline{g} \cdot d\underline{s} = \left[\frac{-(\rho_{out} - \rho_{in})g\ell^*}{1 - \exp(-L/\ell^*)}\right] \cdot$$

$$[1 - \exp(-\ell/\ell^*) + \exp(-L/\ell^*)$$

$$- \exp[(\ell + \varepsilon L)/\ell^*)] \tag{D.10}$$

Therefore, the thermal center of a U-tube steam generator is the following:

$$Z_{SG} = \frac{\ell^*[1 - \exp(-\ell/\ell^*) - \exp((\ell + \varepsilon L)/\ell^*) + \exp(-L/\ell^*)]}{(1 - \exp(-L/\ell^*)} \tag{D.11}$$

As ℓ^* approaches infinity, the thermal center approaches the following limit:

$$\lim_{\ell^* \to \infty} Z_{SG} = \left(\frac{1 + \varepsilon}{2}\right) \ell \tag{D.12}$$

If two-phase condition exists in the tube side, thermal center defined in Eq. (D.3) is not applicable. However, the hydrostatic head can be determined by a similar method.

Since the heat transfer coefficient is assumed uniform along the two-phase section of the steam generator tube, according to Thom's correlation, the heat flux is also uniform in this section. Therefore, Eq. (D.5) can be written as the following:

$$WdH = n_t \pi D_o U_o (T_i - T_S)_{sat} \, ds \tag{D.13}$$

Since

$$dH = H_{fg} \, dX \tag{D.14}$$

Substituting Eq. (D.14) into (D.13), give:

$$\frac{dX}{ds} = \frac{-n_t \pi D_o U_o (T_i - T_S)_{sat}}{WH_{fg}} = \frac{-1}{\ell^*_{2p}} \tag{D.15}$$

The two-phase density is defined as the following:

$$\rho_{2p} = (1 - \alpha)\rho_f + \alpha\rho_g \tag{D.16}$$

The two-phase density is related to the flow quality by the drift-flux model according to the following equation:

$$\alpha = \frac{X}{C_o[(1 - \rho_g/\rho_f)X + \rho_g/\rho_f] + \rho_g U_{gj} A/W} \tag{D.17}$$

or

$$\alpha = \frac{X}{aX + b} \tag{D.18}$$

where

$$a = C_o(1 - \rho_g/\rho_f) \tag{D.19}$$

and

$$b = C_o(\rho_g/\rho_f) + \rho_g U_{gj} A/W \tag{D.20}$$

Substitute Eq. (D.18) into (D.16), obtain a relation for two-phase density in terms of local flow quality, as:

$$\rho_{2p} = \rho_f + \frac{\rho_{fg} X}{aX + b} \tag{D.21}$$

The hydrostatic head of the two-phase section is determined by integrating Eq. (D.21) over the two-phase length. The hydrostatic head of the single-phase section is obtained by integrating Eq. (D.9) over the single-phase length. The overall hydrostatic head in the steam generator tubes is the sum of the two. The results are summarized in Table D.1.

Table D.1. Steam generator hydrostatic head under two-phase conditions.

Case 1: $0 < L_b < \ell$

$$\int_0^{L_b} \rho g ds = g\, \ell^*_{2p} \left\{a_1 X_{in} - a_2 \ln \left(\frac{aX_{in} + b}{b}\right)\right\} \quad (1)$$

$$\int_{L_b}^{L} \rho g ds = \frac{gL_b\{\rho_{out} - \rho_f \exp[-(L - L_b)/\ell^*_{1p}]\}}{\{1 - \exp[-(L - L_b)/\ell^*_{1p}]\}} \quad (2)$$

Case 2: $\ell < L_b < \varepsilon L + \ell$

$$\int_0^{L_b} \rho g ds = g\ell^*_{2p} \left\{a_1 x_{in} \ell/L_b + a_2 \ln \left[\frac{aX_{in} + b}{a(X_{in} - \ell/L_b + b}\right]\right\} \quad (3)$$

$$\int_{L_b}^{L} \rho g ds = g\left\{\rho_f \ell + \frac{(\rho_{out} - \rho_f)}{1 - \exp[-(L - L_b)/\ell^*_{1p}]} \{\ell - \ell^*_{1p} \{\exp[-(\ell + \varepsilon L - L_b)/\ell^*_{1p}] - \exp[-(L - L_b)/\ell^*_{1p}]\}\}\right\} \quad (4)$$

Table D.1. Steam generator hydrostatic head under two-phase conditions (Continued).

Case 3: $\ell + \varepsilon L < L_b < L$

$$\int_0^{L_b} \rho g ds = g \ell_{2p}^* \{a_1 X_{in} [1 - \frac{\ell + \varepsilon L}{L_b}] + a_2 \ln [\frac{a X_{in} [1 - \frac{\ell - \varepsilon L}{L_b}] + b}{b}]\} + (3) \quad (5)$$

$$\int_{L_b}^{L} \rho g ds = g \{\rho_f (L - L_b) + \frac{(\rho_{out} - \rho_f)\{(L - L_b) + \ell_{1p}^* [1 - \exp[-(L - L_b)/\ell_{1p}^*]]\}\}}{1 - \exp[-(L - L_b)/\ell_{1p}^*]} \quad (6)$$

Case 4: $L_b > L$

$$\int_0^{L} \rho g ds = g \ell_{2p}^* \{a_1 [X_{in} (1 - \frac{\ell + \varepsilon L}{L}) - X_{out}] + a_2 \ln [\frac{a X_{in} [1 - \frac{\ell + \varepsilon L}{L}] + b}{A X_{out} + b}]\} \quad (7)$$

$+ (3)$

where

$$a_1 = \rho_f - \rho_{fg}/a$$

$$a_2 = \rho_{fg} b/a^2$$

APPENDIX E

Single-Node Fuel Rod Model

Figure E.1 shows the cross sectional view of a typical PWR fuel rod. The fuel rod consists of three regions: fuel pellet, clad, and gap. In the single-node model the three regions are lumped into one element. The transient heat conduction equation for the one-node fuel rod is the following [M4]:

$$m_R (C_p)_R \frac{dT_R}{dt} = q''_N - U_R(T_R - T_C) \qquad (E.1)$$

Where

T_R = the average fuel rod temperature;

T_C = the average coolant temperature;

U_R = overal heat transfer coefficient;

q''_N = nuclear power per unit outer-surface area.

The rod mass per unit outer-surface area, m_R, is the sum of the volume average mass of the fuel and clad:

$$m_R = m_F + m_{C\ell} \qquad (E.2)$$

where

$$m_F = \frac{\pi r_2^2 \rho_F}{2\pi r_0} \qquad (E.3)$$

and

$$m_{C\ell} = \frac{\pi (r_0^2 - r_1^2) \rho_{C\ell}}{2\pi r_0} \qquad (E.4)$$

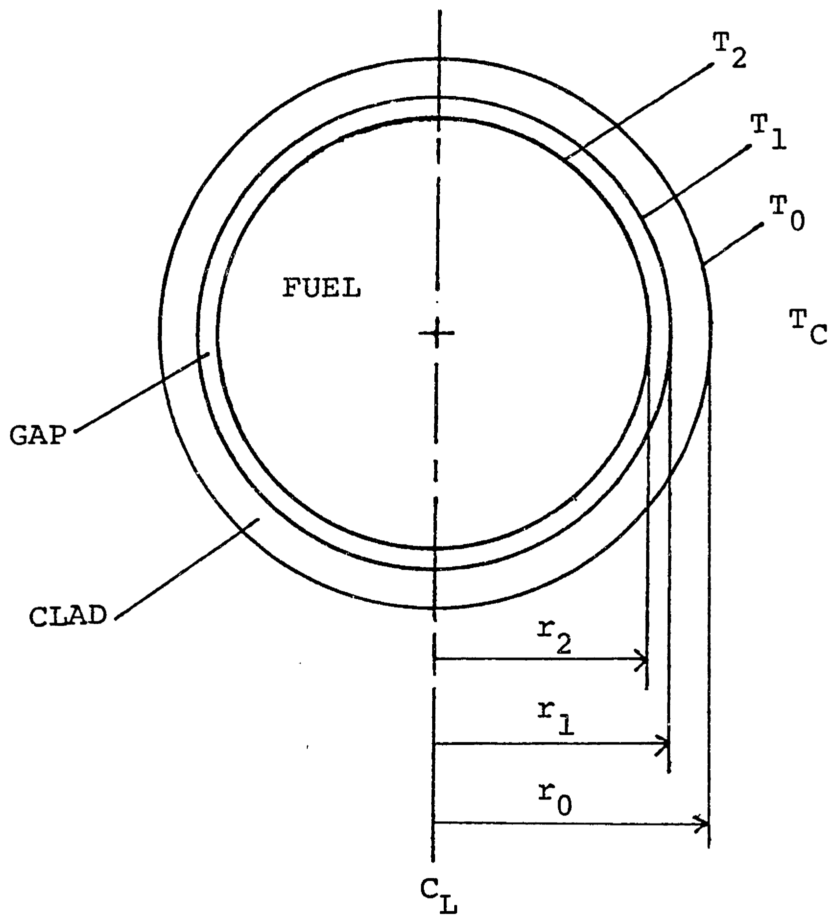


FIGURE E.1 Cross-Sectional View of A Typical Fuel Rod

The average heat capacity, $(C_p)_R$, is a mass average quantity defined by:

$$(C_p)_R = \frac{(mC_p)_F + (mC_p)_{C\ell}}{m_R} \tag{E.5}$$

Assuming a steady-state approximation that the arithmetic average temperatures can be used to represent the energy storage, and thermal conductivities and capacities are uniform in the fuel rod, then the average fuel rod temperature is given by:

$$T_R = \frac{[(mC_p)_{C\ell}(T_0 + T_1) + (mC_p)_F(T_2 - T_3)]}{2(mC_p)_R} \tag{E.6}$$

Define three temperature ratios f_R, f_1, and f_2 as:

$$f_R = (T_R - T_0)/(T_3 - T_0) \tag{E.7}$$

$$f_1 = (T_1 - T_0)/(T_3 - T_0) \tag{E.8}$$

and

$$f_2 = (T_1 - T_0)/(T_3 - T_0) \tag{E.9}$$

Substitute Eqs. (E.7) to (E.9) into Eq. (E.6), obtain the following dimensionless equation:

$$f_R = \frac{[(mC_p)_{C\ell}f_1 + (mC_p)_F(f_2 + 1)]}{2(mC_p)_R} \tag{E.10}$$

By the steady state assumption, the fuel rod surface heat flux is given by the following relationships in terms of heat transfer coefficients and temperature differences in the various regions:

1) Fuel

$$q_0'' = U_F(T_3 - T_2) \tag{E.11}$$

where

$$U_F = 2k_F/r_0 \tag{E.12}$$

2) Gap

$$q_0'' = U_G(T_2 - T_1) \tag{E.13}$$

where

$$U_G = r_1 h_G/r_0 \tag{E.14}$$

3) Clad

$$q_0'' = U_{C\ell}(T_1 - T_0) \tag{E.15}$$

where

$$U_{C\ell} = k_{C\ell}/[r_0 \ell n(r_0/r_1)] \tag{E.16}$$

4) Rod

$$q_0'' = U_T(T_3 - T_0) \tag{E.17}$$

where

$$(1/U_T) = (1/U_{C\ell}) + (1/U_G) + (1/U_F) \tag{E.18}$$

By comparing Eqs. (E.11) to (E.18) with Eqs. (E.7) to (E.9) the following relationship can be found:

$$f_1 = (U_T/U_{C\ell}) \tag{E.19}$$

$$f_2 = (U_T/U_{C\ell}) + (U_T U_G) \tag{E.20}$$

and

$$(1/U_R) = (1/h_R) + (f_R/U_R) \tag{E.21}$$

where h_R is the film heat transfer coefficient at the rod surface.

Having the heat transfer parameters well-defined, we can proceed to solve the fuel rod conduction equation. Re-write Eq. (E.1) as:

$$\frac{dT_R}{dt} = \frac{Q_N/A_C}{(mC_p)_R} + \frac{(T_R - T_C)}{\tau_R} \tag{E.22}$$

where Q_N is the total nuclear power deposited in the fuel, and A_R is the total heat transfer area of the fuel rods, and τ_R is the time constant of the fuel rod:

$$\tau_R = (mC_p)_R/U_R \tag{E.23}$$

Finite-difference Eq. (E.22) semi-implicitly, obtain:

$$\Delta T_R^{n+1} = (1 + \frac{\Delta t}{\tau_R^n})^{-1} [\frac{\Delta t Q_N^{n+1}}{A_R (mC_p)_R^n} - \frac{\Delta t}{\tau_R^n} (T_R^n - T_C^n)] \tag{E.24}$$

The surface heat flux is, therefore, determined by:

$$(q_0'')^{n+1} = U_R^n (T_R^{n+1} - T_C^n) \tag{E.25}$$

hence,

$$(T_3 - T_0)^{n+1} = (q_0'')^{n+1}/U_T^n \tag{E.26}$$

By substituting Eq. (E.26) into Eq. (E.7), the new fuel rod surface temperature is given by the following:

$$T_0^{n+1} = T_R^{n+1} - [(q_0'')^{n+1} f_R^n/U_T^n] \tag{E.27}$$

Finally, the thermal power produced in the core is determined by:

$$Q_{th}^{n+1} = A_R U_R^n (T_R^{n+1} - T_C^n) \tag{E.28}$$

The computational procedure is the following:

1) Determine m_R from Eq. (E.2) and $(C_p)_R$ from Eq. (E.3);
2) Determine the heat transfer coefficients U_F, $U_{C\ell}$, U_G, and U_T from Eqs. (E.12), (E.14), (E.16), and (E.18), respectively;
3) Determine f_1 from Eq. (E.19), f_2 from Eq. (E.20), and f_R from Eq. (E.10), then U_R from Eq. (E.21);
4) Determine T_R from Eq. (E.24);
5) Determine the surface heat flux from Eq. (E.25);
6) Determine the surface termperature T_0 from Eq. (E.27);
7) Determine the thermal power from Eq. (E.28).

APPENDIX F

Wall Conduction Parameters

As discussed in Chapter 3, the transient wall conduction is characterized by the thermal-capacity equivalent length L_c and the thermal-conductivity equivalent length L_k. This appendix discusses a procedure in which these parameters are chosen for a particular transient in consideration. The transient heat conduction is given by Eq. (3.17), or:

$$(\rho C_p)_w \; L_c \frac{d\langle T_w\rangle}{dt} = q'' \tag{F.1}$$

The surface heat flux can be approximated by the following (Eq. (3.18)):

$$q'' = \frac{T_v - \langle T_w\rangle}{L_k/k_w + 1/h_{wc}} \tag{F.2}$$

where

T_v = vapor temperature, and

h_{wc} = film condensation coefficient.

The boundary conditions for Eq. (F.1) are that wall is adiabatic at the outer surface, and vapor temperature is a linear function of time, that is:

$$T_v = a\,t + T_o \tag{F.3}$$

where

a = time coefficient (1/s), and

T_o = initial temperature.

Eq. (F.1) is solved by a finite-difference method as described below. substituting Eq. (F.2) into Eq. (F.1) to obtain:

$$\langle T_w \rangle^{n+1} = \langle T_w \rangle^n + \frac{\frac{\Delta t}{\tau_w^n} \left(T_v^n - \langle T_w \rangle^n \right)}{1 + \Delta t / \tau_w^n} \qquad \text{(F.4)}$$

where

$$\tau_w = \frac{(\rho C_p)_w \, L_w}{L_k / k_w + 1 / h_{wc}} \qquad \text{(F.5)}$$

The wall heat flux is then obtained by substituting Eq. (F.4) into Eq. (F.2), as:

$$(q'')^{n+1} = \frac{\left(T_v^{n+1} - \langle T_w \rangle^{n+1} \right)}{L_k / k_w + 1 / h_{wc}} \qquad \text{(F.6)}$$

The solution given by Eq. (F.6) can be compared with the analytical solution given as the following. The wall temperature as a function of space and time in an infinite slab with zero initial temperature and adiabatic condition at x=0, and T_{wi}=a at x=L, is given by the infinite summation as [D2]:

$$T_w(x,t) = \frac{2}{L} \sum_{n=0}^{\infty} \exp[-\alpha_w(2n + 1)^2\pi^2 t/4L^2] \tag{F.7}$$

$$\cos\left[\frac{(2n + 1)\pi x}{2L}\right]\left[\frac{(2n + 1)\pi\alpha_w(-1)^n}{2L}\right]$$

$$\int_0^t \exp[\alpha_w(2n + 1)^2 \pi x/4L^2]\phi(\lambda)\, d\lambda \quad ; \quad 0 < X < L$$

where

$$\alpha_w = k_w/(\rho C_p)_w \tag{F.8}$$

and

$$\phi(t) = a\ t$$

The wall heat flux at x=L can be approximated by:

$$q'' \simeq -k_w \frac{\partial T_w}{\partial x}\Bigg|_{x=0.95L} \tag{F.9}$$

Substituting Eq. (F.7) into Eq. (F.9) yield:

$$q'' \simeq \frac{-2ak_w}{\alpha_w L} \sum_{n=0}^{\infty} \frac{(-1)^n}{\alpha_w \beta^2} \sin(0.95\ L\beta)$$

$$[\exp(-\alpha_w\beta^2 t) + \alpha_w\beta^2 - 1] \tag{F.10}$$

where

$$\beta = (2n+1)\pi/2L$$

Therefore, the procedure is the following:

1. Assuming vapor temperature increases linearly

with time, and approximate the coefficeint a in Eq. (F.3) by:

$$\alpha_w = \frac{\Delta P}{\Delta t} \left(\frac{\partial T_v}{\partial p}\right) \qquad (F.11)$$

2. Substitute Eq. (F.11) into Eq. (F.10) and solve for the heat flux history;
3. Choose a pair of L_c and L_k and substitute into Eqs. (F.4) and (F.6), compair q" calculated by Eq. (F.6) with the result obtained in Step 2;
4. Repeat Step 3 until satisfactory match is obtained.

The procedure is used to determine L_c's and L_k's for the MIT pressurizer and Connecticut Yankee pressurizer transient analyses of Chapter 3. Figures F.1 and F.2 illustrate the heat flux histories for the two cases.

For a pressure vessel with a short conduction time constant compared with the duration of transient, it can be shown that $L_k = L/3$. The derivation is given below. The asyntotic solution for a transient conduction equation,

$$(\rho C_p)_w L \frac{\partial T_w}{\partial t} = k_w \frac{\partial^2 T_w}{\partial x^2} \qquad (F.12)$$

is given by:

$$T_w(x,t) = f(x) + a\,t \qquad (F.13)$$

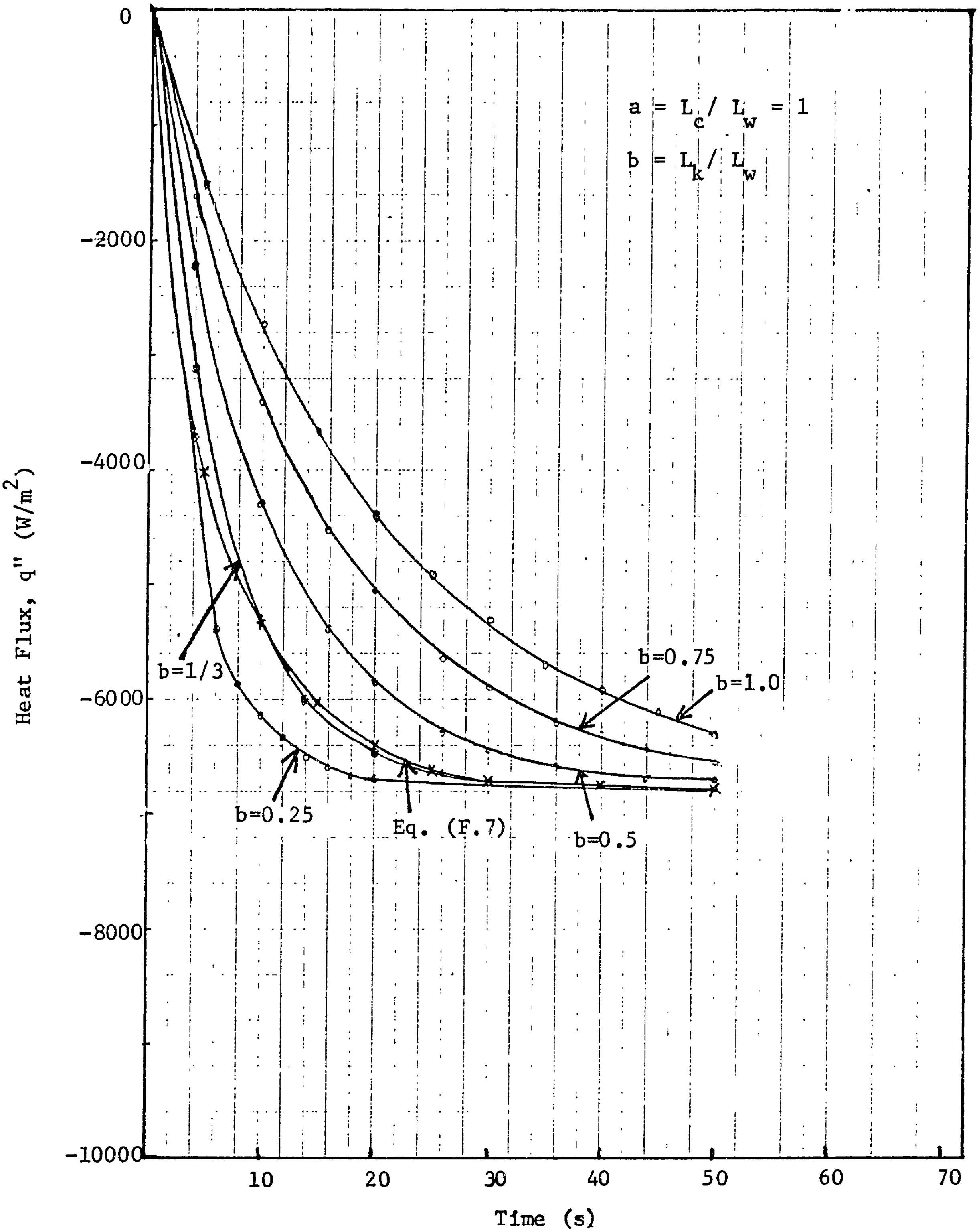


FIGURE F.1 MIT Pressurizer Experiment I01
Wall Heat Flux vs Time.

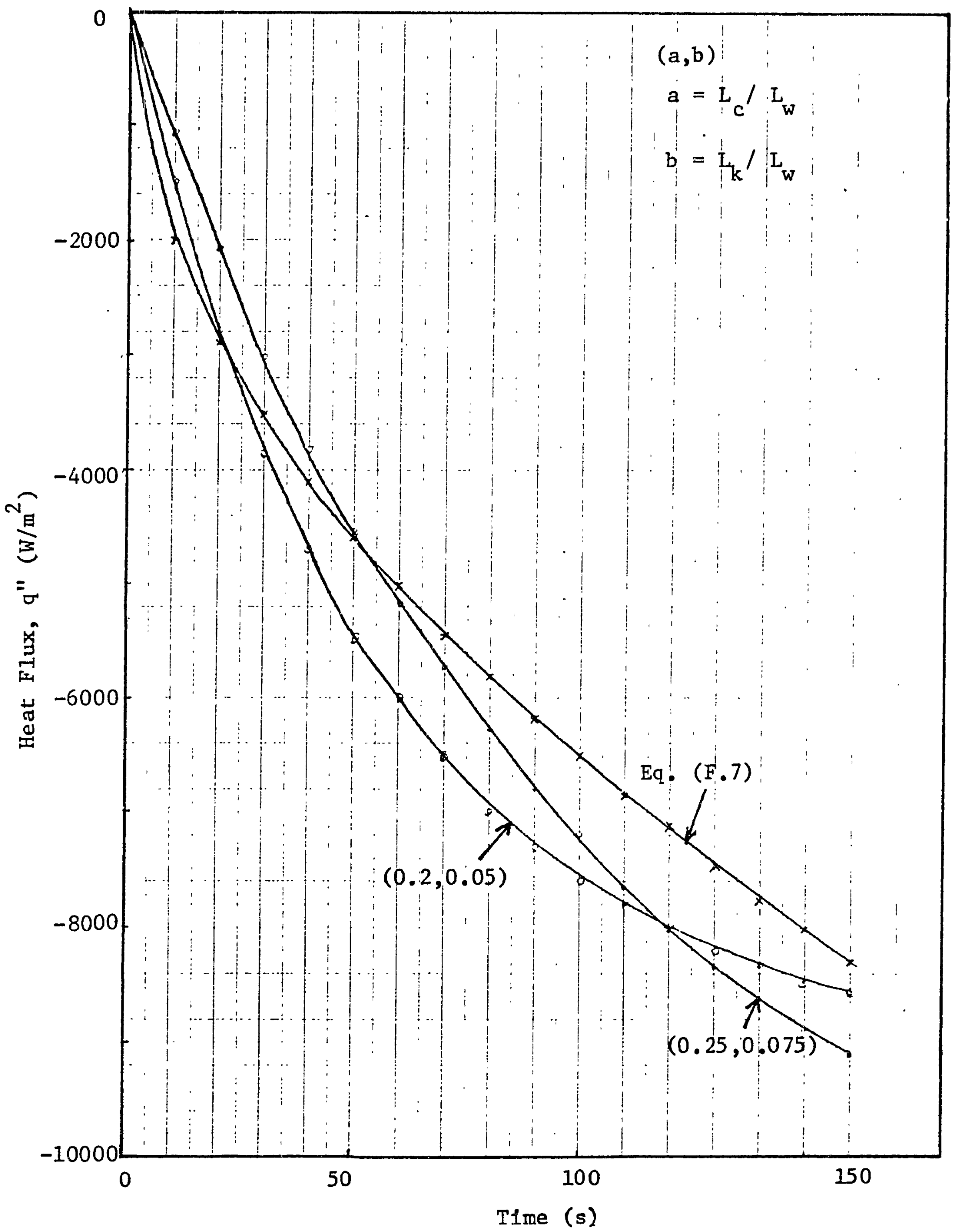


FIGURE F.2 Connecticut Yankee Transient Pressurizer Wall Heat Flux vs Time.

with boundary conditions:

$$-k_w \frac{\partial T_w}{\partial x}\bigg|_{x=L} = 0 \tag{F.14}$$

and

$$-k_w \frac{\partial T_w}{\partial x}\bigg|_{x=0} = q'' = h_{wc}\left[T_v - T_w(0,t)\right] \tag{F.15}$$

where

$$T_v = T_{vo} + a\ t \tag{F.16}$$

Substitute Eq. (F.13) into Eq. (F.12) to obtain:

$$(\rho C_p)_w\ L\ a = k_w\ f'' \tag{F.17}$$

Integrate Eq. (F.17) tvice to get:

$$f(x) = (\rho C_p/k)_w L\ a\ x^2/2 + C_1 x + C_0 \tag{F.18}$$

The integration constant C_1 is determined by substituting Eq. (F.17) into Eqs. (F.14) and (F.15), as:

$$C_1 = -\left(\frac{\rho C_p}{k}\right)_w aL^2 = -q''/k_w \tag{F.19}$$

Let

$$T_w(0,t) = C_0 + a\ t \tag{F.20}$$

Combine Eqs. (F.20) and (F.15), obtain:

$$C_0 = T_v(t) - a\ t - q''/h_{wc} \tag{F.21}$$

Substitute Eqs. (F.19) and (F.21) into (F.18), obtain:

$$T_w = q'' \left(\frac{x^2}{2Lk_w} - \frac{x}{k_w} - \frac{1}{h_{wc}}\right) + T_v(t) \tag{F.22}$$

The average temperature is then given by:

$$\langle T_w(t) \rangle = -q'' \left(\frac{L}{3k_w} + \frac{1}{h_{wc}}\right) + T_v(t) \tag{F.23}$$

Comparing Eqs. (F.2) and (F.23) give:

$$q'' = \frac{T_v - \langle T_w \rangle}{\left(\frac{L}{3k_w} + \frac{1}{h_{wc}}\right)} = \frac{T_v - \langle T_w \rangle}{\left(\frac{L_k}{k_w} + \frac{1}{h_{wc}}\right)} \tag{F.24}$$

therefore, $L_k = L/3$.

APPENDIX G

Pump Characteristic Curves

In order to solve the momentum and pump-rotor speed equations, relations for the pressure head delivered by the pump and the torque on the rotor must be known. These two relations are usually provided by the pump manufacturers in terms of a set of dimensionless characteristic curves. These curves are obtained by testing the pump at various conditions. The test data are represented in the head-flow and torque-flow planes in terms of dimensionless ratios: [R7]

$$h/a^2 = F(\nu/a) \tag{G.1}$$

and

$$b/a^2 = G(\nu/a) \tag{G.2}$$

where

$a = \omega/\omega_R$, rotor speed ratio;

$b = T/T_R$, break-horse-power torque ratio;

$h = H/H_R$, pump head ratio;

$\nu = Q/Q_R$, volumetric discharge flow ratio.

A set of pump characteristic curves usually consist of four curves. In each dimensionless plane, one curve for positive rotor speed and one for negative speed are provided. In the present model, in case of positive speed and flow, the characteristic curves for the Maine Yankee reactor coolant pump are used as the representative curves.

The curves are third order polynomial fits using the information given in the FSAR [F1] and provided by the Yankee Atomic Electric Company [Y1]. The head capacity curve, Eq. (G.1), is given by the following relations:

$$(h/a^2) = 1.80 - 0.30(\nu/a) + 0.35(\nu/a^2) - 0.85(\nu/a^3) \quad (G.3)$$

while the break-horse-power torque curve, Eq. (G.2), is given by the following:

$$(b/a^2) = 1.37 - 1.28(\nu/a) + 1.61(\nu/a^2) - 0.70(\nu/a)^3 \quad (G.4)$$

Using speed ratio as parameter, Eqs. (G.3) and (G.4) are plotted in Figures G.1 and G.2. As shown in the figures, Eqs. (G.3) and (G.4) are extrapolated into the reverse flow and negative head regions.

For zero rotor speed, the pump head is given by the following relation, as [W3]:

$$h = -4.181 \times 10^{-3} \left|\nu\right| \nu \quad (G.5)$$

In case of negative rotor speed, the representative characteristic curves for a centrifugal pump given in Rust [R7] are used. The polynomial fits of the head capacity and torque curves are given respectively as the following:

$$(h/a^2) = 0.50 + 0.51(\nu/a) - 0.26(\nu/a^2) + 0.25(\nu/a^3) \quad (G.6)$$

and

$$(b/a^2) = -0.65 + 1.90(\nu/a) - 1.28(\nu/a^2) + 0.54(\nu/a^3) \quad (G.7)$$

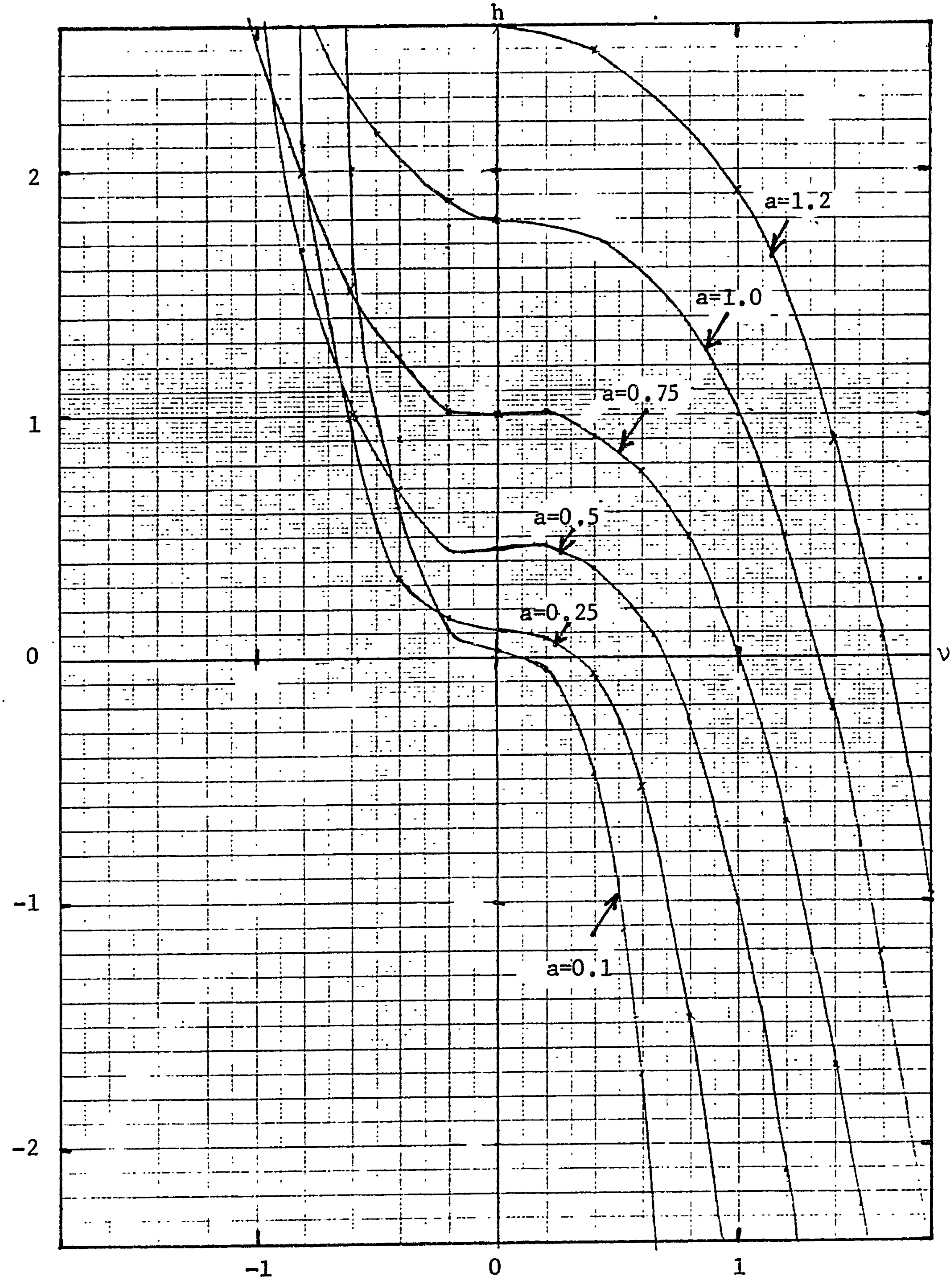


FIGURE G.1 Maine Yankee Pump Head-Capacity Curves (a>0).

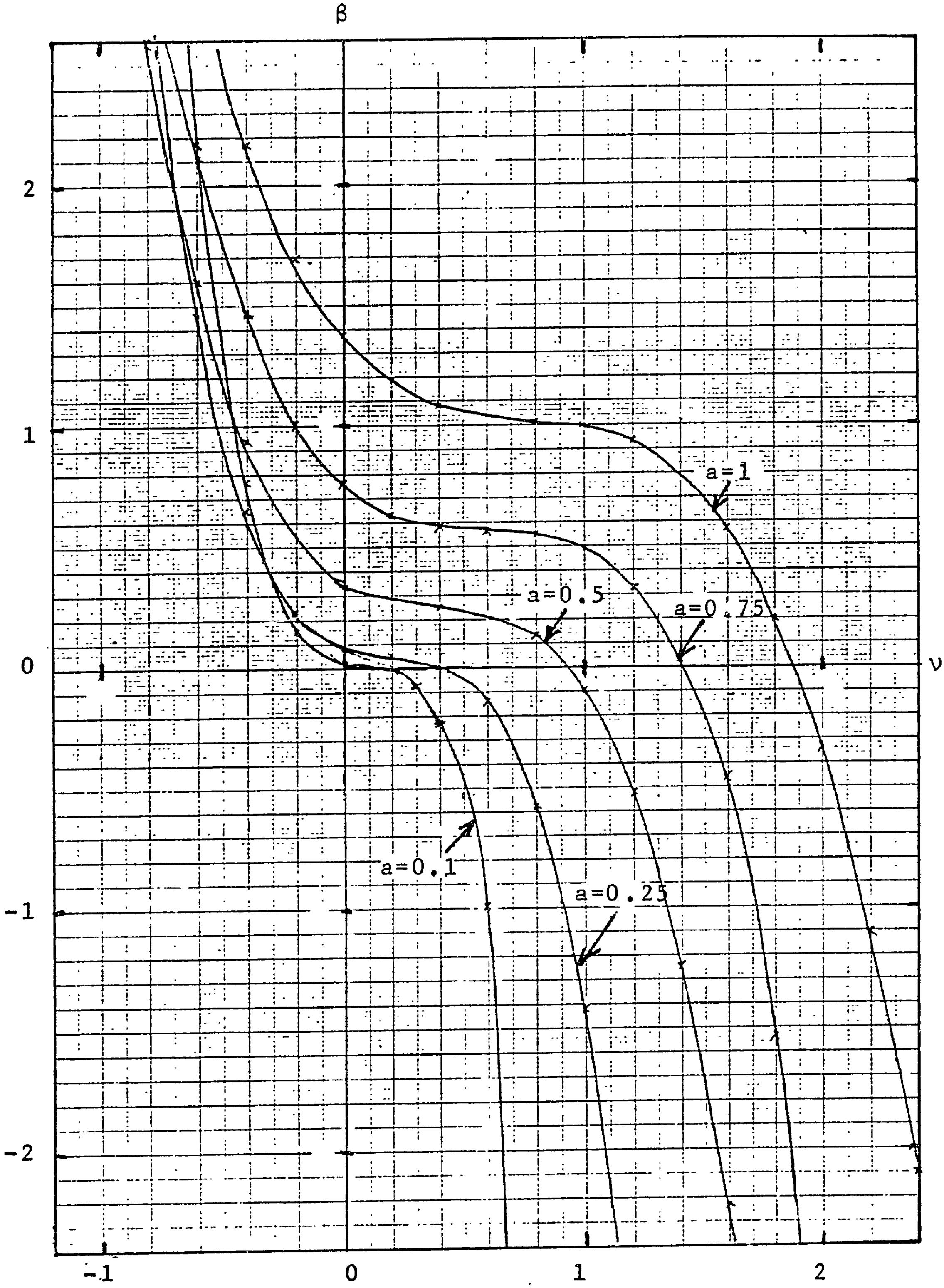


FIGURE G.2 Maine Yankee Pump Torque-Capacity Curves (a>0).

Eqs. (G.6) and (G.7) are plotted in Figures G.3 and G.4 for various rotor speed ratio.

To avoid discontinuity at zero rotor speed, linear interpolation is used between Eqs. (G.3) and (G.5) and also between Eqs. (G.6) and (G.5) to determine the pump head for $-0.1 <= a <= 0.1$. Similarly, linear interpolation is performed between Eqs. (G.4) and (G.7) to determine the break-horse-power torque for $-0.1 <= a <= 0.1$.

To ensure that the second law of thermodynamics is not violated during the interpolation, the following inequality must be kept:

$$\omega T_b > \frac{W \Delta P_{pump}}{\rho} \qquad (G.8)$$

If the second low is violated, the following equation is used to determine the pump torque instead of the linear interpolation, except when $\omega = 0$:

$$T_b = \frac{W \Delta P_{pump}}{0.95 \omega \rho} \qquad (G.9)$$

The complete set of pump characteristic curves, Eqs. (G.3) to (G.7) can be represented in the form of so-called synoptic curves, as illustrated in Figure G.5, which is known as the Karman-Knapp circle diagram. In Figure G.5, four quadrants of pump operation can be identified and defined in Table G.1. During a coastdown transient, the pump will pass from the normal pumping region, Quadrant I,

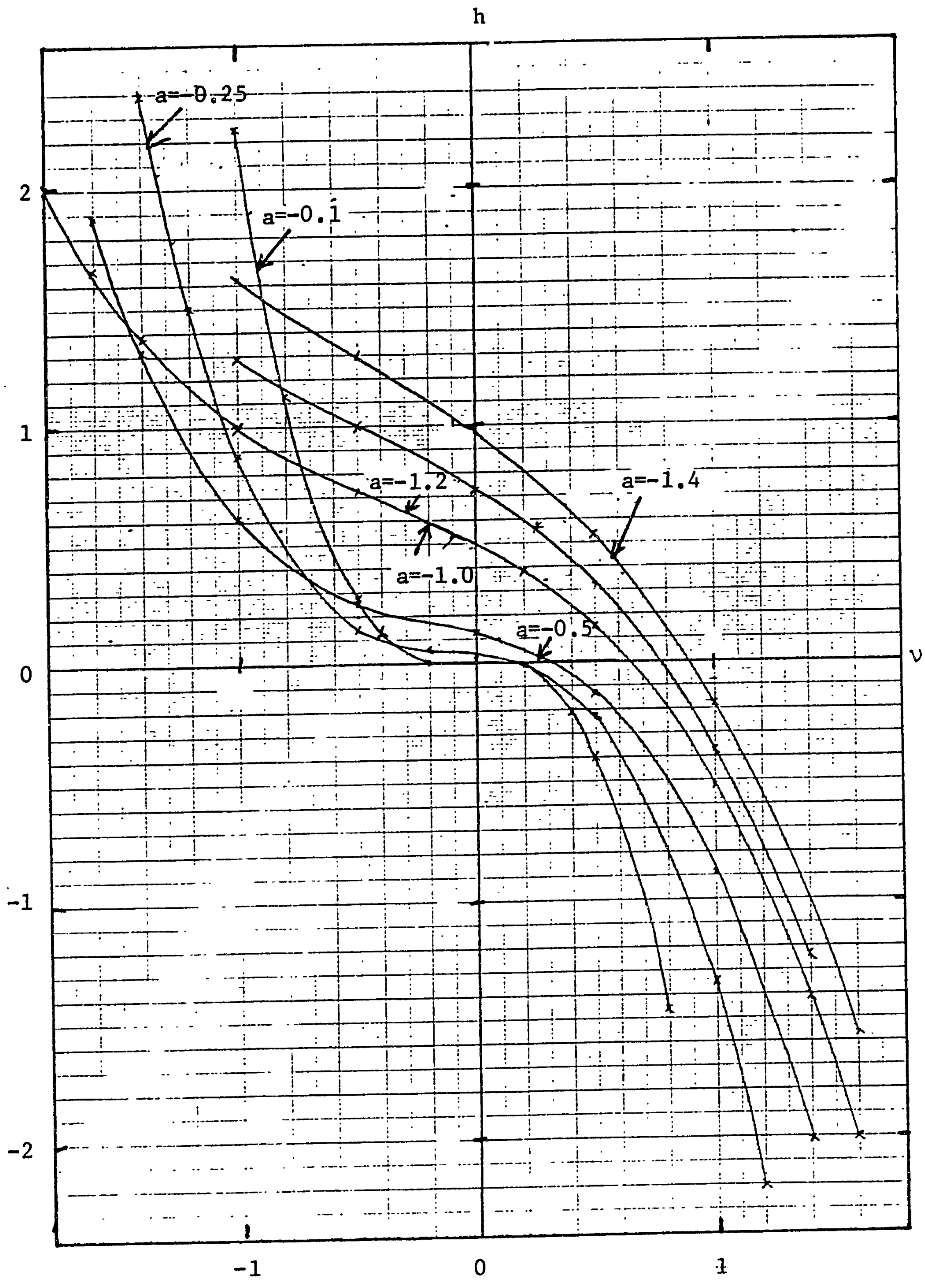


FIGURE G.3 Pump Head-Capacity Curves for a<0 (adopted from [R7]).

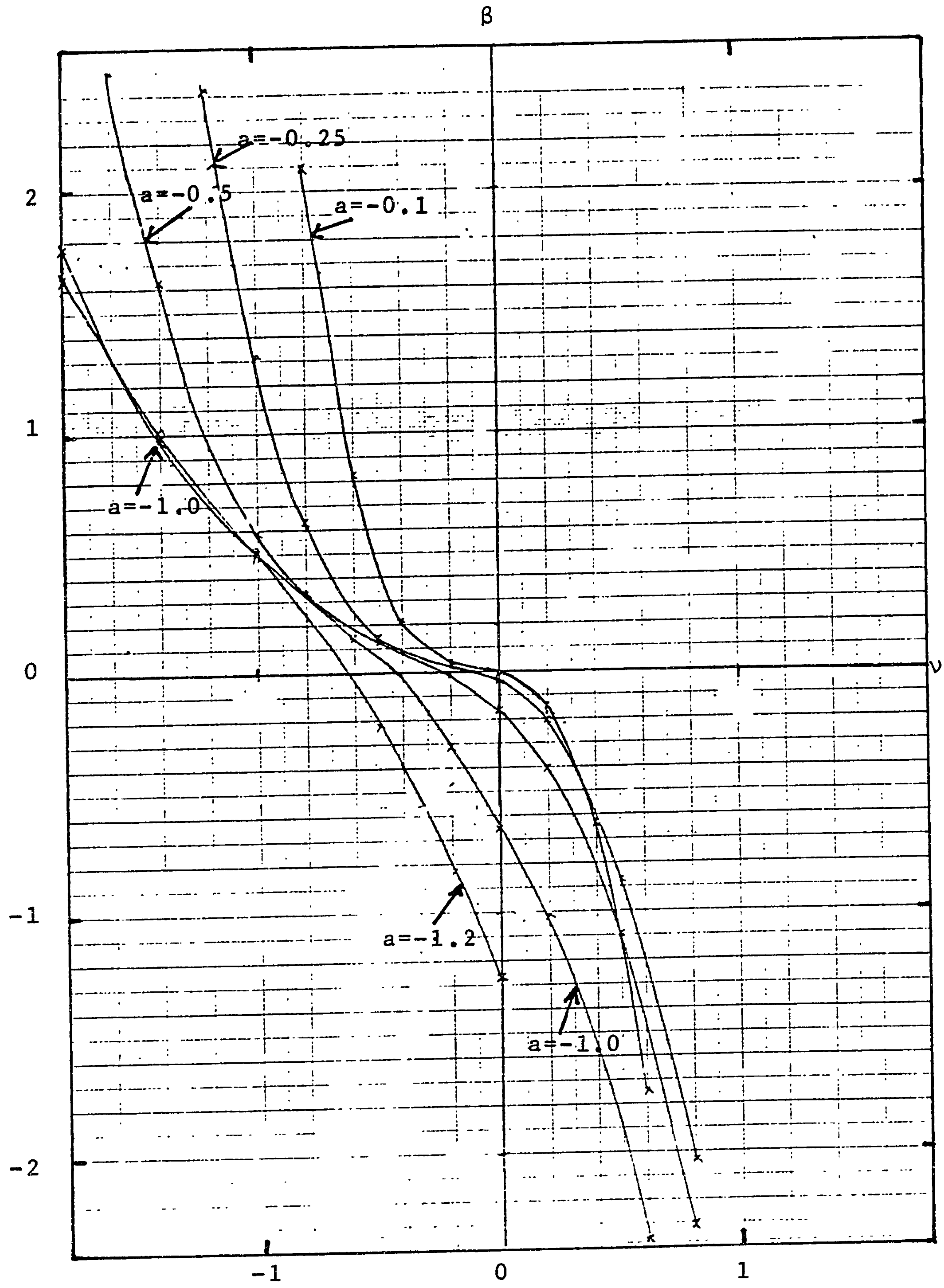


FIGURE G.4 Pump Torque-Capacity Curves
for a<0 (adopted from [R7]).

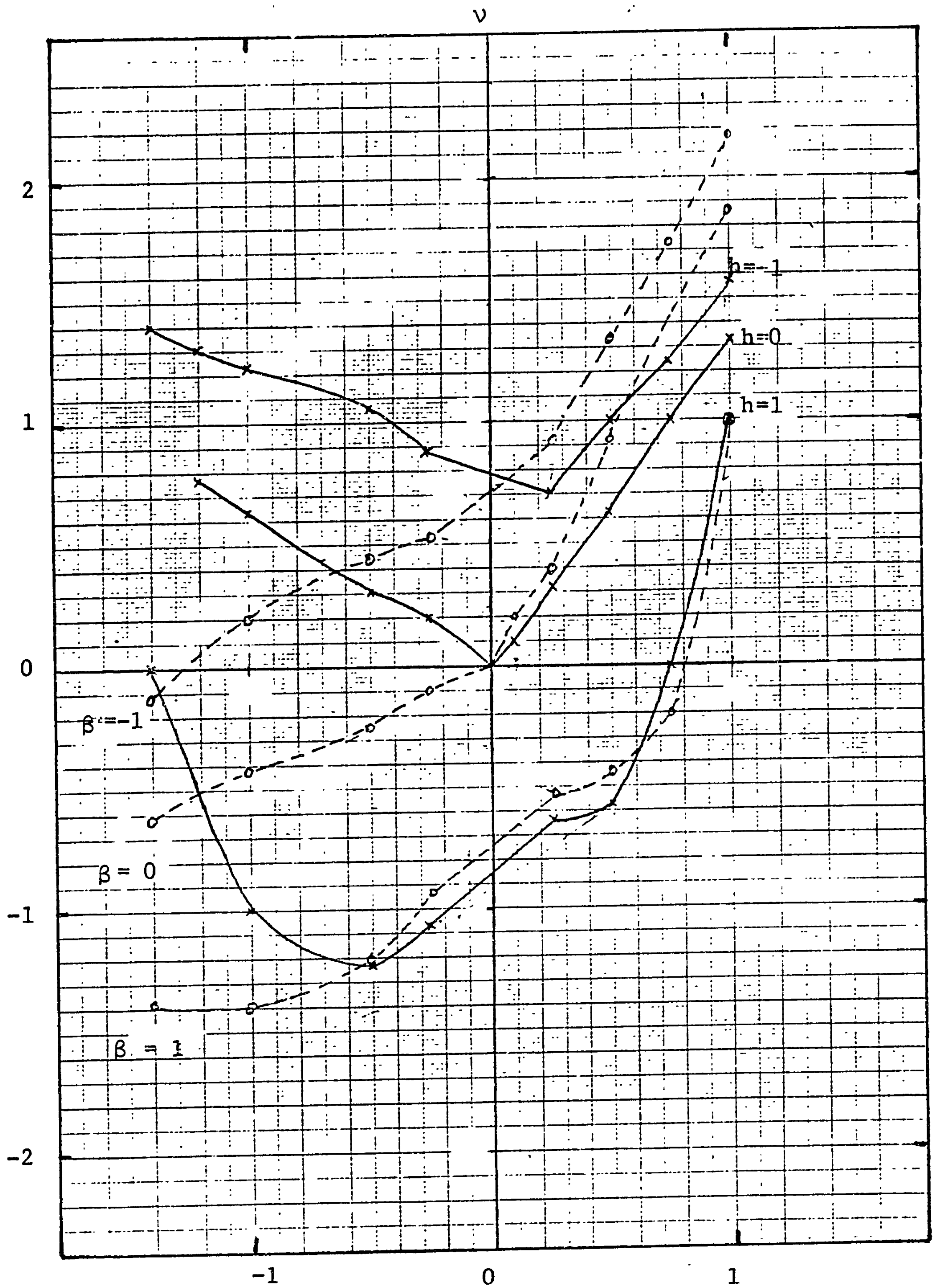


FIGURE G.5 Karman-Knapp Circle Diagram.

through region of reverse flow and positive rotation (energy dissipation), Quadrant IV, to turbine operation region, Quadrant III, for which both flow and speed are reversed. In most of reactor coolant pumps, the rotor shaft is equipped with anti-reverse ratchet, therefore, Quadrant IV is avoided [R7].

Finally, the net pump suction head is determined by the following equation [W3]:

$$(h/a^2) = 1.15(\nu/a^2)^2 - 0.149 \tag{G.10}$$

If the pump head calculated in Eqs. (G.3) or (G.6) is below that in Eq. (G.10), the pump is tripped automatically to prevent it from operation.

TABLE G.1 Quadrants of Pump Operation

Quadrant	v	a	Definition
I	> 0	> 0	Normal Pumping
II	> 0	< 0	Turbine Pumping
III	< 0	< 0	Normal Turbine
IV	< 0	> 0	Energy Dissipation

APPENDIX H

Validation Inputs

page

```
Maine Yankee Natural Circulation
1,3,3,3
700,2,2.2,35
0
1.386e7,546.6,4.442,125.66,6060,5.061e6
2.4e9,6060.6,1.386e7,584.8,555.2,25.7
4.97,15.0,0.569,0.72,1.104,0.72,0.569,3.42,1.73,6.16
1.355e-2,4.37,0.851,0.87,1.661e-2,0.87,0.851,4.37,0.26,5.64
23.08,50.88,3.63,4.79,9.63,4.79,11.53,2.53,8.1,38.59
3.48,1.99,6.385,0.83,15.91,0.83,17.77,0.74,4.69,3.38
6.93,1.74,2.0,1.0
3.48,1.118e-2,9.86e-3,9.703e-3,36352.,9.250e4,2.439e4,
0.92,0.0,5.678e3,4.442e3
1.905e-2,1.661e-2,5703,15.91,7.06,6.2
42.475,22.653,1.214,1.004,9.982,0.124,1.5,0.2,0.333,0.083
30.0,3.39,1.5e5,1.35e6,1.5686e7,1.35e7,1.5341e7,1.35e7,
6.32e-2,11.4,2.e4
4.56e-3,1.644e7,1.713e7,1.731e7,1.748e7
1.86,2.77e-2,15.0,1.0,1.6203e7,1.5513e7
0,0,0
0,0,0,0
3.93e4,4214.0,4.01e4,1.e-7,5.1e5,739.0,787.,6060,125.66,0.1
1.8,-0.3,0.35,-0.85
1.37,-1.28,1.61,-0.7
0.14,0.15,54.86,-0.0245,0.184,0.46,0.19,0.475,0.275,0.19,
0.51,0.0235,0.047,1.228e-4,0.32,0.135,1,1,5,0,4
3,0
0,1.0,4,1.0417,10,1.0556
0
```

Maine Yankee Natural Circulation

1. OPERATING CONDITIONS AT 100% POWER

Thermal Power, MW:	2400.0
Pressure, MPa:	13.86
Hot Leg Temp, deg C:	311.7
Cold Leg Temp, deg C:	282.1
Total Mass Flow Rate, kg/s:	0.1818E+05

2. INITIAL CONDITIONS

Thermal Power, %:	35.00
Pressure, MPa:	13.86
Secondary Pressure, MPa:	5.06
Cold Leg Temp, deg C:	273.5
Pressurizer Water Level, m:	4.44
Pump Rotor Speed, rad/s:	125.66
Loop Mass Flow Rate, kg/s:	6060.
Maximum Time Step Size:	2.00

3. REACTOR COOLANT SYSTEM PARAMETERS

Number of Hot Legs: 3
Number of Cold Legs: 3

Component	Volume (m3)	Min Area (m2)	Fl Path (m)	Eq Dia (m)
Core	23.080	4.970	3.480	0.014
RV IP	2.530	3.420	0.740	4.370
RV DC	8.100	1.730	4.690	0.260
RV LP	38.590	6.160	3.380	5.640
RV UP	50.880	15.000	1.990	4.370
Hot Leg	3.630	0.569	6.385	0.851
Cold Leg	11.530	0.569	17.770	0.851
SG Tube	9.630	1.104	15.910	0.017
SG IP	4.790	0.720	0.830	0.870
SG EP	4.790	0.720	0.830	0.870

Elevations

Tube Support Sheet from Core Inlet, m:	6.93
Core Mid-Plane from Core Inlet:	1.74
Suction-Leg U-Bend Height:	2.00
Core Exit to R/V nozzle:	1.00

4. CORE PARAMETERS

No. of Fuel Rods:	36352.0
Fuel Rod Active Length, m:	3.48
Clad ID, m:	0.9860E-02
Clad OD, m:	0.1118E-01
Fuel Pin D, m:	0.9703E-02
UO2 Wt, kg:	0.9250E+05

Zircaloy Wt, kg:	0.2439E+05
Gap Heat Transfer Coef, W/m2-C:	0.5678E+04

5. STEAM GENERATOR PARAMETERS

No. of Tubes:	5703
Avg. Tube Length, m:	15.91
Avg. Tube Height, m:	7.06
Tube ID, m:	0.0166
Tube OD, m:	0.0191
Tube Metal Volume, m3:	6.2000

6. PRESSURIZER PARAMETERS

A. - Vessel -

Volume, m3:	42.47
Inside Radius, m:	1.214
Height, m:	9.98
Wall Thickness, m:	0.124
Water Volume at 100% Power, m3:	22.65
Wall Condensation Multiplier:	1.50
Flashing Suppression Multipler:	0.20
Wall Capacity Length Factor:	0.3330
Wall Conduction Length Factor:	0.0830

B. - Heater -

Water Level Required to Cover Heaters, m:	3.39
Prop. Heater Power, KW:	150.
Prop. Heater High P Set-point, MPa:	15.69
Prop. Heater Low P Set-point, MPa:	13.50
Backup Heater Power, KW:	1350.
Backup Heater High P Set-point, MPa:	15.34
Backup Heater Low P Set-point, MPa:	13.50
Volume Occupied by Heaters, m3:	0.06
Surface Area of Heaters, m2:	11.40
Heater Heat Transfer Coefficient, W/m2-K:	20000.

C. - Spray Valve -

Max. Spray Mass Flow Rate, kg/s:	1.86
Spray Valve Cross-Sectional Area, m2:	0.0277
Elev. of Spray Line from Cold Leg, m:	15.00
Elev. of Pressurizer from Hot Leg, m:	1.00
High Pressure Set-point, MPa:	16.20
Low Pressure Set-point, MPa:	15.51

D. - Relief and Safety Valves -

Max. Valve Opening Area, m2:	0.0046
Relief Valve P Set-point, MPa:	16.44
Safety Valve No. 1 Set-point, MPa:	17.13
Safety Valve No. 2 Set-point, MPa:	17.31
Safety Valve No. 3 Set-point, MPa:	17.48

6. CHARGING AND LETDOWN

Max. Charging Flow Rate, kg/s:	0.00E+00
Max. Letdown Flow Rate, kg/s:	0.00E+00
Charging Fluid Enthalpy, J/kg:	0.00E+00

7. HIGH-PRESSURE-SAFETY-INJECTION SYSTEM

Rated HPSI Mass Flow Rate, kg/s:	0.00
Injection Initiation Pressure, MPa:	0.00
Injection Termination Pressure, MPa:	0.00
Injection Enthalpy, kJ/kg:	0.

8. PUMP

Rated Breakhorse Power Torque, N-m:	0.3930E+05
Rated Electric Torque:	0.4010E+05
Rated Pump Head Capacity, Pa:	0.5100E+06
Rated Fluid Density, kg/m**3:	739.00
Rated Pump Speed, rad/s:	125.66
Rated Mass flow Rate, kg/s:	0.6060E+04
Pump Rotor & Shaft Inertia, kg-m2:	0.4214E+04
Pump Loss Multiplier:	0.10

8.a PUMP HEAD CHARACTERISTIC COEFFICIENTS

hc0 = 1.8000
hc1 = -0.3000
hc2 = 0.3500
hc3 = -0.8500

8.b PUMP TORQUE CHARACTERISTIC COEFFICIENTS

tb0 = 1.3700
tb1 = -1.2800
tb2 = 1.6100
tb3 = -0.7000

9. form loss factors:

Core expansion:	0.1400
Core contraction:	0.1500
Spacer loss coefficient:	54.8600
Spacer loss factor exponent:	-0.0245
Ratio of projected grid cross section to undisturbed flow cross section:	0.1840
R/V upper plenum & exit nozzle:	0.4600
S/G inlet plenum:	0.1900
S/G tube expansion:	0.4750
S/G tube contraction:	0.2750
S/G exit plenum:	0.1900
R/V inlet nozzle and 90 deg turn:	0.5100
Thermal shield (down-comer) expansion:	0.0235
Thermal shield (down-comer) contraction:	0.0470
R/V lower plenum:	0.0001
90 deg bend:	0.3200
Valve:	0.1350
No. of valves in hot leg:	1

No. of valves in cold leg:	1
No. of spacer grids:	5
No. of 90-deg bends in hot leg:	0
No. of 90-deg bends in cold leg:	4

10a. FORCING FUNCTION FOR SECONDARY PRESSURE

Time	Factor
0.0	1.0000
4.0	1.0417
10.0	1.0556

*** PUMP 1 IS TRIPPED AT 0.00 SEC ***

```
San Onefre-2 Natural Circulation (1.5% Power)
1,4,2,4
1000,50,5000,1.5
0.
1.551e7,559.,4.3,124.67,4662.0,6.9e6
3.39e9,4662.0,1.551e7,594.8,562.5,32.3
5.1,0.892,0.894,0.912,1.493,0.469,0.456,0.455,3.09,2.6
0.012,1.067,1.067,1.067,1.661e-2,0.762,0.762,0.762,0.78,2.6
19.45,74.93,3.96,7.93,36.22,3.88,8.79,2.21,31.46,23.19
3.81,2.9,4.46,3.69,18.44,3.45,12.5,0.98,10.15,1.98
7.63,1.905,2.0,1.2
3.81,8.43e-3,8.26e-3,8.25e-3,51212.,1.016e5,2.907e4,
0.92,0.0,5.678e3,5.12e3
1.905e-2,1.661e-2,6890,18.44,8.11,1.784
42.475,22.653,1.214,1.004,9.982,0.124,1.5,0.5,0.333,0.083
30.0,3.39,1.5e5,1.35e6,1.5686e7,1.35e7,1.5341e7,1.35e7,
6.32e-2,11.4,2.e4
4.56e-3,1.644e7,1.713e7,1.731e7,1.748e7
17.0,8.11e-3,15.0,1.0,1.6203e7,1.5513e7
2.839e-2,1.262e-2,1.015e6
0,0,0,0
4.32e4,4214.0,4.43e4,1.e-7,6.9e5,746.0,787.,4662.0,124.67,1.08
1.8,-0.3,0.35,-0.85
1.37,-1.28,1.61,-0.7
0.14,0.15,54.86,-0.0245,0.184,0.7,0.19,0.475,0.275,0.19,
0.41,0.0234,0.047,2.44e-4,0.14,0.134,0,0,5,1,4
0,0
0
```

San Onefre-2 Natural Circulation (1.5% Power)

1. OPERATING CONDITIONS AT 100% POWER

Thermal Power, MW:	3390.0
Pressure, MPa:	15.51
Hot Leg Temp, deg C:	321.7
Cold Leg Temp, deg C:	289.4
Total Mass Flow Rate, kg/s:	0.1865E+05

2. INITIAL CONDITIONS

Thermal Power, %:	1.50
Pressure, MPa:	15.51
Secondary Pressure, MPa:	6.90
Cold Leg Temp, deg C:	285.9
Pressurizer Water Level, m:	4.30
Pump Rotor Speed, rad/s:	124.67
Loop Mass Flow Rate, kg/s:	4662.
Maximum Time Step Size:	50.00

3. REACTOR COOLANT SYSTEM PARAMETERS

Number of Hot Legs: 2
Number of Cold Legs: 4

Component	Volume (m3)	Min Area (m2)	Fl Path (m)	Eq Dia (m)
Core	19.450	5.100	3.810	0.012
RV IP	2.210	0.455	0.980	0.762
RV DC	31.460	3.090	10.150	0.780
RV LP	23.190	2.600	1.980	2.600
RV UP	74.930	0.892	2.900	1.067
Hot Leg	3.960	0.894	4.460	1.067
Cold Leg	8.790	0.456	12.500	0.762
SG Tube	36.220	1.493	18.440	0.017
SG IP	7.930	0.912	3.690	1.067
SG EP	3.880	0.469	3.450	0.762

Elevations

Tube Support Sheet from Core Inlet, m:	7.63
Core Mid-Plane from Core Inlet:	1.91
Suction-Leg U-Bend Height:	2.00
Core Exit to R/V nozzle:	1.20

4. CORE PARAMETERS

No. of Fuel Rods:	51212.0
Fuel Rod Active Length, m:	3.81
Clad ID, m:	0.8260E-02
Clad OD, m:	0.8430E-02
Fuel Pin D, m:	0.8250E-02
UO2 Wt, kg:	0.1016E+06

Zircaloy Wt, kg:	0.2907E+05
Gap Heat Transfer Coef, W/m2-C:	0.5678E+04

5. STEAM GENERATOR PARAMETERS

No. of Tubes:	6890
Avg. Tube Length, m:	18.44
Avg. Tube Height, m:	8.11
Tube ID, m:	0.0166
Tube OD, m:	0.0191
Tube Metal Volume, m3:	1.7840

6. PRESSURIZER PARAMETERS

A. - Vessel -

Volume, m3:	42.47
Inside Radius, m:	1.214
Height, m:	9.98
Wall Thickness, m:	0.124
Water Volume at 100% Power, m3:	22.65
Wall Condensation Multiplier:	1.50
Flashing Suppression Multipler:	0.50
Wall Capacity Length Factor:	0.3330
Wall Conduction Length Factor:	0.0830

B. - Heater -

Water Level Required to Cover Heaters, m:	3.39
Prop. Heater Power, KW:	150.
Prop. Heater High P Set-point, MPa:	15.69
Prop. Heater Low P Set-point, MPa:	13.50
Backup Heater Power, KW:	1350.
Backup Heater High P Set-point, MPa:	15.34
Backup Heater Low P Set-point, MPa:	13.50
Volume Occupied by Heaters, m3:	0.06
Surface Area of Heaters, m2:	11.40
Heater Heat Transfer Coefficient, W/m2-K:	20000.

C. - Spray Valve -

Max. Spray Mass Flow Rate, kg/s:	17.00
Spray Valve Cross-Sectional Area, m2:	0.0081
Elev. of Spray Line from Cold Leg, m:	15.00
Elev. of Pressurizer from Hot Leg, m:	1.00
High Pressure Set-point, MPa:	16.20
Low Pressure Set-point, MPa:	15.51

D. - Relief and Safety Valves -

Max. Valve Opening Area, m2:	0.0046
Relief Valve P Set-point, MPa:	16.44
Safety Valve No. 1 Set-point, MPa:	17.13
Safety Valve No. 2 Set-point, MPa:	17.31
Safety Valve No. 3 Set-point, MPa:	17.48

6. CHARGING AND LETDOWN

Max. Charging Flow Rate, kg/s:	0.28E-01
Max. Letdown Flow Rate, kg/s:	0.13E-01
Charging Fluid Enthalpy, J/kg:	0.10E+07

7. HIGH-PRESSURE-SAFETY-INJECTION SYSTEM

Rated HPSI Mass Flow Rate, kg/s:	0.00
Injection Initiation Pressure, MPa:	0.00
Injection Termination Pressure, MPa:	0.00
Injection Enthalpy, kJ/kg:	0.

8. PUMP

Rated Breakhorse Power Torque, N-m:	0.4320E+05
Rated Electric Torque:	0.4430E+05
Rated Pump Head Capacity, Pa:	0.6900E+06
Rated Fluid Density, kg/m**3:	746.00
Rated Pump Speed, rad/s:	124.67
Rated Mass flow Rate, kg/s:	0.4662E+04
Pump Rotor & Shaft Inertia, kg-m2:	0.4214E+04
Pump Loss Multiplier:	1.08

8.a PUMP HEAD CHARACTERISTIC COEFFICIENTS

hc0 = 1.8000
hc1 = -0.3000
hc2 = 0.3500
hc3 = -0.8500

8.b PUMP TORQUE CHARACTERISTIC COEFFICIENTS

tb0 = 1.3700
tb1 = -1.2800
tb2 = 1.6100
tb3 = -0.7000

9. form loss factors:

Core expansion:	0.1400
Core contraction:	0.1500
Spacer loss coefficient:	54.8600
Spacer loss factor exponent:	-0.0245
Ratio of projected grid cross section to undisturbed flow cross section:	0.1840
R/V upper plenum & exit nozzle:	0.7000
S/G inlet plenum:	0.1900
S/G tube expansion:	0.4750
S/G tube contraction:	0.2750
S/G exit plenum:	0.1900
R/V inlet nozzle and 90 deg turn:	0.4100
Thermal shield (down-comer) expansion:	0.0234
Thermal shield (down-comer) contraction:	0.0470
R/V lower plenum:	0.0002
90 deg bend:	0.1400
Valve:	0.1340
No. of valves in hot leg:	0

No. of valves in cold leg:	0
No. of spacer grids:	5
No. of 90-deg bends in hot leg:	1
No. of 90-deg bends in cold leg:	4

Initial-State Pressure Losses (Pa)

Core friction loss:	38305.
Core entrance & exit losses:	2580.
Spacer form loss:	64924.
R/V exit form loss:	49747.
Hot leg friction loss:	1799.
Hot leg form loss:	9949.
S/G inlet plenum form loss:	13485.
S/G tube friction loss:	333752.
S/G tube form losses:	26212.
S/G exit plenum form loss:	12958.
Cold leg friction loss:	7083.
Cold leg form losses:	38193.
R/V entrance form losses:	27962.
R/V down-comer friction loss:	2103.
R/V down-comer form losses:	1704.
R/V lower plenum form loss:	84850.
Hot-leg stop valve loss:	0.
Cold-leg stop valve loss:	0.

Bouyancy pressure rise:	0.
Total losses in the loop:	443432.
Total losses in the R/V:	272175.

```
 LOFT PUMP COASTDOWN (Test L6-2)
1,2,1,2
200,1.0,2.0,73.5
0
1.475e7,554.5,1.225,235.6,226.75,5.45e6
5.06e7,240.0,1.551e7,577.7,556.7,21.0
0.16,0.396,0.0993,0.1297,0.2337,0.1297,0.0507,0.0507,0.1424,0.1424
3.6e-3,0.94,0.3556,0.4064,0.0126,0.4064,0.254,0.254,0.102,0.612
0.295,0.9,0.351,0.335,0.88,0.127,0.437,0.17,0.536,0.711
1.68,2.44,5.56,2.58,4.55,2.58,9.27,0.41,3.833,1.151
4.5,0.84,1.3,2.06
1.68,0.0107,9.46e-3,9.08e-3,1300.0,1474.2,302.9,0.93,0.,
5.678e3,73.4
1.27e-2,1.26e-2,1845,4.55,2.05,0.223
0.96,0.64,0.42,0.182,2.0,0.124,1.5,0.2,0.333,0.083
30.0,0.2,1.2e4,3.6e4,1.561e7,1.414e7,1.547e7,1.438e7,
0.003,0.38,2.0e4
1.0e-3,1.662e7,1.767e7,1.767e7,1.779e7
1.0,5.07e-4,2.5,0.5,1.569e7,1.551e7
0,0,0
0.25,1.335e7,6.0e6,1.154e6
500.0,1.43,510,1.e-7,3.776e5,746,50.0,226.75,235.6,1.5
1.40,-0.32,0.26,-0.34
0.62,0.03,0.88,-0.53
0.36,0.266,54.86,-0.0245,0.184,0.7,6.5,0.6,0.4,6.5,
0.41,0.4,0.047,2.44e-3,0.14,0.45,2,4,5,3,7
7,0
0,1.0,5,1,20,1.138,50,1.145,100,1.149,150,1.161,200,1.178
0
```

LOFT PUMP COASTDOWN (Test L6-2)

1. OPERATING CONDITIONS AT 100% POWER

Thermal Power, MW:	50.6
Pressure, MPa:	15.51
Hot Leg Temp, deg C:	304.6
Cold Leg Temp, deg C:	283.6
Total Mass Flow Rate, kg/s:	0.4800E+03

2. INITIAL CONDITIONS

Thermal Power, %:	73.50
Pressure, MPa:	14.75
Secondary Pressure, MPa:	5.45
Cold Leg Temp, deg C:	281.4
Pressurizer Water Level, m:	1.22
Pump Rotor Speed, rad/s:	235.60
Loop Mass Flow Rate, kg/s:	227.
Maximum Time Step Size:	1.00

3. REACTOR COOLANT SYSTEM PARAMETERS

Number of Hot Legs:	1
Number of Cold Legs:	2

Component	Volume (m3)	Min Area (m2)	Fl Path (m)	Eq Dia (m)
Core	0.295	0.160	1.680	0.004
RV IP	0.170	0.051	0.410	0.254
RV DC	0.536	0.142	3.833	0.102
RV LP	0.711	0.142	1.151	0.612
RV UP	0.900	0.396	2.440	0.940
Hot Leg	0.351	0.099	5.560	0.356
Cold Leg	0.437	0.051	9.270	0.254
SG Tube	0.880	0.234	4.550	0.013
SG IP	0.335	0.130	2.580	0.406
SG EP	0.127	0.130	2.580	0.406

Elevations

Tube Support Sheet from Core Inlet, m:	4.50
Core Mid-Plane from Core Inlet:	0.84
Suction-Leg U-Bend Height:	1.30
Core Exit to R/V nozzle:	2.06

4. CORE PARAMETERS

No. of Fuel Rods:	1300.0
Fuel Rod Active Length, m:	1.68
Clad ID, m:	0.9460E-02
Clad OD, m:	0.1070E-01
Fuel Pin D, m:	0.9080E-02
U02 Wt, kg:	0.1474E+04

Zircaloy Wt, kg:	0.3029E+03
Gap Heat Transfer Coef, W/m2-C:	0.5678E+04

5. STEAM GENERATOR PARAMETERS

No. of Tubes:	1845
Avg. Tube Length, m:	4.55
Avg. Tube Height, m:	2.05
Tube ID, m:	0.0126
Tube OD, m:	0.0127
Tube Metal Volume, m3:	0.2230

6. PRESSURIZER PARAMETERS

A. - Vessel -

Volume, m3:	0.96
Inside Radius, m:	0.420
Height, m:	2.00
Wall Thickness, m:	0.124
Water Volume at 100% Power, m3:	0.64
Wall Condensation Multiplier:	1.50
Flashing Suppression Multipler:	0.20
Wall Capacity Length Factor:	0.3330
Wall Conduction Length Factor:	0.0830

B. - Heater -

Water Level Required to Cover Heaters, m:	0.20
Prop. Heater Power, KW:	12.
Prop. Heater High P Set-point, MPa:	15.61
Prop. Heater Low P Set-point, MPa:	14.14
Backup Heater Power, KW:	36.
Backup Heater High P Set-point, MPa:	15.47
Backup Heater Low P Set-point, MPa:	14.38
Volume Occupied by Heaters, m3:	0.00
Surface Area of Heaters, m2:	0.38
Heater Heat Transfer Coefficient, W/m2-K:	20000.

C. - Spray Valve -

Max. Spray Mass Flow Rate, kg/s:	1.00
Spray Valve Cross-Sectional Area, m2:	0.0005
Elev. of Spray Line from Cold Leg, m:	2.50
Elev. of Pressurizer from Hot Leg, m:	0.50
High Pressure Set-point, MPa:	15.69
Low Pressure Set-point, MPa:	15.51

D. - Relief and Safety Valves -

Max. Valve Opening Area, m2:	0.0010
Relief Valve P Set-point, MPa:	16.62
Safety Valve No. 1 Set-point, MPa:	17.67
Safety Valve No. 2 Set-point, MPa:	17.67
Safety Valve No. 3 Set-point, MPa:	17.79

6. CHARGING AND LETDOWN

Max. Charging Flow Rate, kg/s:	0.00E+00
Max. Letdown Flow Rate, kg/s:	0.00E+00
Charging Fluid Enthalpy, J/kg:	0.00E+00

7. HIGH-PRESSURE-SAFETY-INJECTION SYSTEM

Rated HPSI Mass Flow Rate, kg/s:	0.25
Injection Initiation Pressure, MPa:	13.35
Injection Termination Pressure, MPa:	6.00
Injection Enthalpy, kJ/kg:	1154.

8. PUMP

Rated Breakhorse Power Torque, N-m:	0.5000E+03
Rated Electric Torque:	0.5100E+03
Rated Pump Head Capacity, Pa:	0.3776E+06
Rated Fluid Density, kg/m**3:	746.00
Rated Pump Speed, rad/s:	235.60
Rated Mass flow Rate, kg/s:	0.2268E+03
Pump Rotor & Shaft Inertia, kg-m2:	0.1430E+01
Pump Loss Multiplier:	1.50

8.a PUMP HEAD CHARACTERISTIC COEFFICIENTS

hc0 = 1.4000
hc1 = -0.3200
hc2 = 0.2600
hc3 = -0.3400

8.b PUMP TORQUE CHARACTERISTIC COEFFICIENTS

tb0 = 0.6200
tb1 = 0.0300
tb2 = 0.8800
tb3 = -0.5300

9. form loss factors:

Core expansion:	0.3600
Core contraction:	0.2660
Spacer loss coefficient:	54.8600
Spacer loss factor exponent:	-0.0245
Ratio of projected grid cross section to undisturbed flow cross section:	0.1840
R/V upper plenum & exit nozzle:	0.7000
S/G inlet plenum:	6.5000
S/G tube expansion:	0.6000
S/G tube contraction:	0.4000
S/G exit plenum:	6.5000
R/V inlet nozzle and 90 deg turn:	0.4100
Thermal shield (down-comer) expansion:	0.4000
Thermal shield (down-comer) contraction:	0.0470
R/V lower plenum:	0.0024
90 deg bend:	0.1400
Valve:	0.4500
No. of valves in hot leg:	2

No. of valves in cold leg:	4
No. of spacer grids:	5
No. of 90-deg bends in hot leg:	3
No. of 90-deg bends in cold leg:	7

10a. FORCING FUNCTION FOR SECONDARY PRESSURE

Time	Factor
0.0	1.0000
5.0	1.0000
20.0	1.1380
50.0	1.1450
100.0	1.1490
150.0	1.1610
200.0	1.1780

```
 LOFT SMALL BREAK (Test L3-7)
1,2,1,2
1000,1.0,36.0,96.84
39.3
1.49e7,556.0,1.1,235.6,240.0,5.52e6
5.06e7,240.0,1.551e7,577.7,556.7,21.0
0.16,0.396,0.0993,0.1297,0.2337,0.1297,0.0507,0.0507,0.1424,0.1424
3.6e-3,0.94,0.3556,0.4064,0.0126,0.4064,0.254,0.254,0.102,0.612
0.295,0.9,0.351,0.335,0.88,0.127,0.437,0.17,0.536,0.711
1.68,2.44,5.56,2.58,4.55,2.58,9.27,0.41,3.833,1.151
4.5,0.84,0,2.06
1.68,0.0107,9.46e-3,9.08e-3,1300.0,1474.2,302.9,0.93,0.,
5.678e3,73.4
1.27e-2,1.26e-2,1845,4.55,2.05,0.223
0.96,0.64,0.42,0.182,2.0,0.124,1.5,0.2,0.333,0.083
30.0,0.2,1.2e4,3.6e4,1.561e7,1.414e7,1.547e7,1.438e7,
0.003,0.38,2.0e4
1.0e-3,1.662e7,1.767e7,1.767e7,1.779e7
1.0,5.07e-4,2.5,0.5,1.569e7,1.551e7
0,0,0
0.25,1.316e7,1.0e6,1.154e6
500.0,1.43,510,1.e-7,3.776e5,746,50.0,240.0,235.6,1.0
1.40,-0.32,0.26,-0.34
0.62,0.03,0.88,-0.53
0.36,0.266,54.86,-0.0245,0.184,0.7,6.5,0.6,0.4,6.5,
0.41,0.4,0.047,2.44e-3,0.14,0.45,2,4,5,3,7
9,0
0,1.0,40,1,60,1.196,120,1.186,240,1.223,320,1.225,400,1.223,
800,1.174,1800,1.129
1
1.32e-5,1.5e5,1.15
```

LOFT SMALL BREAK (Test L3-7)

1. OPERATING CONDITIONS AT 100% POWER

Thermal Power, MW:	50.6
Pressure, MPa:	15.51
Hot Leg Temp, deg C:	304.6
Cold Leg Temp, deg C:	283.6
Total Mass Flow Rate, kg/s:	0.4800E+03

2. INITIAL CONDITIONS

Thermal Power, %:	96.84
Pressure, MPa:	14.90
Secondary Pressure, MPa:	5.52
Cold Leg Temp, deg C:	282.9
Pressurizer Water Level, m:	1.10
Pump Rotor Speed, rad/s:	235.60
Loop Mass Flow Rate, kg/s:	240.
Maximum Time Step Size:	1.00

3. REACTOR COOLANT SYSTEM PARAMETERS

Number of Hot Legs: 1
Number of Cold Legs: 2

Component	Volume (m3)	Min Area (m2)	Fl Path (m)	Eq Dia (m)
Core	0.295	0.160	1.680	0.004
RV IP	0.170	0.051	0.410	0.254
RV DC	0.536	0.142	3.833	0.102
RV LP	0.711	0.142	1.151	0.612
RV UP	0.900	0.396	2.440	0.940
Hot Leg	0.351	0.099	5.560	0.356
Cold Leg	0.437	0.051	9.270	0.254
SG Tube	0.880	0.234	4.550	0.013
SG IP	0.335	0.130	2.580	0.406
SG EP	0.127	0.130	2.580	0.406

Elevations

Tube Support Sheet from Core Inlet, m:	4.50
Core Mid-Plane from Core Inlet:	0.84
Suction-Leg U-Bend Height:	0.00
Core Exit to R/V nozzle:	2.06

4. CORE PARAMETERS

No. of Fuel Rods:	1300.0
Fuel Rod Active Length, m:	1.68
Clad ID, m:	0.9460E-02
Clad OD, m:	0.1070E-01
Fuel Pin D, m:	0.9080E-02
UO2 Wt, kg:	0.1474E+04

Zircaloy Wt, kg:	0.3029E+03
Gap Heat Transfer Coef, W/m2-C:	0.5678E+04

5. STEAM GENERATOR PARAMETERS

No. of Tubes:	1845
Avg. Tube Length, m:	4.55
Avg. Tube Height, m:	2.05
Tube ID, m:	0.0126
Tube OD, m:	0.0127
Tube Metal Volume, m3:	0.2230

6. PRESSURIZER PARAMETERS

A. - Vessel -

Volume, m3:	0.96
Inside Radius, m:	0.420
Height, m:	2.00
Wall Thickness, m:	0.124
Water Volume at 100% Power, m3:	0.64
Wall Condensation Multiplier:	1.50
Flashing Suppression Multipler:	0.20
Wall Capacity Length Factor:	0.3330
Wall Conduction Length Factor:	0.0830

B. - Heater -

Water Level Required to Cover Heaters, m:	0.20
Prop. Heater Power, KW:	12.
Prop. Heater High P Set-point, MPa:	15.61
Prop. Heater Low P Set-point, MPa:	14.14
Backup Heater Power, KW:	36.
Backup Heater High P Set-point, MPa:	15.47
Backup Heater Low P Set-point, MPa:	14.38
Volume Occupied by Heaters, m3:	0.00
Surface Area of Heaters, m2:	0.38
Heater Heat Transfer Coefficient, W/m2-K:	20000.

C. - Spray Valve -

Max. Spray Mass Flow Rate, kg/s:	1.00
Spray Valve Cross-Sectional Area, m2:	0.0005
Elev. of Spray Line from Cold Leg, m:	2.50
Elev. of Pressurizer from Hot Leg, m:	0.50
High Pressure Set-point, MPa:	15.69
Low Pressure Set-point, MPa:	15.51

D. - Relief and Safety Valves -

Max. Valve Opening Area, m2:	0.0010
Relief Valve P Set-point, MPa:	16.62
Safety Valve No. 1 Set-point, MPa:	17.67
Safety Valve No. 2 Set-point, MPa:	17.67
Safety Valve No. 3 Set-point, MPa:	17.79

6. CHARGING AND LETDOWN

Max. Charging Flow Rate, kg/s:	0.00E+00
Max. Letdown Flow Rate, kg/s:	0.00E+00
Charging Fluid Enthalpy, J/kg:	0.00E+00

7. HIGH-PRESSURE-SAFETY-INJECTION SYSTEM

Rated HPSI Mass Flow Rate, kg/s:	0.25
Injection Initiation Pressure, MPa:	13.16
Injection Termination Pressure, MPa:	1.00
Injection Enthalpy, kJ/kg:	1154.

8. PUMP

Rated Breakhorse Power Torque, N-m:	0.5000E+03
Rated Electric Torque:	0.5100E+03
Rated Pump Head Capacity, Pa:	0.3776E+06
Rated Fluid Density, kg/m**3:	746.00
Rated Pump Speed, rad/s:	235.60
Rated Mass flow Rate, kg/s:	0.2400E+03
Pump Rotor & Shaft Inertia, kg-m2:	0.1430E+01
Pump Loss Multiplier:	1.00

8.a PUMP HEAD CHARACTERISTIC COEFFICIENTS

hc0 = 1.4000
hc1 = -0.3200
hc2 = 0.2600
hc3 = -0.3400

8.b PUMP TORQUE CHARACTERISTIC COEFFICIENTS

tb0 = 0.6200
tb1 = 0.0300
tb2 = 0.8800
tb3 = -0.5300

9. form loss factors:

Core expansion:	0.3600
Core contraction:	0.2660
Spacer loss coefficient:	54.8600
Spacer loss factor exponent:	-0.0245
Ratio of projected grid cross section to undisturbed flow cross section:	0.1840
R/V upper plenum & exit nozzle:	0.7000
S/G inlet plenum:	6.5000
S/G tube expansion:	0.6000
S/G tube contraction:	0.4000
S/G exit plenum:	6.5000
R/V inlet nozzle and 90 deg turn:	0.4100
Thermal shield (down-comer) expansion:	0.4000
Thermal shield (down-comer) contraction:	0.0470
R/V lower plenum:	0.0024
90 deg bend:	0.1400
Valve:	0.4500
No. of valves in hot leg:	2

No. of valves in cold leg:	4
No. of spacer grids:	5
No. of 90-deg bends in hot leg:	3
No. of 90-deg bends in cold leg:	7

10a. FORCING FUNCTION FOR SECONDARY PRESSURE

Time	Factor
0.0	1.0000
40.0	1.0000
60.0	1.1960
120.0	1.1860
240.0	1.2230
320.0	1.2250
400.0	1.2230
800.0	1.1740
1800.0	1.1290

11. COLD-LEG BREAK AREAS

Break Down-Stream Pressure, MPa:	0.1500
Critical Mass Flow Rate Multiplier:	1.1500

Loop 1 Break Area, m2: 0.1320E-04

Initial-State Pressure Losses (Pa)	
Core friction loss:	51785.
Core entrance & exit losses:	3923.
Spacer form loss:	50029.
R/V exit form loss:	10855.
Hot leg friction loss:	1845.
Hot leg form loss:	6513.
S/G inlet plenum form loss:	95588.
S/G tube friction loss:	14203.
S/G tube form losses:	3398.
S/G exit plenum form loss:	91670.
Cold leg friction loss:	4115.
Cold leg form losses:	13821.
R/V entrance form losses:	5782.
R/V down-comer friction loss:	2689.
R/V down-comer form losses:	3354.
R/V lower plenum form loss:	562.
Hot-leg stop valve loss:	3489.
Cold-leg stop valve loss:	25385.

Bouyancy pressure rise:	0.
Total losses in the loop:	260028.
Total losses in the R/V:	128979.

APPENDIX I

Code Input Description

APPENDIX I

Code Input Description

The input file for the SPK code includes the following group of data cards. For each card, the FORTRAN variable names and their meanings are given. Except for Card 1, all input variables are read in with free format.

Card 1 ititle(16a4)

- ititle: A short title containing 64 characters to identify the run.

Card 2 ind,nloop,nh,nc

- ind: indicator of system representation; 1 = single loop, 0 = multiple loop.
- nloop: number of loops in the Reactor Coolant System.
- nh: number of hot legs.
- nc: number of cold legs.

Card 3 t,dtmax,tscram,ppow

- t: total simulation time (s).
- dtmax: maximum time step size (s); used only with empty pressurizer.
- tscram: time when the reactor is scrammed (s).
- ppow: initial power level in percent of reactor power (%).

Card 4 trip(m)

trip: time when the primary pump(s) is(are) tripped (s) (m=1, if ind=1; m=nloop, if ind=0).

Card 5 pi1,tcold1,wlevel,wi1,dmi1,pisec1

pi1: initial pressurizer pressure (Pa).

tcold1: initial cold leg temperature (K).

wlevel: initial pressurizer water level (m).

wi1: initial pump rotor speed (rad/s).

dmi1: initial loop mass flow rate (Kg/s).

pisec1: initial secondary pressure (Pa).

Card 6 qi0,dmi0,pi0,thot0,tcold0,delt0

qi0: rated thermal power (W).

dmi0: mass flow rate at full power (kg/s).

pi0: primary pressure at full power (Pa).

thot0: hot leg temperature at full power (K).

tcold0: cold leg temperature at full power (K).

delt0: core temperature rise at full power (K).

Card 7 ar1,ar2,ar3,ar4,ar5,ar6,ar7,ar8,ar9,ar10

ar1: minmum flow area in reactor core (m2).

ar2: minmum flow area in reactor vessel upper plenum.

ar3: minmum flow area in hot leg.

ar4: minmum flow area in steam generator inlet plenum.

ar5: minmum flow area in steam generator tubes.

ar6: minmum flow area in steam generator exit plenum.

ar7: minmum flow area in cold leg.

ar8: minmum flow area in reactor vessel inlet plenum.

ar9: minmum flow area in reactor vessel downcomer.

ar10: minmum flow area in reactor lower plenum.

Card 8 d1,d2,d3,d4,d5,d6,d7,d8,d9,d10

d1: hydraulic diameter in reactor core (m).

d2: hydraulic diameter in reactor vessel upper plenum.

d3: hydraulic diameter in hot leg.

d4: hydraulic diameter in steam generator inlet plenum.

d5: hydraulic diameter in steam generator tube.

d6: hydraulic diameter in steam generator exit plenum.

d7: hydraulic diameter in cold leg.

d8: hydraulic diameter in reactor vessel inlet plenum.

d9: hydraulic diameter in reactor vessel downcomer.

d10: hydraulic diameter in reactor vessel lower plenum.

Card 9 v1,v2,v3,v4,v5,v6,v7,v8,v9,v10

v1: coolant volume in reactor core (m3).

v2: coolant volume in reactor vessel upper plenum.

v3: coolant volume in hot leg.

v4: coolant volume in steam generator inlet plenum.

v5: coolant volume in steam generator tubes.

v6: coolant volume in steam generator exit plenum.

v7: coolant volume in cold leg.

v8: coolant volume in reactor vessel inlet plenum.

v9: coolant volume in reactor vessel downcomer.

v10: coolant volume in reactor vessel lower plenum.

Card 10 z1,z2,z3,z4,z5,z6,z7,z8,z9,z10

z1: flow path length in reactor core (m).

z2: flow path length in reactor vessel upper plenum.

z3: flow path length in hot leg.

z4: flow path length in steam generator inlet plenum.

z5: flow path length in steam generator tube.

z6: flow path length in steam generator exit plenum.

z7: flow path length in cold leg.

z8: flow path length in reactor vessel inlet plenum.

z9: flow path length in reactor vessel downcomer.

z10: flow path length in reactor vessel lower plenum.

Card 11 zs,zc0,zcl,zep

zs: elevation from core inlet to steam generator tube support sheet (m).

zc0: elevation from core inlet to core mid-plane (m).

zcl: height of U-ben in suction leg (m).

zep: elevation from core exit to reactor exit nozzle (m).

Card 12 fuell,dclo,dcli,dfo,rodn,fmass,clmass,ftd, fpuo2,hgap,afuel

fuell: active fuel rod length (m).

dclo: clad O.D. (m).

dcli: clad I.D. (m).

dfo: fuel pin diameter (m).

rodn: no. of fuel rods.

fmass: uranium mass in core (kg).

clmass: cladding mass in core (kg).

ftd: fuel theoretical density faction.

fpuo2: plutonium fraction by volume.

hgap: gap heat transfer coefficient (W/m2-K).

afuel: total fuel heat transfer area (m2).

Card 13 do,di,ntube,tbl,zl,voltm

do: tube outer diameter (m).

di: tube inner diameter (m).

ntube: no. of tubes

tbl: average tube length (m).

zl: average tube height (m).

voltm: tube metal volume (m3).

Card 14 volprz,wvo,r,vt,wall,dw,fwc,ffl

volprz: pressurizer volume (m3).

wv0: water volume at 100% power (m3).

r: pressurizer inside radius (m).

vt: pressurizer top wall volume (m3).

wall: overall height (m).

dw: wall thickness (m).

fwc: wall condensation multiplier.

ffl: flashing suppression multiplier.

Card 15 tau1,wlmin,qp,qb,pph,ppl,pbh,pbl,vh,ah,hh

tau1: heater power time constant (s).

wlmin: min. water level before uncovering heaters (m3).

qp: prop. heater electric power (W).

qb: backup heater electric power (W).

pph: proportional heater high pressure set point (Pa).

ppl: proportional heater low pressure set point (Pa).

pbh: backup heater high pressure set point (Pa).

pbl: backup heater low pressure set point (Pa).

vh: volume of heater banks (m3).

ah: surface area of heaters (m2).

hh: heat transfer coefficient of heaters (W/m2-K).

Card 16 amax,prv,psv1,psv2,psv3

amax: maximum valve opening area (m2).

prv: relief valve pressure set point (Pa).

psv1: safety valve no. 1 pressure set point (Pa).

psv2: safety valve no. 2 pressure set point (Pa).

psv3: safety valve no. 3 pressure set point (Pa).

Card 17 wspmax,asp,hesp,hisp,psh,psl

wspmax: maximum spray mass flow rate (kg/s).

asp: spray line cross-sectional area (m2).

hesp: elevation of spray line top from cold leg (m).

hisp: elevation of pressurizer inlet nozzle from hot leg (m).

psh: spray valve high pressure set point (Pa).

psl: spray valve low pressure set point (Pa).

Card 18 gpmc,gpml,hch0

gpmc: maximum charge flow rate (kg/s).

gpml: maximum let-down flow rate (kg/s).

hch0: charging fluid enthalpy (J/kg).

Card 19 whpsi,psih,psil,hsi

whpsi: rated high-pressure-injection mass flow rate (kg/s).

psih: injection initiation pressure (Pa).

psil: injection termination pressure (Pa).

hsi: high-pressure-injection enthalpy (J/kg).

Card 20 tr,tiner,ter,taue,phcr,ror,twbr,dmirr,wirr,fh

tr: rated brake-horse-power torque (N-m).

tinrt: pump inertia (kg-m2).

ter: rated electric torque (N-m).

taue: electric torque decay time constant (s).

phcr: rate pump head (Pa).

ror: rated density (kg/m3).

twbr: rated windage & bearing torque (N-m).

dmirr: rated mass flow rate (kg/s).

wirr: rated pump rotor speed (rad/s).

fh: pump loss coefficient multiplier.

Card 21 hc0,hc1,hc2,hc3

pump head characteristic coefficients:

hr = hc0 + hc1*x + hc2*x*x + hc3*x*x*x

where

hr = pump head ratio, and

x = flow ratio/speed ratio.

Card 22 tb0,tb1,tb2,tb3

pump torque characteristic coefficients:

tbr = tb0 + tb1*x + tb2*x*x + tb3*x*x*x

where

tbr = pump torque ratio.

Card 23 sk1e,sk1c,cv1,cv2,ep,sk2,sk3,sk5e,sk5c,sk6, sk83,sk9e,sk9c,sk10,sk90,skv,nvh,nvc,ns, nbh,nbc

form loss factors:

sk1e: core expansion.

sk1c: core contraction.

cv1: spacer loss coefficient.

cv2: spacer loss factor exponent.

ep: ratio of projected grid cross section to undisturbed flow cross section.

sk2: R/V upper plenum & exit nozzle.

sk4: S/G inlet plenum.

sk5e: S/G tube expansion.

sk5c: S/G tube contraction.

sk6: S/G exit plenum.

sk8e: R/V inlet nozzle and 90 deg turn.

sk9e: R/V downcomer expansion.

sk9c: R/V downcomer contraction.

sk10: R/V lower plenum.

sk90: 90 deg bend.

skv: stop valve.

nvh: no. of stop valves in each hot leg.

nvc: no. of stop valves in each cold leg.

ns: no. of spacer grids.

nbh: no. of 90 deg bends in each hot leg.

nbc: no. of 90 deg bends in each cold leg.

Card 24 np,nq

np: number of entries in psfac.

nq: number of entries in qfac.

Card 25 psfac(i),tps(i),i=1,np

(not required if np=0).

psfac: secondary pressure multiplier at tps.

tps: time (s).

Card 26 qfac(i),tq(i),i=1,nq

(not required if nq=0).

qfac: nuclear power multiplier at tq.

tq: time (s).

Card 27 ibr

ibr: break indicator;

0 = no, 1 = yes.

Card 28 (arbr(i),i=1,m),pbr,fbr

(not required if ibr=0).

arbr: break area (m2).

pbr: break downstream pressure (Pa).

fbr: critical mass flow rate multiplier.

Maine Yankee Total-Loss-of-Flow Transient

```
1,3,3,3
60,2,2.2,35
0
1.386e7,546.6,4.442,125.66,6060,5.061e6
2.4e9,6060.6,1.386e7,584.8,555.2,25.7
4.97,15.0,0.569,0.72,1.104,0.72,0.569,3.42,1.73,6.16
1.355e-2,4.37,0.851,0.87,1.661e-2,0.87,0.851,4.37,0.26,5.64
23.08,50.88,3.63,4.79,9.63,4.79,11.53,2.53,8.1,38.59
3.48,1.99,6.385,0.83,15.91,0.83,17.77,0.74,4.69,3.38
6.93,1.74,2.0,1.0
3.48,1.118e-2,9.86e-3,9.703e-3,36352.,9.250e4,2.439e4,
0.92,0.0,5.678e3,4.442e3
1.905e-2,1.661e-2,5703,15.91,7.06,6.2
42.475,22.653,1.214,1.004,9.982,0.124,1.5,0.2
30.0,3.39,1.5e5,1.35e6,1.5686e7,1.35e7,1.5341e7,1.35e7,
6.32e-2,11.4,2.e4
4.56e-3,1.644e7,1.713e7,1.731e7,1.748e7
5.19e-2,1.86,2.77e-2,15.0,1.0,1.6203e7,1.5513e7,6.9e5
0,0,0
0,0,0,0
3.93e4,42140,4.01e4,1.e-7,5.1e5,739.0,787.,6060,125.66,0.1
1.8,-0.3,0.35,-0.85
1.37,-1.28,1.61,-0.7
0.14,0.15,54.86,-0.0245,0.184,0.46,0.19,0.475,0.275,0.19,
0.51,0.0235,0.047,1.228e-4,0.32,0.135,1,1,5,0,4
3,0
0,1.0,4,1.0417,10,1.0556
0
```

Maine Yankee Total-Loss-of-Flow Transient

1. OPERATING CONDITIONS AT 100% POWER

Thermal Power, MW:	2400.0
Pressure, MPa:	13.86
Hot Leg Temp, deg C:	311.7
Cold Leg Temp, deg C:	282.1
Total Mass Flow Rate, kg/s:	0.1818E+05

2. INITIAL CONDITIONS

Thermal Power, %:	35.00
Pressure, MPa:	13.86
Secondary Pressure, MPa:	5.06
Cold Leg Temp, deg C:	273.5
Pressurizer Water Level, m:	4.44
Pump Rotor Speed, rad/s:	125.66
Loop Mass Flow Rate, kg/s:	6860.
Maximum Time Step Size:	2.00

3. REACTOR COOLANT SYSTEM PARAMETERS

Number of Hot Legs: 3
Number of Cold Legs: 3

Component	Volume(m3)	Min Area(m2)	Fl Path(m)	Eq Dia(m)
Core	23.000	4.970	3.480	0.014
RV IP	2.530	3.420	0.740	4.370
RV DC	8.100	1.730	4.690	0.260
RV LP	38.590	6.160	3.380	5.640
RV UP	50.880	15.000	1.990	4.370
Hot Leg	3.630	0.569	6.385	0.851
Cold Leg	11.530	0.569	17.770	0.851
SG Tube	9.630	1.104	15.910	0.017
SG IP	4.790	0.720	0.830	0.870
SG EP	4.790	0.720	0.830	0.870

Elevations

Tube Support Sheet from Core Inlet, m:	6.93
Core Mid-Plane from Core Inlet:	1.74
Suction-Leg U-Bend Height:	2.00
Core Exit to R/V nozzle:	1.00

4. CORE PARAMETERS

No. of Fuel Rods:	36352.0
Fuel Rod Active Length, m:	3.48
Clad ID, m:	0.9860E-02
Clad OD, m:	0.1118E-01
Fuel Pin D, m:	0.9703E-02
UO2 Wt, kg:	0.9250E+05
Zircaloy Wt, kg:	0.2439E+05
Gap Heat Transfer Coef, W/m2-C:	0.5678E+04

5. STEAM GENERATOR PARAMETERS

No. of Tubes:	5703
Avg. Tube Length, m:	15.91
Avg. Tube Height, m:	7.06
Tube ID, m:	0.0166

Tube OD, m:	0.0191
Tube Metal Volume, m3:	6.2000

6. PRESSURIZER PARAMETERS

A. - Vessel -

Volume, m3:	42.47
Inside Radius, m:	1.214
Height, m:	9.98
Wall Thickness, m:	0.124
Water Volume at 100% Power, m3:	22.65
Wall Condensation Multiplier:	1.50
Flashing Suppression Multipler:	0.20
Wall Capacity Length Factor:	0.3333
Wall Conduction Length Factor:	0.0833

B. - Heater -

Water Level Required to Cover Heaters, m:	3.39
Prop. Heater Power, KW:	150.
Prop. Heater High P Set-point, MPa:	15.69
Prop. Heater Low P Set-point, MPa:	13.50
Backup Heater Power, KW:	1350.
Backup Heater High P Set-point, MPa:	15.34
Backup Heater Low P Set-point, MPa:	13.50
Volume Occupied by Heaters, m3:	0.06
Surface Area of Heaters, m2:	11.40
Heater Heat Transfer Coefficient, W/m2-K:	20000.

C. - Spray Valve -

Max. Spray Mass Flow Rate, kg/s:	1.86
Spray Valve Cross-Sectional Area, m2:	0.0277
Elev. of Spray Line from Cold Leg, m:	15.00
Elev. of Pressurizer from Hot Leg, m:	1.00
High Pressure Set-point, MPa:	16.20
Low Pressure Set-point, MPa:	15.51

D. - Relief and Safety Valves -

Max. Valve Opening Area, m2:	0.0046
Relief Valve P Set-point, MPa:	16.44
Safety Valve No. 1 Set-point, MPa:	17.13
Safety Valve No. 2 Set-point, MPa:	17.31
Safety Valve No. 3 Set-point, MPa:	17.48

6. CHARGING AND LETDOWN

Max. Charging Flow Rate, kg/s:	0.00E+00
Max. Letdown Flow Rate, kg/s:	0.00E+00
Charging Fluid Enthalpy, J/kg:	0.00E+00

7. HIGH-PRESSURE-SAFETY-INJECTION SYSTEM

Rated HPSI Mass Flow Rate, kg/s:	0.00
Injection Initiation Pressure, MPa:	0.00
Injection Termination Pressure, MPa:	0.00
Injection Enthalpy, kJ/kg:	0.

8. PUMP

Rated Breakhorse Power Torque, N-m:	0.3930E+05
Rated Electric Torque:	0.4010E+05
Rated Pump Head Capacity, Pa:	0.5100E+06
Rated Fluid Density, kg/m**3:	739.00
Rated Pump Speed, rad/s:	125.66

Rated Mass flow Rate, kg/s:	0.6660E+04
Pump Rotor & Shaft Inertia, kg-m2:	0.4214E+04
Pump Loss Multiplier:	0.10

8.a PUMP HEAD CHARACTERISTIC COEFFICIENTS

hc0 = 1.8000
hc1 = -0.3000
hc2 = 0.3500
hc3 = -0.8500

8.b PUMP TORQUE CHARACTERISTIC COEFFICIENTS

tb0 = 1.3700
tb1 = -1.2800
tb2 = 1.6100
tb3 = -0.7000

9. form loss factors:

Core expansion:	0.1400
Core contraction:	0.1500
Spacer loss coefficient:	54.8600
Spacer loss factor exponent:	-0.0245
Ratio of projected grid cross section to undisturbed flow cross section:	0.1840
R/V upper plenum & exit nozzle:	0.4600
S/G inlet plenum:	0.1900
S/G tube expansion:	0.4750
S/G tube contraction:	0.2750
S/G exit plenum:	0.1900
R/V inlet nozzle and 90 deg turn:	0.5100
Thermal shield (down-comer) expansion:	0.0235
Thermal shield (down-comer) contraction:	0.0470
R/V lower plenum:	0.0001
90 deg bend:	0.3200
Valve:	0.1350
No. of valves in hot leg:	1
No. of valves in cold leg:	1
No. of spacer grids:	5
No. of 90-deg bends in hot leg:	0
No. of 90-deg bends in cold leg:	4

10a. FORCING FUNCTION FOR SECONDARY PRESSURE

Time	Factor
0.0	1.0000
4.0	1.0417
10.0	1.0556

*** PUMP 1 IS TRIPPED AT 0.00 SEC ***

Initial-State Pressure Losses (Pa)	
Core friction loss:	29843.
Core entrance & exit losses:	2566.
Spacer form loss:	64608.
R/V exit form loss:	31792.
Hot leg friction loss:	3214.
Hot leg form loss:	0.
S/G inlet plenum form loss:	12856.
S/G tube friction loss:	206318.
S/G tube form losses:	24983.
S/G exit plenum form loss:	12856.
Cold leg friction loss:	8758.
Cold leg form losses:	86608.
R/V entrance form losses:	34508.
R/V down-comer friction loss:	9384.
R/V down-comer form losses:	5839.
R/V lower plenum form loss:	40595.
Hot-leg stop valve loss:	9330.
Cold-leg stop valve loss:	9134.
Bouyancy pressure rise:	0.
Total losses in the loop:	374057.
Total losses in the R/V:	218254.

Time, sec:	0.00	Pressurizer	
Thermal Power, MW:	840.0000	Vapor Temp, C:	336.89
Nuclear Power, MW:	840.0000	Liquid Temp, C:	336.89
Pressure, MPa:	13.8600	Water Level, m:	4.44
Avg Fuel Temp, C:	367.5674	Void Fraction:	0.5599
Secondary Temp, C:	265.6607	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
17454.1680	273.4400	282.8972	282.8972	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	280.0000	5818.0562	125.6600	282.8972	273.4400	273.4400	0.0000	0.0000	0.0000	0.5923E+06	1507.5809

Time, sec:	1.00	Pressurizer	
Thermal Power, MW:	840.0001	Vapor Temp, C:	337.02
Nuclear Power, MW:	840.0000	Liquid Temp, C:	336.83
Pressure, MPa:	13.8735	Water Level, m:	4.45
Avg Fuel Temp, C:	367.5674	Void Fraction:	0.5575
Secondary Temp, C:	266.3864	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
16551.7402	273.4831	282.8470	282.8837	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	266.2263	5517.2466	117.3427	282.8925	273.6948	273.5700	0.0000	0.0000	0.0000	0.5920E+06	1458.4950

Time, sec:	2.06	Pressurizer	
Thermal Power, MW:	838.2638	Vapor Temp, C:	337.32
Nuclear Power, MW:	840.0000	Liquid Temp, C:	336.69
Pressure, MPa:	13.9047	Water Level, m:	4.45
Avg Fuel Temp, C:	367.6203	Void Fraction:	0.5585
Secondary Temp, C:	266.9878	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
15497.6387	273.5654	283.1038	282.9579	0.0000	0.0000	0.0006

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	250.1540	5165.8794	109.6058	282.9278	273.8963	273.7267	0.0000	0.0000	0.0000	0.5043E+06	1441.6853

*** REACTOR IS SCRAMMED AT 3.20 SEC ***

Time, sec:	3.20	Pressurizer	
Thermal Power, MW:	651.2002	Vapor Temp, C:	337.25
Nuclear Power, MW:	50.7559	Liquid Temp, C:	336.72
Pressure, MPa:	13.8968	Water Level, m:	4.45
Avg Fuel Temp, C:	348.0733	Void Fraction:	0.5587
Secondary Temp, C:	267.7072	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
14470.7393	273.6587	282.4500	282.7971	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	230.9124	4823.5796	102.3960	282.8640	274.0337	273.8637	0.0000	0.0000	0.0000	0.4381E+06	1392.0032

Time, sec:	4.42	Pressurizer	
Thermal Power, MW:	504.9370	Vapor Temp, C:	336.91
Nuclear Power, MW:	47.4354	Liquid Temp, C:	336.91
Pressure, MPa:	13.8630	Water Level, m:	4.44
Avg Fuel Temp, C:	332.0036	Void Fraction:	0.5605
Secondary Temp, C:	268.2653	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
13502.2197	273.7473	281.4812	282.3784	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	212.6580	4500.7397	95.6530	282.6268	274.0589	273.9557	0.0000	0.0000	0.0000	0.3826E+06	1319.5930

Time, sec:	5.73	Pressurizer	
Thermal Power, MW:	387.7850	Vapor Temp, C:	336.73
Nuclear Power, MW:	45.0009	Liquid Temp, C:	336.73
Pressure, MPa:	13.8337	Water Level, m:	4.40
Avg Fuel Temp, C:	319.0144	Void Fraction:	0.5650
Secondary Temp, C:	268.4462	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000

Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
12596.6240	273.8125	280.4354	281.7639	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	201.4299	4198.8745	89.3415	282.3744	273.9110	273.9676	0.0000	0.0000	0.0000	0.3345E+06	1238.0919

Time, sec:	7.13	Pressurizer	
Thermal Power, MW:	295.5998	Vapor Temp, C:	336.50
Nuclear Power, MW:	43.2147	Liquid Temp, C:	336.50
Pressure, MPa:	13.7942	Water Level, m:	4.34
Avg Fuel Temp, C:	308.6995	Void Fraction:	0.5711
Secondary Temp, C:	268.6398	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
11751.1514	273.8311	279.4134	281.0283	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	186.5793	3917.0505	83.4341	281.9432	273.6649	273.8918	0.0000	0.0000	0.0000	0.2925E+06	1165.3154

Time, sec:	8.64	Pressurizer	
Thermal Power, MW:	224.5247	Vapor Temp, C:	336.23
Nuclear Power, MW:	41.6292	Liquid Temp, C:	336.23
Pressure, MPa:	13.7480	Water Level, m:	4.27
Avg Fuel Temp, C:	300.6461	Void Fraction:	0.5782
Secondary Temp, C:	268.8469	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
10961.6201	273.7956	278.4618	280.2108	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	169.2940	3653.8733	77.9061	281.3679	273.3739	273.7428	0.0000	0.0000	0.0000	0.2556E+06	1088.6219

Time, sec:	10.26	Pressurizer	
Thermal Power, MW:	170.8252	Vapor Temp, C:	335.93
Nuclear Power, MW:	40.2347	Liquid Temp, C:	335.93
Pressure, MPa:	13.6982	Water Level, m:	4.20
Avg Fuel Temp, C:	294.4555	Void Fraction:	0.5858
Secondary Temp, C:	269.0335	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
10224.0098	273.7055	277.5988	279.3844	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	151.4656	3400.0298	72.7346	280.6943	273.0600	273.5379	0.0000	0.0000	0.0000	0.2234E+06	1009.6301

Time, sec:	11.99	Pressurizer	
Thermal Power, MW:	131.0437	Vapor Temp, C:	335.62
Nuclear Power, MW:	38.9896	Liquid Temp, C:	335.62

Pressure, MPa:	13.6452	Water Level, m:	4.13
Avg Fuel Temp, C:	289.7607	Void Fraction:	0.5939
Secondary Temp, C:	269.0335	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
9535.4453	273.5585	276.8249	278.5744	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	137.1252	3178.4817	67.8974	279.9747	272.6704	273.2707	0.0000	0.0000	0.0000	0.1952E+06	933.6405

Time, sec:	13.85	Pressurizer	
Thermal Power, MW:	102.1452	Vapor Temp, C:	335.30
Nuclear Power, MW:	37.8709	Liquid Temp, C:	335.30
Pressure, MPa:	13.5915	Water Level, m:	4.06
Avg Fuel Temp, C:	286.2358	Void Fraction:	0.6020
Secondary Temp, C:	269.0335	Heater Power, KW:	0.0003
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
8892.2646	273.3584	276.1333	277.8016	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	122.5031	2964.0881	63.3739	279.2356	272.2669	272.9576	0.0000	0.0000	0.0000	0.1705E+06	867.2253

Time, sec:	15.85	Pressurizer	
Thermal Power, MW:	81.5209	Vapor Temp, C:	334.98
Nuclear Power, MW:	36.8617	Liquid Temp, C:	334.98
Pressure, MPa:	13.5387	Water Level, m:	3.98
Avg Fuel Temp, C:	283.6033	Void Fraction:	0.6100
Secondary Temp, C:	269.0335	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
8291.3320	273.1144	275.5154	277.0777	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	100.4287	2763.7773	59.1447	278.5008	271.8802	272.6190	0.0000	0.0000	0.0000	0.1489E+06	807.2099

Time, sec:	18.00	Pressurizer	
Thermal Power, MW:	67.0186	Vapor Temp, C:	334.67
Nuclear Power, MW:	35.9474	Liquid Temp, C:	334.67
Pressure, MPa:	13.4879	Water Level, m:	3.91
Avg Fuel Temp, C:	281.6357	Void Fraction:	0.6176
Secondary Temp, C:	269.0335	Heater Power, KW:	0.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
7729.7534	272.8383	274.9631	276.4000	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	95.3312	2576.5845	55.1910	277.7876	271.5236	272.2733	0.0000	0.0000	0.0000	0.1300E+06	752.3683

Time, sec:	20.30	Pressurizer	
Thermal Power, MW:	56.8927	Vapor Temp, C:	334.42
Nuclear Power, MW:	35.1139	Liquid Temp, C:	334.42
Pressure, MPa:	13.4457	Water Level, m:	3.85
Avg Fuel Temp, C:	280.1535	Void Fraction:	0.6248
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
7204.9316	272.5439	274.4707	275.7953	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	83.4387	2401.6438	51.4954	277.1112	271.2033	271.9355	0.0000	0.0000	0.0000	0.1135E+06	702.1164

Time, sec:	22.77	Pressurizer	
Thermal Power, MW:	49.8406	Vapor Temp, C:	334.18
Nuclear Power, MW:	34.3478	Liquid Temp, C:	334.18
Pressure, MPa:	13.4070	Water Level, m:	3.79
Avg Fuel Temp, C:	279.0207	Void Fraction:	0.6315
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
6714.4829	272.2414	274.0309	275.2373	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	72.7727	2238.1609	48.0415	276.4752	270.9190	271.6140	0.0000	0.0000	0.0000	0.9901E+05	656.1669

Time, sec:	25.43	Pressurizer	
Thermal Power, MW:	44.8908	Vapor Temp, C:	333.97
Nuclear Power, MW:	33.6368	Liquid Temp, C:	333.97
Pressure, MPa:	13.3723	Water Level, m:	3.73
Avg Fuel Temp, C:	278.1366	Void Fraction:	0.6376
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
6256.2139	271.9406	273.6387	274.7318	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	63.3115	2085.4045	44.8136	275.8835	270.6692	271.3148	0.0000	0.0000	0.0000	0.8638E+05	614.0313

Time, sec:	28.28	Pressurizer	
Thermal Power, MW:	41.3471	Vapor Temp, C:	333.78
Nuclear Power, MW:	32.9695	Liquid Temp, C:	333.78
Pressure, MPa:	13.3415	Water Level, m:	3.68
Avg Fuel Temp, C:	277.4295	Void Fraction:	0.6433
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
4380.9966	270.6618	272.2741	272.8935	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	31.2920	1460.3322	31.5907	273.6078	269.8401	270.2090	0.0000	0.0000	0.0000	0.4348E+05	453.0213

Time, sec:	46.05	Pressurizer	
Thermal Power, MW:	32.5528	Vapor Temp, C:	333.21
Nuclear Power, MW:	30.0007	Liquid Temp, C:	333.21
Pressure, MPa:	13.2473	Water Level, m:	3.49
Avg Fuel Temp, C:	275.2850	Void Fraction:	0.6639
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
4077.5186	270.4682	272.1027	272.6446	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	27.2826	1359.1729	29.4459	273.2805	269.7372	270.0598	0.0000	0.0000	0.0000	0.3788E+05	429.7999

Time, sec:	50.44	Pressurizer	
Thermal Power, MW:	31.5032	Vapor Temp, C:	333.16
Nuclear Power, MW:	29.4454	Liquid Temp, C:	333.16
Pressure, MPa:	13.2396	Water Level, m:	3.46
Avg Fuel Temp, C:	275.0174	Void Fraction:	0.6668
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
3794.4805	270.2955	271.9605	272.4295	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	23.8619	1264.8268	27.4432	272.9915	269.6491	269.9304	0.0000	0.0000	0.0000	0.3297E+05	409.3941

Time, sec:	55.16	Pressurizer	
Thermal Power, MW:	30.5503	Vapor Temp, C:	333.13
Nuclear Power, MW:	28.8978	Liquid Temp, C:	333.13
Pressure, MPa:	13.2353	Water Level, m:	3.44
Avg Fuel Temp, C:	274.7863	Void Fraction:	0.6692
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
3530.5977	270.1427	271.8459	272.2464	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	20.9550	1176.8658	25.5736	272.7390	269.5736	269.8183	0.0000	0.0000	0.0000	0.2872E+05	391.7632

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
5828.0918	271.6490	273.2903	274.2762	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	54.9904	1942.6973	41.7974	275.3381	270.4516	271.0412	0.0000	0.0000	0.0000	0.7533E+05	575.4584

Time, sec:	31.34	Pressurizer	
Thermal Power, MW:	38.7276	Vapor Temp, C:	333.62
Nuclear Power, MW:	32.3360	Liquid Temp, C:	333.62
Pressure, MPa:	13.3149	Water Level, m:	3.63
Avg Fuel Temp, C:	276.8499	Void Fraction:	0.6484
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
5428.2236	271.3724	272.9823	273.8674	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	47.7262	1809.4070	38.9794	274.8389	270.2630	270.7944	0.0000	0.0000	0.0000	0.6569E+05	540.2133

Time, sec:	34.63	Pressurizer	
Thermal Power, MW:	36.7118	Vapor Temp, C:	333.48
Nuclear Power, MW:	31.7282	Liquid Temp, C:	333.48
Pressure, MPa:	13.2922	Water Level, m:	3.59
Avg Fuel Temp, C:	276.3642	Void Fraction:	0.6530
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
5054.8413	271.1144	272.7118	273.5025	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	41.4206	1684.9470	36.3467	274.3851	270.1002	270.5743	0.0000	0.0000	0.0000	0.5726E+05	508.1454

Time, sec:	38.17	Pressurizer	
Thermal Power, MW:	35.0910	Vapor Temp, C:	333.37
Nuclear Power, MW:	31.1394	Liquid Temp, C:	333.37
Pressure, MPa:	13.2734	Water Level, m:	3.55
Avg Fuel Temp, C:	275.9503	Void Fraction:	0.6571
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
4706.2837	270.8772	272.4764	273.1787	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	35.9742	1568.7612	33.8875	273.9753	269.9602	270.3797	0.0000	0.0000	0.0000	0.4991E+05	479.1133

Time, sec:	41.97	Pressurizer	
Thermal Power, MW:	33.7325	Vapor Temp, C:	333.28
Nuclear Power, MW:	30.5647	Liquid Temp, C:	333.28
Pressure, MPa:	13.2585	Water Level, m:	3.52
Avg Fuel Temp, C:	275.5938	Void Fraction:	0.6607

Time, sec:	60.24	Pressurizer	
Thermal Power, MW:	29.6752	Vapor Temp, C:	333.13
Nuclear Power, MW:	28.3580	Liquid Temp, C:	333.13
Pressure, MPa:	13.2343	Water Level, m:	3.42
Avg Fuel Temp, C:	274.5883	Void Fraction:	0.6714
Secondary Temp, C:	269.0335	Heater Power, KW:	1500.0000
Charging Mass Flow Rate, Kg/s:	0.0000	Spray Mass Flow Rate:	0.0000
Break Mass Flow Rate:	0.0000	R/V Mass Flow Rate:	0.0000
Safety Injection Mass Flow Rate:	0.0000	S/V Mass Flow Rate:	0.0000

Core Mass Flow	I-P Temp	Core Temp	E-P Temp	I-P Void	Core Void	E-P Void
3284.6638	270.0083	271.7569	272.0933	0.0000	0.0000	0.0000

Loop	S/G Power	Mass Flow	Pump Speed	H-L Temp	S/G Temp	C-L Temp	H-L Void	S/G Void	C-L Void	Pump Hd	Bouy Hd
1	18.4922	1094.8879	23.8284	272.5209	269.5089	269.7216	0.0000	0.0000	0.0000	0.2500E+05	376.8449

APPENDIX J

Code Listing

```
c    *************************************
c    * SPK:  System and Pressurizer Kode *
c    *************************************
c
c      Single- and Multiple-Loop Version:
c
c           PWR Primary System
c           Single and Two-Phase Formulations
c
c      written by:  Shih-Ping Kao
c
c      updated:  July 3, 1984
c
c-------------------
c      main program
c-------------------
c
c      System Variables
c
c      1.  flow path components
c
c          dt     = time step size
c          dmi    = mass flow rate in each loop
c          dmic   = mass flow rate through core
c          dpb    = bouyancy pressure rise
c          dpl    = total pressure drop in the loop
c          hip    = core inlet plenum mixture enthalpy
c          hcr    = core exit enthalpy
c          hep    = core exit plenum enthalpy
c          hcl    = cold-leg exit enthalpy
c          hhl    = hot-leg exit enthalpy
c          hsg    = S/G exit enthalpy
c          qi     = nuclear power
c          qisg   = S/G power output
c          qith   = thermal power
c          pi     = pressure
c          ti     = time
c          tfa    = average fuel temp.
c          tca    = avg. core coolant temp.
c          tcr    = core exit temp.
c          tcl    = cold-leg exit temp. or core inlet temp.
c          thl    = hot-leg exit temp. or S/G inlet temp.
c          tsg    = S/G exit temp.
c          wi     = pump speed
c          zcsg   = S/G thermal center elevation
c          zc     = hydrostatic head
c
c      2.  pressurizer
c
c          alp    = volume fraction of vapor
c          hlp    = liquid enthalpy
c          hvp    = vapor enthalpy
c          pi     = pressurizer pressure
```

```
c
c         tlp    = liquid temperature
c         tvp    = vapor temperature
c         tsat   = saturation temperature
c         vlp    = liquid volume
c         vvp    = vapor volume
c         smass  = vapor mass
c         wmass  = liquid mass
c         wlevel = water level
c
c         wro    = rainout mass flow rate
c         wfl    = flashing mass flow rate
c         wsp    = spray mass flow rate
c         wsc    = spray condensate mass flow rate
c         wwc    = wall condensate mass flow rate
c         wsu    = surge mass flow rate
c
c     variables supplied by the user as functions of time:
c
c         qi     = nuclear power
c         pisec  = S/G secondary-side power
c         wsi    = safety-injection mass flow rate
c
c--------------------
c     common blocks
c--------------------
c
c     common/dimens/:  system dimensions
c
c         ari = flow area, m**2
c         di  = hydraulic diameter, m
c         vi  = volume, m**3
c         zi  = flow path, m
c     where i = the following
c         1:  core
c         2:  core upper plenum & R/V underhead
c         3:  hot leg
c         4:  S/G inlet plenum
c         5:  S/G tube
c         6:  S/G exit plenum
c         7:  cold leg
c         8:  R/V inlet plenum
c         9:  thermal shield (down comer)
c         10:  core lower plenum & support structure
c
c  zs     = distance from core inlet to S/G tube support sheet
c  zc0    = distance from core inlet to core mid-plane
c  nloop  = number of loops in the primary system
c
c
c     common/input/:  input parameters
c
c         t      = total simulation time
```

```
c         dtmax  = maximum time step size
c         tscram = time of scram
c         trip   = time of pump trip
c         ind    = (1/0) (single-/multiple-loop configuration)
c         ppow   = initial power in percent
c         pi1    = initial pressure
c         tcold1 = initial cold-leg temperature
c         wlevel = initial pressurizer water level
c         wi1    = initial pump rotor speed
c         dmi1   = initial loop mass flow rate
c         pisec1 = initial secondary pressure
c
c
c     common/prime/:  flow path component parameters
c
c         vip    = core exit plenum
c         vcr    = active core volume
c         vep    = core inlet plenum
c         vhl    = hot side volume (excluding core and S/G tube volume)
c         vsg    = S/G tube volume
c         vcl    = cold side volume (excluding core and S/G tube volume)
c         arcr   = core flow area
c         arhl   = total hot-leg flow area
c         arsg   = total S/G flow area
c         arcl   = total cold-leg flow area
c         dcr    = core subchannel hydraulic diameter
c         dhl    = hot-leg inner diameter
c         dsg    = S/G tube inner diameter
c         dcl    = cold-leg inner diameter
c
c
c     common/pfull/:  full power operation conditions
c
c         qi0    = 100% thermal power (watts)
c         dmi0   = mass flow rate at 100% power (Kg/s)
c         pi0    = pressure at full power (pa)
c         thot0  = hot leg temperature at full power (K)
c         tcold0 = cold leg temperature at full power (K)
c         delt0  = 100% power core temperature rise (K)
c
c
c     common/fuel/:  core parameters
c
c         fuell  = active fuel rod length
c         dclo   = clad O.D.
c         dcli   = clad I.D.
c         dfo    = fuel pin diameter
c         rodn   = no. of fuel rods
c         fmass  = uranium mass in core
c         clmass = cladding mass in core
c         ftd    = fuel theoretical density
c         fpuo2  = plutonium fraction by volume
c         hgap   = gap heat transfer coefficient
```

```
c        afuel  = total fuel heat transfer area
c
c
c    common/steam/:  steam generator parameters
c
c        do     = tube outer diameter
c        di     = tube inner diameter
c        ntube  = no. of tubes
c        tbl    = average tube length
c        zl     = average tube height
c        voltm  = tube metal volume
c
c
c    common/press/:  pressurizer paremeters
c
c        volprz = pressurizer volume
c        wv0    = water volume at 100% power
c        r      = pressurizer inside radius
c        vt     = pressurizer top wall volume
c        wall   = overall height (inside)
c        dw     = wall thickness
c        fwc    = wall condensation multiplier
c        ffl    = flashing suppression multiplier
c
c
c    common/presl/:  pressurizer wall parameters
c
c        vsp   = semisphere volume
c        ac    = cross-sectional area of the prz cylinder
c        acc   = wall cross-section area
c
c
c    common/heat/:  pressurizer heater parameters
c
c        taul  = heater power time constant
c        pph   = prop. heater high p set point
c        ppl   = prop. heater low p set point
c        pbh   = backup heater high p set point
c        pbl   = backup heater low p set point
c        qp    = prop. heater electric power
c        qb    = backup heater electric power
c        wlmin = min. water level before uncovering heaters
c        vh    = volume of heater banks
c        ah    = surface area of heaters
c        hh    = heat transfer coefficient of heaters
c
c
c    common/valves/:  pressurizer safety and relief valve parameters
c
c        amax  = maximum valve opening area
c        taua  = valve opening time constant
c        prv   = relief valve set point
c        psv1  = safety valve no. 1 set point
```

```
c
c     common/sprays/:  pressurizer spray valve parameters
c
c         wspmax= maximum spray mass flow rate
c         asp   = spray line cross-sectional area
c         hesp  = elevation of the spray line from the cold leg
c         hisp  = elevation of the prz. from the hot leg
c         psh   = spray valve high p set point
c         psl   = spray valve low p set point
c
c
c      common/charge/:  charge and let-down parameters
c
c         gpmc  = maximum charge flow rate
c         gpml  = maximum let-down flow rate
c         hch0  = charge fluid enthalpy (200 F, 2000 psia)  (J/Kg)
c
c
c     common/safe/:  High-Pressure-Safety-Injection system parameters
c
c         whpsi = rated HPSI mass flow rate
c         psih  = injection initiation pressure
c         psil  = injection termination pressure
c         hsi   = HPSI enthalpy
c
c
c     common/pumps/:  pump parameters
c
c         fp    = pump resistance coefficient
c         tr    = rated BHP torque (N-m)
c         tinert = pump inertia (kg-m**2)
c         ter   = rated electric torque (N-m)
c         twbr  = rated windage & bearing torque (N-m)
c         taue  = electric torque decay time constant (sec)
c         phcr  = rate pump head, phcr (Pa)
c         ror   = rated density, ror(kg/m**3)
c         dmirr= rated mass flow rate (kg/s)
c         wirr  = rated pump rotor speed (rad/s)
c         fh    = pump loss multiplier
c
c
c     common/head/:  pump head characteristic coefficients
c
c         hrp   = hc0 + hc1*x + hc2*x*x + hc3*x*x*x
c         x     = flow ratio/speed ratio (dmir/wr)
c
c
c     common/torq/:  pump torque characteristic coefficients
c
c         tbr   = tb0 + tb1*x + tb2*x*x + tb3*x*x*x
c
c
c     common/loss/:  form loss factors
```

```
c
c        sk1e  = core expansion
c        sk1c  = core contraction
c        cv1   = spacer loss coefficient
c        cv2   = spacer loss factor exponent
c        ep    = ratio of projected grid cross section
c                to undisturbed flow cross section
c        sk2   = R/V upper plenum & exit nozzle
c        sk4   = S/G inlet plenum
c        sk5e  = S/G tube expansion
c        sk5c  = S/G tube contraction
c        sk6   = S/G exit plenum
c        sk8e  = R/V inlet nozzle and 90 deg turn
c        sk9e  = thermal shield (down-comer) expansion
c        sk9c  = thermal shield (down-comer) contraction
c        sk10  = R/V lower plenum
c        sk90  = 90 deg bend
c        skv   = stop valve
c        nvh   = no. of stop valves in each hot leg
c        nvc   = no. of stop valves in each cold leg
c        ns    = no. of spacer grids
c        nbh   = no. of 90 deg bends in each hot leg
c        nbc   = no. of 90 deg bends in each cold leg
c
c
c     common/force/:  force functions
c
c        psfac = forcing function for secondary pressure
c        qfac  = forcing function for nuclear power
c        tps   = time vector for psfac
c        tq    = time vector for qfac
c        np    = number of entries in psfac table
c        nq    = number of entries in qfac table
c
c
c     common/break/:  break areas
c
c        ibr   = break indicator  (0/1) (no/yes breaks)
c        arbr  = break area  (m)
c        pbr   = break down-stream pressure
c        fbr   = critical mass flow rate multiplier
c
      common /dimens/ar1,ar2,ar3,ar4,ar5,ar6,ar7,ar8,ar9,ar10,
     &       d1,d2,d3,d4,d5,d6,d7,d8,d9,d10,
     &       v1,v2,v3,v4,v5,v6,v7,v8,v9,v10,
     &       z1,z2,z3,z4,z5,z6,z7,z8,z9,z10,zs,zc0,zc1,zep
      common /finert/za1,za2,za3,za4,za5,za6,za7,za8,za9,za10,
     &       za1,za1rv
      common /input/t,dtmax,tscram,ppow,ind,nloop,nh,nc,rn,pi1,
     &       tcold1,wlevel,wi1,dmi1,pisec1,n,m,isw
      common /prime/vip,vcr,vep,vhl,vsg,vcl,
     &      arip,arcr,arep,arhl,arsg,arcl,
     &      dip,dcr,dep,dhl,dsg,dcl
```

```
      common /pfull/qi0,dmi0,pi0,thot0,tcold0,delt0
      common/fuel/fuell,dclo,dcli,dfo,rodn,fmass,clmass,ftd,fpuo2,
     &           hgap,afuel
      common/steam/do,di,ntube,tbl,zl,voltm
      common /press/volprz,wv0,r,vt,wall,dw,fwc,ffl
      common /presl/vsp,ac,acc
      common /heat/taul,wlmin,qp,qb,pph,ppl,pbh,pbl,vh,ah,hh
      common /valves/amax,prv,psv1,psv2,psv3
      common /sprays/wspmax,asp,hesp,hisp,psh,psl
      common /charge/gpmc,gpml,hch0
      common /safe/whpsi,psih,psil,hsi
      common /pumps/tr,tinert,ter,taue,phcr,ror,twbr,dmirr,wirr,fh
      common /head/hc0,hc1,hc2,hc3
      common /torq/tb0,tb1,tb2,tb3
      common /loss/sk1e,sk1c,cv1,cv2,ep,sk2,sk4,sk5e,sk5c,sk6,
     &       sk8e,sk9e,sk9c,sk10,sk90,skv,nvh,nvc,ns,nbh,nbc
      common /force/psfac(30),tps(30),qfac(30),tq(30),np,nq
      common /break/ibr,arbr(4),pbr,fbr
      dimension trip(4),ititle(12)
      data pei,tabs,p6/3.1416,273.14,1.e6/
      call timing(icpu)
      read(5,10) (ititle(i),i=1,12)
10    format(1x,12a4)
c         nh  = no. of hot legs
c         nc  = no. of cold legs
      read(5,11) ind,nloop,nh,nc
      rn=nc/nh
      m=nloop
      if(ind.gt.0) m=1
      read(5,11) t,dtmax,tscram,ppow
      read(5,11) (trip(i),i=1,m)
      read(5,11) pi1,tcold1,wlevel,wi1,dmi1,pisec1
11    format(v)
c
c     read system parameters
c
      read(5,11) qi0,dmi0,pi0,thot0,tcold0,delt0
      read(5,11) ar1,ar2,ar3,ar4,ar5,ar6,ar7,ar8,ar9,ar10
      read(5,11) d1,d2,d3,d4,d5,d6,d7,d8,d9,d10
      read(5,11) v1,v2,v3,v4,v5,v6,v7,v8,v9,v10
      read(5,11) z1,z2,z3,z4,z5,z6,z7,z8,z9,z10
      read(5,11) zs,zc0,zc1,zep
      read(5,11) fuell,dclo,dcli,dfo,rodn,fmass,clmass,ftd,
     &           fpuo2,hgap,afuel
      read(5,11) do,di,ntube,tbl,zl,voltm
      read(5,11) volprz,wv0,r,vt,wall,dw,fwc,ffl
      read(5,11) taul,wlmin,qp,qb,pph,ppl,pbh,pbl,vh,ah,hh
      read(5,11) amax,prv,psv1,psv2,psv3
      read(5,11) wspmax,asp,hesp,hisp,psh,psl
      read(5,11) gpmc,gpml,hch0
      read(5,11) whpsi,psih,psil,hsi
      read(5,11) tr,tinert,ter,taue,phcr,ror,twbr,dmirr,wirr,fh
      read(5,11) hc0,hc1,hc2,hc3
```

```
      read(5,11) tb0,tb1,tb2,tb3
      read(5,11) sk1e,sk1c,cv1,cv2,ep,sk2,sk4,sk5e,sk5c,sk6,
     &           sk8e,sk9e,sk9c,sk10,sk90,skv,nvh,nvc,ns,nbh,nbc
      read(5,11) np,nq
      if(np.gt.0) read(5,11) (tps(i),psfac(i),i=1,np)
      if(nq.gt.0) read(5,11) (tq(i),qfac(i),i=1,nq)
      read(5,11) ibr
      if(ibr.gt.0) read(5,11) (arbr(i),i=1,m),pbr,fbr
c------------------------------------------------
c     calculate flow-path component dimensions
c------------------------------------------------
      vip=v8+v9+v10
      vcr=v1
      vep=v2
      if(ind.gt.0) go to 40
      vhl=v3+v4
      vsg=v5
      vcl=v6+v7
      go to 50
40    vhl=nh*(v3+v4)
      vsg=nh*v5
      vcl=nc*(v6+v7)
50    dip=d6
      dcr=d1
      dep=d2
      dhl=d3
      dsg=d5
      dcl=d7
      arip=ar6
      arcr=ar1
      arep=ar2
      arhl=ar3
      arsg=ar5
      arcl=ar7
c
c     calculate pressurizer wall parameters
c
      acc=4.*pei*((r+dw)*(r+dw)-r*r)
      ac=pei*r*r
      vsp=2.*ac*r/3.
c------------------------------
c     calculate fluid inertia
c------------------------------
      za1=z1/ar1
      za2=z2/ar2
      za3=z3/ar3
      za4=z4/ar4
      za5=z5/ar5
      za6=z6/ar6
      za7=z7/ar7
      za8=z8/ar8
      za9=z9/ar9
      za10=z10/ar10
```

```
      za1=za3+za4+za5+za6+za7
      za1rv=za1+za2+za8+za9+za10
c
c     print input parameters
c
      write(6,10) (ititle(i),i=1,12)
      write(6,100) (qi0/p6),(pi0/p6),(thot0-tabs),(tcold0-tabs),
     &             (nloop*dmi0)
100   format(//,' 1.  OPERATING CONDITIONS AT 100% POWER',/,
     &       ' Thermal Power, MW:            ',f6.1,/,
     &       ' Pressure, MPa:                ',f6.2,/,
     &       ' Hot Leg Temp, deg C:          ',f6.1,/,
     &       ' Cold Leg Temp, deg C:         ',f6.1,/,
     &       ' Total Mass Flow Rate, kg/s:   ',e10.4)
      write(6,110) ppow,(pi1/p6),(pisec1/p6),(tcold1-tabs),wlevel,
     &             wi1,dmi1,dtmax
110   format(//,' 2.  INITIAL CONDITIONS',/,
     &       ' Thermal Power, %:             ',f6.2,/,
     &       ' Pressure, MPa:                ',f6.2,/,
     &       ' Secondary Pressure, MPa:      ',f6.2,/,
     &       ' Cold Leg Temp, deg C:         ',f6.1,/,
     &       ' Pressurizer Water Level, m:   ',f6.2,/,
     &       ' Pump Rotor Speed, rad/s:      ',f6.2,/,
     &       ' Loop Mass Flow Rate, kg/s:    ',f6.0,/,
     &       ' Maximum Time Step Size:       ',f6.2)
      write(6,120) nh,nc
120   format(//,' 3.  REACTOR COOLANT SYSTEM PARAMETERS',//,
     &       ' Number of Hot Legs:    ',i2,/,
     &       ' Number of Cold Legs:   ',i2,//,
     &       ' Component',4x,'Volume(m3)',2x,'Min Area(m2)',2x,
     &       'Fl Path(m)'3x,'Eq Dia(m)')
      write(6,121) v1,ar1,z1,d1,v8,ar8,z8,d8,v9,ar9,z9,d9,
     &     v10,ar10,z10,d10,v2,ar2,z2,d2,v3,ar3,z3,d3,v7,ar7,z7,d7,
     &     v5,ar5,z5,d5,v4,ar4,z4,d4,v6,ar6,z6,d6
121   format(/,' Core',8x,4(3x,f6.3,3x),/,' RV IP',7x,
     &     4(3x,f6.3,3x),/,' RV DC',7x,4(3x,f6.3,3x),/,' RV LP',
     &     7x,4(3x,f6.3,3x),/,' RV UP',7x,4(3x,f6.3,3x),/,
     &       ' Hot Leg',5x,4(3x,f6.3,3x),/,
     &       ' Cold Leg',4x,4(3x,f6.3,3x),/,' SG Tube',5x,
     &     4(3x,f6.3,3x),/,' SG IP',7x,4(3x,f6.3,3x),/,
     &       ' SG EP',7x,4(3x,f6.3,3x))
      write(6,125)zs,zc0,zc1,zep
125   format(/,' Elevations',/,
     &       ' Tube Support Sheet from Core Inlet, m:  ',f6.2,/,
     &       ' Core Mid-Plane from Core Inlet:         ',f6.2,/,
     &       ' Suction-Leg U-Bend Height:              ',f6.2,/,
     &       ' Core Exit to R/V nozzle:                ',f6.2)
      write(6,130) rodn,fuell,dcli,dclo,dfo,fmass,clmass,hgap
130   format(//,' 4.  CORE PARAMETERS',/,
     &       ' No. of Fuel Rods:                 ',f10.1,/,
     &       ' Fuel Rod Active Length, m:        ',f10.2,/,
     &       ' Clad ID, m:                       ',e10.4,/,
     &       ' Clad OD, m:                       ',e10.4,/,
```

```
     &      ' Fuel Pin D, m:                     ',e10.4,/,
     &      ' UO2 Wt, kg:                        ',e10.4,/,
     &      ' Zircaloy Wt, kg:                   ',e10.4,/,
     &      ' Gap Heat Transfer Coef, W/m2-C:    ',e10.4)
      write(6,140) ntube,tbl,zl,di,do,voltm
140   format(//,' 5.  STEAM GENERATOR PARAMETERS',/,
     &      ' No. of Tubes:            ',i6,/,
     &      ' Avg. Tube Length, m:     ',f6.2,/,
     &      ' Avg. Tube Height, m:     ',f6.2,/,
     &      ' Tube ID, m:              ',f6.4,/,
     &      ' Tube OD, m:              ',f6.4,/,
     &      ' Tube Metal Volume, m3:   ',f6.4)
      write(6,150) volprz,r,wall,dw,wv0,fwc,ffl
150   format(//,' 6. PRESSURIZER PARAMETERS',/,' A. - Vessel -',/,
     &      ' Volume, m3:                        ',f6.2,/,
     &      ' Inside Radius, m:                  ',f6.3,/,
     &      ' Height, m:                         ',f6.2,/,
     &      ' Wall Thickness, m:                 ',f6.3,/,
     &      ' Water Volume at 100% Power, m3:    ',f6.2,/,
     &      ' Wall Condensation Multiplier:      ',f6.2,/,
     &      ' Flashing Suppression Multipler:    ',f6.2)
      write(6,152) wlmin,(qp/1.e3),(pph/p6),(ppl/p6),(qb/1.e3),
     &     (pbh/p6),(pbl/p6),vh,ah,hh
152   format(/,' B. - Heater -',/,
     &      ' Water Level Required to Cover Heaters, m:   ',f6.2,/,
     &      ' Prop. Heater Power, KW:                     ',f6.0,/,
     &      ' Prop. Heater High P Set-point, MPa:         ',f6.2,/,
     &      ' Prop. Heater Low P Set-point, MPa:          ',f6.2,/,
     &      ' Backup Heater Power, KW:                    ',f6.0,/,
     &      ' Backup Heater High P Set-point, MPa:        ',f6.2,/,
     &      ' Backup Heater Low P Set-point, MPa:         ',f6.2,/,
     &      ' Volume Occupied by Heaters, m3:             ',f6.2,/,
     &      ' Surface Area of Heaters, m2:                ',f6.2,/,
     &      ' Heater Heat Transfer Coefficient, W/m2-K:   ',f6.0)
      write(6,154) wspmax,asp,hesp,hisp,(psh/p6),(psl/p6)
154   format(/,' C. - Spray Valve -',/,
     &      ' Max. Spray Mass Flow Rate, kg/s:         ',f6.2,/,
     &      ' Spray Valve Cross-Sectional Area, m2:    ',f6.4,/,
     &      ' Elev. of Spray Line from Cold Leg, m:    ',f6.2,/,
     &      ' Elev. of Pressurizer from Hot Leg, m:    ',f6.2,/,
     &      ' High Pressure Set-point, MPa:            ',f6.2,/,
     &      ' Low Pressure Set-point, MPa:             ',f6.2)
      write(6,155) amax,(prv/p6),(psv1/p6),(psv2/p6),(psv3/p6)
155   format(/,' D. - Relief and Safety Valves -',/,
     &      ' Max. Valve Opening Area, m2:          ',f6.4,/,
     &      ' Relief Valve P Set-point, MPa:        ',f6.2,/,
     &      ' Safety Valve No. 1 Set-point, MPa:    ',f6.2,/,
     &      ' Safety Valve No. 2 Set-point, MPa:    ',f6.2,/,
     &      ' Safety Valve No. 3 Set-point, MPa:    ',f6.2)
      write(6,160) gpmc,gpml,hch0
160   format(//,' 6.  CHARGING AND LETDOWN',/,
     &      ' Max. Charging Flow Rate, kg/s:   ',e10.2,/,
     &      ' Max. Letdown Flow Rate, kg/s:    ',e10.2,/,
```

```
     &       ' Charging Fluid Enthalpy, J/kg:   ',e10.2)
      write(6,165) whpsi,psih/p6,psil/p6,hsi/1.e3
165   format(//,' 7.  HIGH-PRESSURE-SAFETY-INJECTION SYSTEM',/,
     &       ' Rated HPSI Mass Flow Rate, kg/s:        ',f6.2,/,
     &       ' Injection Initiation Pressure, MPa:     ',f6.2,/,
     &       ' Injection Termination Pressure, MPa:    ',f6.2,/,
     &       ' Injection Enthalpy, kJ/kg:              ',f6.0)
      write(6,170) tr,ter,phcr,ror,wirr,dmirr,tinert,fh
170   format(//,' 8.  PUMP',/,
     &       ' Rated Breakhorse Power Torque, N-m:  ',e10.4,/,
     &       ' Rated Electric Torque:               ',e10.4,/,
     &       ' Rated Pump Head Capacity, Pa:        ',e10.4,/,
     &       ' Rated Fluid Density, kg/m**3:        ',f10.2,/,
     &       ' Rated Pump Speed, rad/s:             ',f10.2,/,
     &       ' Rated Mass flow Rate, kg/s:          ',e10.4,/,
     &       ' Pump Rotor & Shaft Inertia, kg-m2:   ',e10.4,/,
     &       ' Pump Loss Multiplier:                ',f10.2)
      write(6,171) hc0,hc1,hc2,hc3
171   format(/,' 8.a  PUMP HEAD CHARACTERISTIC COEFFICIENTS',/,
     &         ' hc0 =  ',f7.4,/,' hc1 =  ',f7.4,/,
     &         ' hc2 =  ',f7.4,/,' hc3 =  ',f7.4)
      write(6,172) tb0,tb1,tb2,tb3
172   format(/,' 8.b  PUMP TORQUE CHARACTERISTIC COEFFICIENTS',/,
     &         ' tb0 =  ',f7.4,/,' tb1 =  ',f7.4,/,
     &         ' tb2 =  ',f7.4,/,' tb3 =  ',f7.4)
      write(6,180) sk1e,sk1c,cv1,cv2,ep,sk2,sk4,sk5e,sk5c,sk6,
     &         sk8e,sk9e,sk9c,sk10,sk90,skv,nvh,nvc,ns,nbh,nbc
180   format(//,' 9.  form loss factors:',/,
     &       ' Core expansion:                              ',f8.4,/,
     &       ' Core contraction:                            ',f8.4,/,
     &       ' Spacer loss coefficient:                     ',f8.4,/,
     &       ' Spacer loss factor exponent:                 ',f8.4,/,
     &       ' Ratio of projected grid cross section',/,
     &       '   to undisturbed flow cross section:         ',f8.4,/,
     &       ' R/V upper plenum & exit nozzle:              ',f8.4,/,
     &       ' S/G inlet plenum:                            ',f8.4,/,
     &       ' S/G tube expansion:                          ',f8.4,/,
     &       ' S/G tube contraction:                        ',f8.4,/,
     &       ' S/G exit plenum:                             ',f8.4,/,
     &       ' R/V inlet nozzle and 90 deg turn:            ',f8.4,/,
     &       ' Thermal shield (down-comer) expansion:       ',f8.4,/,
     &       ' Thermal shield (down-comer) contraction:     ',f8.4,/,
     &       ' R/V lower plenum:                            ',f8.4,/,
     &       ' 90 deg bend:                                 ',f8.4,/,
     &       ' Valve:                                       ',f8.4,/,
     &       ' No. of valves in hot leg:                    ',i2,/,
     &       ' No. of valves in cold leg:                   ',i2,/,
     &       ' No. of spacer grids:                         ',i2,/,
     &       ' No. of 90-deg bends in hot leg:              ',i2,/,
     &       ' No. of 90-deg bends in cold leg:             ',i2)
      if(np.gt.0) write(6,190) (tps(i),psfac(i),i=1,np)
      if(nq.gt.0) write(6,191) (tq(i),qfac(i),i=1,nq)
190   format(//,' 10a.  FORCING FUNCTION FOR SECONDARY PRESSURE',
```

```
     &       //4x,'Time',8x,'  Factor'/(4x,f6.1,8x,f6.4))
191   format(//,' 10b.  FORCING FUNCTION FOR NUCLEAR POWER',
     &       //4x,'Time',8x,'Factor'/(4x,f6.1,8x,f6.4))
      if(ibr.gt.0) write(6,195) pbr/p6,fbr
195   format(//,'  11.  COLD-LEG BREAK AREAS',/,
     &       ' Break Down-Stream Pressure, MPa:       ',f10.4,/,
     &       ' Critical Mass Flow Rate Multiplier:   ',f10.4)
      if(ibr.gt.0) write(6,196)  (i,arbr(i),i=1,m)
196   format(/,' Loop ',i2,' Break Area, m2:         ',e10.4)
c-----------------------------
c     solve system equations
c-----------------------------
      call trans(trip)
      call timing(ncpu)
      cpu=0.01*(ncpu-icpu)
      write(0,20)cpu
      write(6,20)cpu
20    format(7h cpu = ,f7.2,4h sec)
c------------------
c     plot graphs
c------------------
      write(0,21)
21    format(1x,"Do you want to plot results?  (0/1)(yes/no)")
      read(0,22)j
22    format(i1)
      if(j.gt.0) go to 30
      call plot(m,n,t,nloop)
30    stop
      end
```

```
      subroutine trans(trip)
c    ******************************************************
c    * perform initial state and transient calculations *
c    ******************************************************
      common /pfull/qi0,dmi0,pi0,thot0,tcold0,delt0
      common /prime/vip,vcr,vep,vhl,vsg,vcl,
     &       arip,arcr,arep,arhl,arsg,arcl,
     &       dip,dcr,dep,dhl,dsg,dcl
      common /press/volprz,wv0,r,vt,wall,dw,fwc,ffl
      common /presl/vsp,ac,acc
      common /pres2/hfp,hgp,hfg,rof,rog,hlp,hvp,alp,rolp,rovp,
     &        drvdp,drvdh,drldp,drldh,vlp,vvp,
     &        wmass,smass,wlevel,tlp,tvp,twi,tw
      common /steam/do,di,ntube,tbl,zl,voltm
      common /safe/wphsi,psih,psil,hsi
      common /dimens/ar1,ar2,ar3,ar4,ar5,ar6,ar7,ar8,ar9,ar10,
     &       d1,d2,d3,d4,d5,d6,d7,d8,d9,d10,v1,v2,v3,v4,v5,v6,
     &       v7,v8,v9,v10,z1,z2,z3,z4.z5,z6,z7,z8,z9,z10,zs,zc0,
     &       zcl,zep
      common /input/t,dtmax,tscram,ppow,ind,nloop,nh,nc,rn,pil,
     &       tcold1,wlevel,wil,dmil,pisecl,n,m,isw
      common /sprays/wspmax,asp,hesp,hisp,psh,psl
      common /spv/cvmax
      common /force/psfac(30),tps(30),qfac(30),tq(30),np,nq
      common /break/ibr,arbr(4),pbr,fbr
      dimension hhl(4),hsg(4),hcl(4),rohl(4),rosg(4),rocl(4),
     &    drdphl(4),drdpsg(4),drdpcl(4),drdhhl(4),drdhsg(4),drdhcl(4),
     &    xhl(4),xsg(4),xcl(4),thl(4),tsg(4),tcl(4),dpb(4),dpl(4),
     &    hc(4),alphl(4),alpsg(4),alpcl(4),qisg(4),dqhl(4),dqsg(4),
     &    dqsgdp(4),dhfmhl(4),dhfmsg(4),dhfmcl(4),dthldp(4),dtsgdp(4),
     &    dtcldp(4),dthldh(4),dtsgdh(4),dtcldh(4),zc(4),cplal(4)
      dimension hflhl(4),hflsg(4),hflcl(4),
     &    dmphl(4),dmpsg(4),dmpcl(4),dmhhl(4),dmhsg(4),
     &    dmhcl(4),dephl(4),depsg(4),depcl(4),dehhl(4),dehsg(4),
     &    dehcl(4),dmi(4),wi(4),dmhhl1(4),dmhsg1(4),dmhcl1(4),
     &    wbr(4),dwbdp(4),wsi(4),rmhl(4),rmsg(4),rmcl(4)
      dimension e15(4),e16(4),e17(4),e23(4),e53(4),e52(4),e51(4),
     &    e63(4),e62(4),e61(4),e73(4),e72(4),e71(4),g5(4),g6(4),g7(4),
     &    r62(4),r72(4),e44(4),r51(4),r61(4),r71(4),
     &    y5(4),y6(4),y7(4),dhhl(4),dhsg(4),dhcl(4)
c
c        rg   = gas constant of H2O, J/kg-K.
c        dpi  = pressure perturbation
      data rg,dpi/462.,1.e2/
      data pei,g,tabs/3.1416,9.80665,273.16/
c
c     properties of carbon steel
c        cndcs = conductivity
c        rocs  = density
c        cpcs  = thermal capacity
      data cndcs,rocs,cpcs/46.,7.858e3,600./
c    ********************************
c    * determines initial conditions *
```

```
c   ********************************
      ti=0.
      isw=1
      n=1
      do 3 i=1,m
3     zc(i)=zc0
      call power(qi,ti,tscram,ppow,dt,isw)
      do 4 i=1,m
      dmi(i)=dmi1
4     wi(i)=wi1
      pi=pi1
c--------------------------------------
c     set initial pressurizer condition
c--------------------------------------
      wsp=0.
      wsc=0.
      wrv=0.
      wsv=0.
      wsum=0.
      wsu=0.
      wcharg=0.
      wro=0.
      wfl=0.
      wwc=0.
      qh=0.
      uerr=0.
      hlp=hf(pi,dhfdp)
      hvp=hg(pi,dhgdp)
      call state(pi,hlp,hvp,hfp,hgp,tsat,dtsdp,tlp,tvp,rolp,rovp,
     1      drldp,drldh,drvdp,drvdh,rof,rog,drfdp,drgdp,dhfdp,dhgdp,
     2      cplp,cpvp,vislp,visvp,cndlp,cndvp,sig)
      hfg=hgp-hfp
      twi=tsat
      tw=twi
      wlevel=wlevel
      wlr=100.*wlevel/wall
      vlp=(wlevel-r)*ac+vsp
      if(wlevel.lt.r) vlp=2.*pei*wlevel*wlevel*wlevel/3.
      wmass=vlp*rolp
      vvp=volprz-vlp
      smass=vvp*rovp
      alp=vvp/volprz
      vwall=vt+acc*(wall-wlevel)
      dhv=0.
      jflag=0
      iflag=0
      jsw=0
      dmhi=smass*hvp+wmass*hlp
c-------------------------------------------------------------
c     determine initial states of the flow path components
c-------------------------------------------------------------
      qith=qi
      tip=tcold1
```

```
      hip=hl(tip,hfp,pi)
      cplip=cpl(hip,pi)
c
c     initial guesses for dmic, dmi(i), and tcr
c
      dmic=nloop*dmi0
      tcr=tip+qith/(cplip*dmic)
      tep=tcr
      do 101 i=1,m
      tsg(i)=tip
101   tcl(i)=tip
      delt=tep-tip
      deltr=delt/delt0
      tfa=tcr
      hcr=hl(tcr,hfp,pi)
      hep=hcr
      do 201 i=1,m
      hhl(i)=hcr
      hsg(i)=hip
201   hcl(i)=hip
      hsu=hcr
      cplcr=cpl(hcr,pi)
      cplip=cpl(hip,pi)
      cpla=0.5*(cplcr+cplip)
      do 221 i=1,m
221   cplal(i)=cpla
      do 301 i=1,m
301   qisg(i)=qith/nh
      qithp=100.*(qith/qi0)
      rocr=rol(hcr,pi,dum1,dum2)
      roip=rol(hip,pi,dum1,dum2)
      roep=rocr
      do 361 i=1,m
      rohl(i)=rocr
      rosg(i)=roip
361   rocl(i)=roip
      xip=0.
      xcr=0.
      xep=0.
      do 37 i=1,m
      xhl(i)=0.
      xsg(i)=0.
37    xcl(i)=0.
c
c     determine initial pump speed and mass flow rate
c
      call pump(dmi,dmic,wi,ti,dt,roip,rocr,roep,rohl,rosg,rocl,
     &     xip,xcr,xep,xhl,xsg,xcl,pi,
     &     dpb,dpl,hc,trip,vislp)
      admic=abs(dmic)
      hcr=hip+qith/dmic+hc(1)/rocl(1)
      hep=hcr
      do 381 i=1,m
```

```
381   hhl(i)=hcr
      call hflow(pi,hip,admic,arip,dip,hfg,hfp,hgp,rof,rog,drfdp,
     &     drgdp,sig,hflip,tip,dtipdp,dtipdh,alpip,xip,roip,drdpip,
     &     drdhip,dhfmip,5)
      call hflow(pi,hcr,admic,arcr,dcr,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hflcr,tcr,dtcrdp,dtcrdh,alpcr,xcr,rocr,drdpcr,
     2     drdhcr,dhfmcr,1)
      call hflow(pi,hep,admic,arep,dep,hfg,hfp,hgp,rof,rog,drfdp,
     &     drgdp,sig,hflep,tep,dtepdp,dtepdh,alpep,xep,roep,drdpep,
     &     drdhep,dhfmep,5)
c --  core energy  --
      call coef(pi,dmic,hfp,hcr,hip,hep,rocr,roip,roep,
     &     alpcr,alpip,drdpcr,drdpip,
     1     drdpep,drdhcr,vcr,depcr,dehcr,dmpcr,dmhcr,dmhcr1,rmcr)
      do 401 i=1,m
      admi=abs(dmi(i))
c
c     determine initial properties
c
      call hflow(pi,hhl(i),rn*admi,arhl,dhl,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hflhl(i),thl(i),dthldp(i),dthldh(i),alphl(i),
     2     xhl(i),rohl(i),drdphl(i),drdhhl(i),dhfmhl(i),2)
      call hflow(pi,hsg(i),rn*admi,arsg,dsg,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hflsg(i),tsg(i),dtsgdp(i),dtsgdh(i),alpsg(i),
     2     xsg(i),rosg(i),drdpsg(i),drdhsg(i),dhfmsg(i),3)
      call hflow(pi,hcl(i),admi,arcl,dcl,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hflcl(i),tcl(i),dtcldp(i),dtcldh(i),alpcl(i),
     2     xcl(i),rocl(i),drdpcl(i),drdhcl(i),dhfmcl(i),4)
c
c     calculate state equation coeficients
c
c --  hot-leg energy  --
      call coef(pi,rn*dmi(i),hfp,hhl(i),hep,hsg(i),rohl(i),roep,
     1     rosg(i),alphl(i),alpep,drdphl(i),drdpep,drdpsg(i),
     &     drdhhl(i),vhl,dephl(i),
     &     dehhl(i),dmphl(i),dmhhl(i),dmhhl1(i),rmhl(i))
c --  steam generator energy  --
      call coef(pi,rn*dmi(i),hfp,hsg(i),hhl(i),hcl(i),rosg(i),rohl(i),
     1     rocl(i),alpsg(i),alphl(i),drdpsg(i),drdphl(i),drdpcl(i),
     &     drdhsg(i),vsg,
     &     depsg(i),dehsg(i),dmpsg(i),dmhsg(i),dmhsg1(i),rmsg(i))
c --  cold-leg energy  --
      call coef(pi,dmi(i),hfp,hcl(i),hsg(i),hip,rocl(i),rosg(i),
     1     roep,alpcl(i),alpsg(i),drdpcl(i),drdpsg(i),drdpcl(i),
     &     drdhcl(i),vcl,
     &     depcl(i),dehcl(i),dmpcl(i),dmhcl(i),dmhcl1(i),rmcl(i))
401    continue
      hflip=hflcl(1)
      hflep=hflhl(1)
      roip=rocl(1)
      roep=rocr
      roa=0.5*(rocr+roip)
      drdpip=drdpcl(1)
```

```
      drdhip=drdhcl(1)
      drdpep=drdpcr
      drdhep=drdhcr
c  -- determine initial secondary-side conditions
      pisec=pisec1
      hfsec=hf(pisec,dum1)
      tsec=tl(hfsec,pisec,dum1,dum2)
      do 501 i=1,m
c  -- determine initial S/G conditions --
      call sgpow(qisg(i),rn*dmi(i),pi,hfp,hfg,hcr,hsg(i),hcl(i),tcr,
     &     tsg(i),tcl(i),dthldp(i),dtsgdp(i),dthldh(i),
     &     dtsgdh(i),cplal(i),tsec,pisec,arsg,xhl(i),
     &     xsg(i),strl1,strl2,dqhl(i),dqsg(i),dqsgdp(i),boil,isw)
c  -- determine initial bouyancy pressure rise --
      call sgbh(bsg,rn*dmi(i),cplal,pi,rof,rog,rohl(i),rosg(i),xhl(i),
     &     xsg(i),xcl(i),strl1,strl2,arsg,dsg,sig,boil)
      dpb(i)=g*(-(rocr+roip)*zc0-0.5*(roep+rohl(i))*(zs-2.*zc0)
     &       +bsg+(rocl(i)-rosg(i))*zcl+0.5*(roip+rocl(i))*zs)
      zcsg=bsg/(rosg(i)-rohl(i))
501   zc(i)=zs+zcsg-zc0
c  -- determine initial fuel temp. --
      tca=0.5*(tcr+tip)
      call fulpin(dt,qi,qith,tfa,tca,hcr,hfp,dmic,pi,cplcr,isw)
c  -- calculate initial time step size --
      call time(dt,dtmax,vcr,rocr,rof,dmic,alp)
c
c     calculate maximum spray-valve loss coefficient
c
      hsv=g*roip*hesp
      hlv=g*roep*(hisp+wlevel)
      dpch=hc(1)-dpl(1)
      dpsp=dpch+hlv-hsv
      cvmax=asp*asp*dpsp/(wspmax*wspmax)
c------------------------------
c     print initial conditions
c------------------------------
      call output(ti,pi,qi,qisg,qith,wi,dmi,dmic,tfa,tca,tip,tcr,
     &     tep,thl,tsg,tcl,tsec,alpip,alpcr,alpep,alphl,alpsg,alpcl,
     &     zc,deltr,tsat,tlp,tvp,wlevel,qh,alp,wlr,wfl,wro,wwc,
     &     wsv,wrv,wsp,wcharg,wsu,wsum,dpl,dpb,hc,jsw,jflag,iflag,
     &     wbr,wsi,m,n,isw)
c    ******************************************************
c    *  solve the system and pressurizer state equations *
c    ******************************************************
10    ti=ti+dt
      isw=0
      nl=1
      nhl=1
      if(m.eq.1) nl=nloop
      if(m.eq.1) nhl=nh
      y2=hip
      y3=hcr
      y4=hep
```

```
      do 5 i=1,m
      y5(i)=hhl(i)
      y6(i)=hsg(i)
5     y7(i)=hcl(i)
      x1=pi
      x2=hlp
      x3=hvp
      x4=alp
c    *********************************
c    * flow path component algorithm *
c    *********************************
c
c     calculate thermal boundary conditions
c
      call power(qi,ti,tscram,ppow,dt,isw)
      call fulpin(dt,qi,qith,tfa,tca,hcr,hfp,dmic,pi,cplcr,isw)
      fact=1.0
      if(np.gt.0) call table(fact,ti,psfac,tps,np)
      pisec=fact*pisec1
      hfsec=hf(pisec,dum1)
      tsec=tl(hfsec,pisec,dum1,dum2)
      do 102 i=1,m
c
c     calculate steam generator power output
c
      call sgpow(qisg(i),rn*dmi(i),pi,hfp,hfg,hhl(i),hsg(i),hcl(i),
     &     thl(i),tsg(i),tcl(i),dthldp(i),dtsgdp(i),dthldh(i),
     &     dtsgdh(i),cplal(i),tsec,pisec,arsg,xhl(i),xsg(i),strl1,
     &     strl2,dqhl(i),dqsg(i),dqsgdp(i),boil,isw)
c
c     calculate steam generator bouyancy head
c
      call sgbh(bsg,rn*dmi(i),cplal(i),pi,rof,rog,rohl(i),rosg(i),
     &     xhl(i),xsg(i),xcl(i),strl1,strl2,arsg,dsg,sig,boil)
      rolch=rocl(i)
c
c     calculate charging and let-down mass flow rates
c
      call charge(wcharg,wld,hcl(i),hcharg,wmass,qithp,wv0,rolch,
     &     dt,nloop)
c
c     calculate safety injection mass flow rate
c
      call hpsi(wsi(i),pi)
102   continue
      if(ibr.eq.0) go to 1101
c
c     calculate cold-leg break mass flow rate
c
      do 1002 i=1,m
      call critic(wbr(i),dwbdp(i),pi,pbr,xcl(i),rocl(i),drdpcl(i),
     &     drfdp,drgdp,dhfdp,dhgdp,hfg,rof,rog,arbr(i),fbr)
1002  continue
```

```
1101  iflow=0
      do 1100 i=1,m
      if(dmi(i).gt.0.) go to 1100
      iflow=i
1100  continue
c---------------------------
c     elements of Matrix E
c---------------------------
      el10=0.
      do 11 i=1,m
11    el10=el10+dmphl(i)+dmpsg(i)+dmpcl(i)+dt*dwbdp(i)
      e11=vip*drdpip+dmpcr+vep*drdpep+el10
      e12=vip*drdhip+dmhcr1
      e13=dmhcr
      e14i=0.
      do 111 i=1,m
      e14i=e14i+dmhhl1(i)
111   continue
      e14=vep*drdhep+e14i
      e21=-vip
      e22i=0.
      do 1110 i=1,m
      if(dmi(i).lt.0.) go to 1110
      e22i=e22i+dmi(i)
1110  continue
      e22=vip*roip+dt*dmic
      if(iflow.gt.0) e22=vip*roip+dt*e22i
      e31=depcr
      e32=-dt*dmic
      e33=dehcr+dt*dmic*dhfmcr
      e41=-vep
      e42=-dt*dmic*dhfmcr
      rflow=0.
      do 1120 i=1,m
      if(dmi(i).gt.0.) go to 1120
      rflow=rflow+dmi(i)
1120  continue
      e43=vep*roep+dt*(dmic+rflow)
      do 12 i=1,m
      if(dmi(i).lt.0.) go to 1130
      e15(i)=dmhhl(i)+dmhsg1(i)
      e16(i)=dmhsg(i)+dmhcl1(i)
      e17(i)=dmhcl(i)
      e23(i)=-dt*nl*dmi(i)*dhfmcl(i)
      e51(i)=dephl(i)
      e52(i)=-dt*nl*dmi(i)
      e53(i)=dehhl(i)+dt*nl*dmi(i)*dhfmhl(i)
      e61(i)=depsg(i)+dt*nhl*dqsgdp(i)
      e62(i)=-dt*nl*dmi(i)*dhfmhl(i)+dt*nhl*dqhl(i)
      e63(i)=dehsg(i)+dt*nl*dmi(i)*dhfmsg(i)+dt*nhl*dqsg(i)+
     &       0.5*voltm*rcin(tsg(i))/cpla1(i)
      e71(i)=depcl(i)
      e72(i)=-dt*nl*dmi(i)*dhfmsg(i)
```

```
      e73(i)=dehcl(i)+dt*(nl*dmi(i)-wbr(i))*dhfmcl(i)
      e44(i)=0.
      r51(i)=0.
      r61(i)=0.
      r71(i)=0.
      go to 12
1130  e15(i)=dmhcl(i)+dmhsg1(i)
      e16(i)=dmhsg(i)+dmhhl1(i)
      e17(i)=dmhhl(i)
      e44(i)=-dt*dmi(i)*dhfmhl(i)
      e51(i)=depcl(i)
      r51(i)=dt*nl*dmi(i)
      e53(i)=dehcl(i)-dt*(nl*dmi(i)-wbr(i))*dhfmcl(i)
      e61(i)=depsg(i)
      e62(i)=dt*nl*dmi(i)*dhfmcl(i)
      e63(i)=dehsg(i)-dt*nl*dmi(i)*dhfmsg(i)+
     &       0.5*voltm*rcin(tsg(i))/cplal(i)
      e71(i)=dephl(i)
      e72(i)=dt*nl*dmi(i)*dhfmsg(i)
      e73(i)=dehhl(i)-dt*nl*dmi(i)*dhfmhl(i)
      e23(i)=0.
      e52(i)=0.
      r62(i)=0.
      r72(i)=0.
12    continue
c---------------------------
c     elemnents of Vector g
c---------------------------
      g10=0.
      do 33 i=1,m
      g10=g10+dt*(wsi(i)-wbr(i))
33    continue
      g1=dt*nl*(wcharg-wld)+g10
      g20=0.0
      do 34 i=1,m
      if(dmi(i).lt.0.) go to 34
      g20=g20+dt*nl*dmi(i)*(hflcl(i)-hip)
34    continue
      g2=g20
      g3=dt*dmic*(hip-hflcr)+dt*qith
      g40=0.0
      do 36 i=1,m
      if(dmi(i).gt.0.) go to 36
      g40=g40+dmi(i)*(hflhl(i)-hep)
36    continue
      g4=dt*g40+dt*dmic*(hflcr-hep)
      do 14 i=1,m
      if(dmi(i).lt.0.) go to 38
      g5(i)=dt*nl*dmi(i)*(hep-hflhl(i))
      g6(i)=dt*nl*dmi(i)*(hflhl(i)-hflsg(i))-dt*nhl*qisg(i)
      hcla=0.5*(hcl(i)+hsg(i))
      g7(i)=dt*nl*dmi(i)*(hflsg(i)-hflcl(i))+
     &     dt*wcharg*(hcharg-hcla)+
```

```
     &       dt*wsi(i)*(hsi-hcla)-dt*wbr(i)*(hflcl(i)-hcla)+
     &       dt*nl*dmi(i)*hc(i)/rocl(i)
      go to 14
38    hcla=0.5*(hcl(i)+hep)
      g5(i)=dt*nl*dmi(i)*(hflcl(i)-hip)+dt*wcharg*(hcharg-hcla)+
     &       dt*wsi(i)*(hsi-hcla)-dt*wbr(i)*(hflcl(i)-hcla)+
     &       dt*dmi(i)*hc(i)/rocl(i)
      g6(i)=dt*nl*dmi(i)*(hflsg(i)-hflcl(i))-dt*nhl*qisg(i)
      g7(i)=dt*nl*dmi(i)*(hflhl(i)-hflsg(i))
14    continue
c----------------------
c     Gauss elimation
c----------------------
      do 16 i=1,m
      e53(i)=1./e53(i)
      e52(i)=e52(i)*e53(i)
      e51(i)=e51(i)*e53(i)
      r51(i)=r51(i)*e53(i)
      e63(i)=1./e63(i)
      e62(i)=e62(i)*e63(i)
      e61(i)=e61(i)*e63(i)
      e73(i)=1./e73(i)
      e72(i)=e72(i)*e73(i)
      e71(i)=e71(i)*e73(i)
      if(dmi(i).lt.0.) go to 120
      r62(i)=-e52(i)*e62(i)
      e61(i)=e61(i)-e51(i)*e62(i)
      r72(i)=-r62(i)*e72(i)
      e71(i)=e71(i)-e61(i)*e72(i)
      if(i.eq.1) go to 125
      g5(i)=g5(i)*e53(i)
      g6(i)=g6(i)*e63(i)-g5(i)*e62(i)
      g7(i)=g7(i)*e73(i)-g6(i)*e72(i)
      go to 16
125   g6(1)=g6(1)*e63(1)
      g7(1)=g7(1)*e73(1)
      go to 16
120   r61(i)=-r51(i)*e62(i)
      e61(i)=e61(i)-e51(i)*e62(i)
      r71(i)=-r61(i)*e72(i)
      e71(i)=e71(i)-e61(i)*e72(i)
      g5(i)=g5(i)*e53(i)
      g6(i)=g6(i)*e63(i)-g5(i)*e62(i)
      if(i.eq.1) go to 16
      g7(i)=g7(i)*e73(i)-g6(i)*e72(i)
16    continue
      e43=1./e43
      e42=e42*e43
      e41i=0.
      do 135 i=1,m
135   e41i=e41i+e44(i)*e71(i)
      e41=e43*(e41-e41i)
      r42=0.
```

```
      do 140 i=1,m
140   r42=r42-e44(i)*r71(i)*e43
      e33=1./e33
      e32=e32*e33
      e31=e31*e33
      g3=g3*e33
      r23=0.
      do 20 i=1,m
20    r23=r23-e23(i)*r72(i)
      r22=-e42*r23
      e22=1./(e22-e32*r22-r42*r23)
      e21i=0.
      do 21 i=1,m
21    e21i=e21i+e71(i)*e23(i)
      e21=e22*(e21-e31*r22-e41*r23-e21i)
      e14i=0.
      do 23 i=1,m
23    e14i=e14i+r72(i)*e17(i)+r62(i)*e16(i)+e52(i)*e15(i)
      e14=e14-e14i
      e13=e13-e42*e14
      e12i=0.
      do 165 i=1,m
165   e12i=e12i+(r71(i)*e17(i)+r61(i)*e16(i)+r51(i)*e15(i))
      e12=e12-(e12i+r42*e14+e32*e13)
      e11i=0.
      do 22 i=1,m
22    e11i=e11i+(e71(i)*e17(i)+e61(i)*e16(i)+e51(i)*e15(i))
      e11=1./(e11-(e11i+e41*e14+e31*e13+e21*e12))
      if(dmi(1).lt.0.) go to 150
      eg2=e12*e22
      eg7=e17(1)-eg2*e23(1)
      eg6=e16(1)-eg7*e72(1)
      eg5=(e15(1)-eg6*e62(1))*e53(1)
      eg4=e43*(e14-eg2*r23)
      eg3=e13-eg2*r22
      go to 175
150   eg2=e12*e22
      eg3=e13-eg2*r22
      eg4=e43*(e14-eg2*r23)
      eg7=e17(1)-eg4*e44(1)
      eg6=e16(1)-eg7*e72(1)
175   gj=0.
      do 24 j=2,m
24    gj=gj+((e17(j)-eg2*e23(j)-eg4*e44(j))*g7(j)+g6(j)*e16(j)+
     &   g5(j)*e15(j))
c---------------------------------------
c     track initial insurge conditions
c---------------------------------------
c
c         iflag  = 1, onset of insurge transient
c                = 0, otherwise
c         jflag  = 1, flashing on (for outsurge only)
c                = 0, flashing off
```

```
c
25    if(wsu.ne.0.) go to 30
      iflag=0
      go to 50
30    if(wsu.gt.0.) go to 40
c
c     outsurge transient
c
      if(wsum.gt.0.) go to 35
      iflag=0
      jflag=0
      hsu=x2
      if(alp.ge.1.) hsu=x3
      hsp=hcl(1)
      go to 50
35    hsu=hsui
      if(pi.lt.pii) jflag=1
      iflag=0
      go to 50
40    hsu=hhl(1)
      if(iflag.gt.0) go to 50
c
c     onset of insurge transient
c
      wsum=0.
      iflag=1
      wmassf=wmass
      hsui=hsu
      pii=pi
c
50    if(dmi(1).lt.0.) go to 190
      p11=e11*(-1.+eg5*(hsu-hhl(1)))*dt
      p12=e11*(g1-gj-eg2*g2-eg3*g3-eg4*g4-eg5*g5(1)-eg6*g6(1)-
     &    eg7*g7(1))
      go to 185
190   p11=e11*(-1.+eg7*e73(1)*(hsu-hhl(1)))*dt
      p12=e11*(g1-eg2*g2-eg3*g3-eg4*g4-e15(1)*g5(1)-eg6*g6(1)-
     &    eg7*e73(1)*g7(1)-gj)
c    *****************************************
c    * two-fluid pressurizer model algorithm *
c    *****************************************
185   wfl=0.
      wro=0.
      dwfl=0.
      dmvdp=vvp*drvdp
      dmvdh=vvp*drvdh
      dmvda=volprz*rovp
      dmldp=vlp*drldp
      dmldh=vlp*drldh
      dmlda=-volprz*rolp
c-----------------------------------
c     calculate boundary conditions
c-----------------------------------
```

```
      hsp=hcl(1)
      rosp=rocl(1)
      rosu=rohl(1)
      if(wsu.lt.0.) rosu=rolp
      call spray(pi,wsp,wlevel,hc(1),dpl(1),rosp,rosu)
      wsc=wsp*(hfp-hsp)/(hvp-hfp)
      call heater(pi,qh,tlp,wlevel,dt,isw)
      cvv=cpvp-rg
      gam=cpvp/cvv
      gamp=(2./(gam+1.))**((gam+1.)/(gam-1.))
      call valve(pi,dt,arv,asv1,asv2,asv3)
      wrv=4.82e-2*arv*sqrt(gam*pi*rovp*gamp)
      wsv=4.82e-2*(asv1+asv2+asv3)*sqrt(gam*pi*rovp*gamp)
c---------------------------------
c     calculate wall condensation
c---------------------------------
      dwl=wall-wlevel
      wl=2.*r+dwl
      arw=ac+2.*dwl*pei*r
      vwall=vt+acc*wl
      if(twi.ge.tsat) go to 60
      if(tw.ge.tsat) tw=twi
      rcpcs=0.5*dw*rocs*cpcs
      dtvwi=tsat-twi
      hwc=0.943*(g*rolp*(rolp-rovp)*cndlp*cndlp*cndlp*(hvp-hfp)
     &    /(vislp*wl*dtvwi))**0.25
      uh=5.*hwc
      uk=cndcs/(0.02*dw)
      ut=1./(1./uh+1./uk)
      tauw=rcpcs/ut
      qwc=ut*(tsat-tw)
      dtw=dt*(tsat-tw)/tauw
      tw=tw+dtw
      wwc=fwc*arw*qwc/(hvp-hfp)
      twi=tsat-qwc/hwc
      go to 70
60    qwc=0.
      wwc=0.
      twi=tsat
      tw=tsat
70    if(jsw.eq.6) go to 600
      if(alp.ge.1.) go to 500
c--------------------------
c     elements of Matrix A
c--------------------------
      a11=-vvp+(x3-hsu)*dmvdp
      a12=smass+(x3-hsu)*dmvdh
      a13=(x3-hsu)*dmvda
      a21=-vlp+(x2-hsu)*dmldp
      a22=wmass+(x2-hsu)*dmldh
      a23=(x2-hsu)*dmlda
      a31=dmvdp
      a32=dmvdh
```

```
      a33=dmvda
      a41=dmldp
      a42=dmldh
      a43=dmlda
c---------------------------
c     elements of Matrix C
c---------------------------
      c11=dt*(hsp-hfp)
      c12=-dt*(hfp-hsu)
      c13=-dt*(x3-hsu)
      c14=-dt*(x3-hsu)
      c15=c14
      c21=-c12
      c22=c21
      c23=c21
      c24=dt
      c31=-dt
      c32=-dt
      c33=-dt
      c34=-dt
      c41=dt
      c42=dt
      c43=dt
      c44=dt
c------------------------------
c     elements of Matrix (-D)
c------------------------------
      d11=dt*(hfp-hsu)
      d12=-dt*(hgp-hsu)
      d21=-d11
      d22=-d12
      d31=dt
      d32=-dt
      d41=-dt
      d42=dt
c-------------------------------------------------------
c     Condition 1:  hvp > hgp, hlp < hfp (first guess)
c-------------------------------------------------------
      f1=c11*wsp+c12*wsc+c13*wwc+c14*wrv+c15*wsv
      f2=c21*wsp+c22*wsc+c23*wwc+c24*qh
      f3=c31*wsc+c32*wwc+c33*wrv+c34*wsv
      f4=c42*wsp+c43*wsc+c44*wwc
      a11i=a11
      a21i=a21
      a31i=a31
      a41i=a41
      if(jflag.eq.0) go to 105
      if(jsw.eq.2.or.jsw.eq.4) go to 105
c
c     include flashing during out-surge
c
      dwfl=-ffl*wmassf*(dhfdp-1./rolp)/(hgp-hfp)
      a11i=a11+d12*dwfl
```

```
      a21i=a21+d22*dwfl
      a31i=a31+d32*dwfl
      a41i=a41+d42*dwfl
c
c     backward elimination
c
105   a43i=1./a43
      a41i=a41i*a43i
      a42i=a42*a43i
      a32i=1./a32
      a31i=(a31i-a41i*a33)*a32i
      r32=-a42i*a33*a32i
      a22i=1./(a22-a42i*a23)
      a21i=(a21i-a41i*a23)*a22i
      r12=-a42i*a13-a12*r32
      r11=a11i-a41i*a13-a31i*a12
      a11i=1./(r11-a21i*r12)
      ae4=a43i*(-a13+a33*a32i*a12+a23*a22i*r12)
      q11i=a11i*ae4*c41
      q12i=a11i*(f1+f4*ae4-f3*a32i*a12-f2*a22i*r12)
      wsu1=(q12i-p12)/(p11-q11i)
      dpi1=q11i*wsu1+q12i
c
c     forward substitution
c
      f4ii=a43i*(c41*wsu1+f4)
      f3i=a32i*(f3-f4ii*a33)
      f2i=a22i*(f2-f4ii*a23)
      dhlp1=f2i-a21i*dpi1
      dhvp1=f3i-a31i*dpi1-r32*dhlp1
      dalp1=f4ii-a41i*dpi1-a42i*dhlp1
      p1=x1+dpi1
      hlp1=x2+dhlp1
      hvp1=x3+dhvp1
      alp1=x4+dalp1
      if(alp1.gt.1.) go to 500
      wfl=dwfl*dpi1/dt
      hfp=hf(p1,dhfdp)
      hgp=hg(p1,dhgdp)
c---------------------------------
c     pressurizer condition check
c---------------------------------
      if(hlp1.lt.hfp) go to 100
      jsw=2
      if(hvp1.gt.hgp) jsw=3
      go to 110
100   jsw=4
      if(hvp1.gt.hgp) jsw=1
110   if(jsw.gt.1.and.wwc.gt.0.) go to 60
      go to (700,200,300,400), jsw
c------------------------------------
c     Condition 2:  hlp=hfp, hvp=hgp
c------------------------------------
```

```
 200    a11=a11+dhgdp*a12
        a21=a21+dhfdp*a22
        a31=a31+dhgdp*a32
        a41=a41+dhfdp*a42
        f1=a12*(hvp1-hgp)
        f2=a22*(hlp1-hfp)
        f3=a32*(hvp1-hgp)
        f4=a42*(hlp1-hfp)
c
c       backward elimination
c
        a43=1./a43
        a41=a41*a43
        d42=d42*a43
        d41=d41*a43
        d31=1./(d31-d41*a33)
        a31=(a31-a41*a33)*d31
        d32=(d32-d42*a33)*d31
        d21=d21-d41*a23
        d22=1./(d22-d42*a23-d32*d21)
        a21=(a21-a41*a23-a31*d21)*d22
        d11=d11-d41*a13
        d12=d12-d42*a13-d32*d11
        a11=1./(a11-a41*a13-a31*d11-a21*d12)
        r2=d22*d12
        r3=d31*(d11-d21*r2)
        r4=a43*(a13-a23*r2-a33*r3)
        q11=-a11*r4*c41
        q12=a11*(f1-f2*r2-f3*r3-f4*r4)
 220    wsu=((q11i-q11)*wsu1+q12+q12i-p12)/(p11-q11)
        dwsu1=wsu-wsu1
        dpi1=q11*dwsu1+q12
        pi=p1+dpi1
        go to (1000,210,310,410), jsw
c
c       forward substitution
c
 210    f4i=a43*(c41*dwsu1+f4)
        f3i=d31*(f3-f4i*a33)
        f2i=f2-f4i*a23
        f2ii=d22*(f2i-f3i*d21)
        wf1=f2ii-a21*dpi1
        wro=f3i-a31*dpi1-d32*wf1
        dalp1=f4i-a41*dpi1-d42*wf1-d41*wro
        alp=alp1+dalp1
        if(alp.gt.1.) go to 500
        hlp=hf(pi,dhfdp)
        hvp=hg(pi,dhgdp)
        jsw=2
        go to 1000
c---------------------------------------
c       Condition 3:  hlp=hfp, hvp > hgp
c---------------------------------------
```

```
300   a21=a21+dhfdp*a22
      a41=a41+dhfdp*a42
      f2=a22*(hlp1-hfp)
      f4=a42*(hlp1-hfp)
c
c     backward elimination
c
      a43=1./a43
      d42=d42*a43
      a41=a41*a43
      a32=1./a32
      d32=(d32-d42*a33)*a32
      a31=(a31-a41*a33)*a32
      d22=1./(d22-d42*a23)
      a21=(a21-a41*a23)*d22
      d12=d12-d42*a13-d32*a12
      a11=1./(a11-a41*a13-a31*a12-a21*d12)
      ae4=a43*(-a13+a12*a33*a32+a23*d22*d12)
      p11=a11*ae4*c41
      p12=a11*(ae4*f4-f2*d22*d12)
      go to 220
c
c     forward substitution
c
310   f4ii=a43*(c41*dwsu1+f4)
      f3i=-f4ii*a33*a32
      f2i=(f2-f4ii*a23)*d22
      wf1=f2i-a21*dpi1
      dhvp1=f3i-a31*dpi1-d32*wf1
      dalp1=f4ii-a41*dpi1-d42*wf1
      alp=alp1+dalp1
      if(alp.gt.1.) go to 500
      hvp=hvp1+dhvp1
      hlp=hf(pi,dum)
      hgp=hg(pi,dum)
      if(hvp.le.hgp) ierr=7
      go to 1000
c-------------------------------------
c     Condition 4:  hlp < hfp, hvp=hgp
c-------------------------------------
400   f1=a12*(hvp1-hgp)
      f3=a32*(hvp1-hgp)
      a31=a31+dhgdp*a32
      a11=a11+dhgdp*a12
c
c     backward elimination
c
      a43=1./a43
      a41=a41*a43
      a42=a42*a43
      d41=d41*a43
      d31=1./(d31-d41*a33)
      a31=(a31-a41*a33)*d31
```

```
      r32=-a42*a33*d31
      d21=d21-d41*a23
      a22=a22-a42*a23
      a21=a21-a41*a23
      a22=1./(a22-r32*d21)
      a21=(a21-a31*d21)*a22
      d11=d11-d41*a13
      r12=-a42*a13
      a11=a11-a41*a13
      r12=r12-r32*d11
      a11=a11-a31*d11
      a11=1./(a11-a21*r12)
      r3=d31*(d11-d21*a22*r12)
      q11=a11*(-a13+a23*a22*r12+r3*a33)*a43*c41
      q12=a11*(f1-f3*r3)
      go to 220
c
c     forward substitution
c
410   f4i=c41*dwsu1*a43
      f3i=(f3-f4i*a33)*d31
      f2i=(-f4i*a23-f3i*d21)*a22
      dhlp1=f2i-a21*dpi1
      wro=f3i-a31*dpi1-r32*dhlp1
      dalp1=f4i-a41*dpi1-a42*dhlp1-d41*wro
      alp=alp1+dalp1
      if(alp.gt.1.) go to 500
      hlp=hlp1+dhlp1
      hvp=hg(pi,dhgdp)
      hfp=hf(pi,dhfdp)
      go to 1000
c--------------------------------------
c     Condition 5:  empty pressurizer
c--------------------------------------
500   hsu=x3
      a12=smass+(x3-hsu)*dmvdh
      a11=-vvp+(x3-hsu)*dmvdp
      a22=dmvdh
      a21=dmvdp
      f11=dt*(wsp*(hsp-hfp)-
     1    wsc*(hfp-hsu)-(wrv+wsv)*(x3-hsu))
      f21=-dt*(wsc+wrv+wsv)
      a22i=1./a22
      a21i=a21*a22i
      a11i=1./(a11-a21i*a12)
      q11i=-dt*a22i*a12*a11i
      q12i=a11i*(f11-f21*a22i*a12)
      wsu1=(q12i-p12)/(p11-q11i)
      dpi1=p11*wsu1+p12
      wsuv=wsu1+v1p*rolp/dt
      if(wsuv.gt.0.) wsuv=0.
      f21i=(f21+dt*wsuv)*a22i
      f11i=a11i*(f11-f21i*a12)
```

```
      dhv=f21i-a21i*dpi1
      pi=x1+dpi1
      hfp=hf(pi,dhfdp)
      hgp=hg(pi,dhgdp)
      hvp=x3+dhv
      if(hvp.lt.hgp) go to 510
      hlp=hfp
      dp=dpi1
      dhl=0.
      dmw=0.
c       if(wsu1.gt.0.) dmw=dt*wsu1
c       dal=-dmw/rolp/volprz
c       alp=x4+dal
      alp=1.0
      dal=0.
      wfl=0.
      wsu=wsu1
      jsw=5
      go to 1001
510   a11=a11+dhgdp*a12
      a21=a21+dhgdp*a22
      f11=a12*(hvp-hgp)
      f21=a22*(hvp-hgp)
      a12=dt*(hfp-hsu)
      a22=dt
      a22=1./a22
      a21=a21*a22
      a11=1./(a11-a21*a12)
      q11=-dt*a22*a12*a11
      q12=a11*(f11-f21*a22*a12)
      wsu1=(q12-p12)/(p11-q11)
      dpi1=p11*wsu1+p12
      f21i=(f21+dt*wsu1)*a22
      f11i=a11*(f11-f21i*a12)
      wro=f21i-a21*dpi1
      p1=x1+dpi1
      hfp=hf(p1,dhfdp)
      hgp=hg(p1,dhgdp)
      pi=p1
      hvp=hgp
      hlp=hfp
      dhv=0.
      dhl=0.
      dp=dpi1
      alp=1.
      dal=0.
      wmass=0.
      dmw=0.
      wfl=0.
      wsu=wsu1
      jsw=5
      go to 1001
c-------------------------------------------
```

```
c     Condition 6:  insurge in an initially
c                   empty pressurizer
c-----------------------------------------
600   a11i=dmvdp
      a12i=dmvdh
      a13i=dmlda+dmvda
      a21i=a11i
      a22i=a12i
      a23i=dmvda
      a31i=-vvp+(x3-hsu)*dmvdp
      a32i=smass+(x3-hsu)*dmvdh
      a33i=(x3-hsu)*dmvda
      f11i=dt*(wsp-wrv+wsv)
      f12i=-dt*(wsc+wrv+wsv)
      f13i=dt*(wsp*(hsp-hfp)-wsc*(hfp-hsu)-(wrv+wsv)*(x3-hsu))
      a33i=1./a33i
      a31i=a31i*a33i
      a32i=a32i*a33i
      f13i=f13i*a33i
      a22i=1./(a22i-a32i*a23i)
      a21i=(a21i-a31i*a23i)*a22i
      f12i=(f12i-f13i*a23i)*a22i
      a11i=1./(a11i-a31i*a13i-a21i*a12i)
      q11=a11i*dt
      q12=a11i*(f11i-f13i*a13i-f12i*a12i)
      wsu=(q12-p12)/(p11-q11)
      dp=q11*wsu+q12
      dhv=f12i-a12i*dp
      dal=f13i-a31i*dp-a32i*dhv
      pi=x1+dp
      hvp=x3+dhv
      alp=1.+dal
      dhl=0.
      wmass=0.
      dmw=dmlda*dal
      hfp=hf(pi,dum)
      hlp=hsu
      if(hlp.gt.hfp) hlp=hfp
      wfl=0.
      wro=0.
      jsw=6
      go to 1001
700   pi=p1
      hlp=hlp1
      hvp=hvp1
      alp=alp1
      wsu=wsu1
1000  if(alp.ge.1.) go to 25
      if(alp.lt.0.) alp=0.0
      dp=pi-x1
      dhl=hlp-x2
      dhv=hvp-x3
      dal=alp-x4
```

```
1001  wsum=wsum+dt*wsu
c-----------------------------------
c     update pressurizer properties
c-----------------------------------
      call state(pi,hlp,hvp,hfp,hgp,tsat,dtsdp,tlp,tvp,rolp,rovp,
     1     drldp,drldh,drvdp,drvdh,rof,rog,drfdp,drgdp,dhfdp,dhgdp,
     2     cplp,cpvp,vislp,visvp,cndlp,cndvp,sig)
      hfg=hgp-hfp
      vwall=vwall+acc*wall*dal
      vvp=alp*volprz
      vlp=volprz-vvp
      dms=vvp*(drvdp*dp+drvdh*dhv)+volprz*rovp*dal
      if(jsw.gt.4) go to 114
      dmw=vlp*(drldp*dp+drldh*dhl)-volprz*rolp*dal
114   smass=smass+dms
      wmass=wmass+dmw
      smass=vvp*rovp
      wmass=vlp*rolp
      if(smass.lt.0.) smass=0.
      if(wmass.lt.0.) wmass=0.
      dmh=smass*hvp+wmass*hlp
      gain=dt*(wsu*hsu-wwc*hfg+qh)
      if(gain.eq.0.) go to 115
      uerr=(dmh-dmhi-volprz*dp-gain)/gain
      go to 116
115   uerr=dmh-dmhi-volprz*dp
116   dmhi=dmh
c  -- calculate water level --
      wlevel=r+(vlp-vsp)/ac
      if(vlp.lt.vsp.and.vlp.gt.0.) wlevel=(1.5*vlp/pei)**0.3333
      if(vlp.eq.0.) wlevel=0.
      wlr=100.*wlevel/wall
c-----------------------------------------------------------------
c     forward substitution:  flow path component state equations
c-----------------------------------------------------------------
      if(dmi(1).lt.0.) go to 805
      g5(1)=e53(1)*(g5(1)-dt*wsu*(hsu-hhl(1)))
      g6(1)=g6(1)-g5(1)*e62(1)
      g7(1)=g7(1)-g6(1)*e72(1)
      go to 806
805   g7(1)=g7(1)-dt*wsu*(hsu-hhl(1))
      g7(1)=e73(1)*g7(1)-g6(1)*e72(1)
806   g4i=0.
      do 807 j=2,m
807   g4i=g4i-g7(j)*e44(j)
      g4=e43*(g4-g7(1)*e44(1)-g4i)
      g2i=0.
      do 800 i=1,m
800   g2i=g2i+(g7(i)*e23(i))
      g2=e22*(g2-g2i-g4*r23-g3*r22)
      dhip=g2-e21*dp
      dhcr=g3-e31*dp-e32*dhip
      dhep=g4-e41*dp-r42*dhip-e42*dhcr
```

```
      do 810 i=1,m
      if(dmi(i).lt.0.) go to 819
      dhhl(i)=g5(i)-e51(i)*dp-e52(i)*dhep
      dhsg(i)=g6(i)-e61(i)*dp-r62(i)*dhep
      dhcl(i)=g7(i)-e71(i)*dp-r72(i)*dhep
      go to 810
819   dhcl(i)=g5(i)-e51(i)*dp-r51(i)*dhip
      dhsg(i)=g6(i)-e61(i)*dp-r61(i)*dhip
      dhhl(i)=g7(i)-e71(i)*dp-r71(i)*dhip
810   continue
      hsu=hhl(1)
c
c     update flow-path-component state variables
c
      hip=y2+dhip
      hcr=y3+dhcr
      hep=y4+dhep
      do 820 i=1,m
      hhl(i)=y5(i)+dhhl(i)
      hsg(i)=y6(i)+dhsg(i)
      hcl(i)=y7(i)+dhcl(i)
      qisg(i)=qisg(i)+dqsg(i)*dhsg(i)+dqhl(i)*dhhl(i)+dqsgdp(i)*dp
820   continue
      if(wsu.lt.0.) hsu=hlp
      if(alp.eq.1.) hsu=hvp
c-------------------------------------------------------------
c     calculate mixing-cup enthalpies and update properties
c-------------------------------------------------------------
      admic=abs(dmic)
      call hflow(pi,hip,admic,arip,dip,hfg,hfp,hgp,rof,rog,drfdp,
     &     drgdp,sig,hflip,tip,dtipdp,dtipdh,alpip,xip,roip,drdpip,
     &     drdhip,dhfmip,5)
      call hflow(pi,hcr,admic,arcr,dcr,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hflcr,tcr,dtcrdp,dtcrdh,alpcr,xcr,rocr,drdpcr,
     2     drdhcr,dhfmcr,1)
      call hflow(pi,hep,admic,arep,dep,hfg,hfp,hgp,rof,rog,drfdp,
     &     drgdp,sig,hflep,tep,dtepdp,dtepdh,alpep,xep,roep,drdpep,
     &     drdhep,dhfmep,5)
c
c     calculate state equation coeficients
c
c --  core energy  --
      call coef(pi,dmic,hfp,hcr,hip,hep,rocr,roip,roep,
     &     alpcr,alpip,drdpcr,drdpip,
     1     drdpep,drdhcr,vcr,depcr,dehcr,dmpcr,dmhcr,dmhcr1,rmcr)
      do 830 i=1,m
      admi=abs(dmi(i))
      call hflow(pi,hhl(i),rn*admi,arhl,dhl,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hflhl(i),thl(i),dthldp(i),dthldh(i),alphl(i),
     2     xhl(i),rohl(i),drdphl(i),drdhhl(i),dhfmhl(i),2)
      call hflow(pi,hsg(i),rn*admi,arsg,dsg,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hflsg(i),tsg(i),dtsgdp(i),dtsgdh(i),alpsg(i),
     2     xsg(i),rosg(i),drdpsg(i),drdhsg(i),dhfmsg(i),3)
```

```
      call hflow(pi,hcl(i),admi,arcl,dcl,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hflcl(i),tcl(i),dtcldp(i),dtcldh(i),alpcl(i),
     2     xcl(i),rocl(i),drdpcl(i),drdhcl(i),dhfmcl(i),4)
c --  hot-leg energy  --
      call coef(pi,rn*dmi(i),hfp,hhl(i),hep,hsg(i),rohl(i),roep,
     1     rosg(i),alphl(i),alpep,drdphl(i),drdpep,drdpsg(i),
     &     drdhhl(i),vhl,dephl(i),
     &     dehhl(i),dmphl(i),dmhhl(i),dmhhl1(i),rmhl(i))
c --  steam generator energy  --
      call coef(pi,rn*dmi(i),hfp,hsg(i),hhl(i),hcl(i),rosg(i),rohl(i),
     1     rocl(i),alpsg(i),alphl(i),drdpsg(i),drdphl(i),drdpcl(i),
     &     drdhsg(i),vsg,
     &     depsg(i),dehsg(i),dmpsg(i),dmhsg(i),dmhsg1(i),rmsg(i))
c --  cold-leg energy  --
      call coef(pi,dmi(i),hfp,hcl(i),hsg(i),hip,rocl(i),rosg(i),
     1     roep,alpcl(i),alpsg(i),drdpcl(i).drdpsg(i),drdpcl(i),
     &     drdhcl(i),vcl,
     &     depcl(i),dehcl(i),dmpcl(i),dmhcl(i),dmhcl1(i),rmcl(i))
830   continue
      tca=0.5*(tcr+tip)
      delt=tep-tip
      deltr=delt/delt0
      roa=0.5*(rocr+roip)
c  -- update thermal power ratio --
      qithp=(qith/qi0)*100.
c  -- calculate thermal capacities --
      cplcr=cpl(hcr,pi)
      cplip=cpl(hip,pi)
      cpla=0.5*(cplcr+cplip)
      do 850 i=1,m
      cplhl=cpl(hhl(i),pi)
      cplsg=cpl(hsg(i),pi)
      cplcl=cpl(hcl(i),pi)
850   cplal(i)=0.5*(cplsg+cplhl)
c-----------------------------
c     calculate bouyancy head
c-----------------------------
      do 840 i=1,m
      dpb(i)=g*(-2.*rmcr*zc0-roep*zep-rmhl(i)*(zs-zep-2.*zc0)
     &        +bsg+rmsg(i)*(zs-zep-2.*zc0)+(rmcl(i)-rmsg(i))*zcl
     &        +roip*(2.*zc0+zep))
        if(dpb(i).lt.0.) dpb(i)=-dpb(i)
c       dpb(i)=g*(-(rocr+roip)*zc0-roep*zep+0.5*(roep+
c     &      rohl(i))*(zs-2.*zc0-zep)+bsg+
c     &       (rocl(i)-rosg(i))*zcl+0.5*(rosg(i)+rocl(i))*
c     &       (zs-2.*zc0-zep)+rcip*(zep+2.*zc0))
      if(dmi(i).lt.0.) go to 842
      zcsg=bsg/(rosg(i)-rohl(i))
      go to 840
842   zcsg=bsg/(rosg(i)-rohl(i))
840   zc(i)=zs+zcsg-zc0
c--------------------------------------------
c     calculate mass flow rate and pump speed
```

```
c------------------------------------------
      call pump(dmi,dmic,wi,ti,dt,roip,rocr,roep,rohl,rosg,rocl,
     &     xip,xcr,xep,xhl,xsg,xcl,pi,
     &     dpb,dpl,hc,trip,vislp)
c--------------------------------
c     calculate new time step size
c--------------------------------
      call time(dt,dtmax,vcr,rocr,rof,dmic,alp)
c------------------
c     print output
c------------------
      call output(ti,pi,qi,qisg,qith,wi,dmi,dmic,tfa,tca,tip,tcr,
     &     tep,thl,tsg,tcl,tsec,alpip,alpcr,alpep,alphl,alpsg,alpcl,
     &     zc,deltr,tsat,tlp,tvp,wlevel,qh,alp,wlr,wfl,wro,wwc,
     &     wsv,wrv,wsp,wcharg,wsu,wsum,dpl,dpb,hc,jsw,jflag,iflag,
     &     wbr,wsi,m,n,isw)
      jsw=1
      n=n+1
      if(ti.le.t) go to 10
      return
      end
```

```
      subroutine table(fx,x,f,y,n)
c------------------------------
c     perform tabular look-up
c------------------------------
c         fx    = quantity to be found
c         x     = independent variable
c         f     = array containing ordinate values
c         y     = array containing abcissa values
c         n     = number of values in f or y arrays
c
      dimension f(1),y(1)
      do 120 i=1,n
      if(x-y(i)) 130,110,120
110   if(i.eq.n) go to 140
120   continue
      go to 170
130   if(i.eq.1) go to 160
140   slope=(x-y(i-1))/(y(i)-y(i-1))
150   fx=f(i-1)+slope*(f(i)-f(i-1))
      return
160   fx=1.0
      return
170   fx=f(n)
      return
      end
```

```
      subroutine coef(pi,dmi,hfp,hm,hm1,hm0,rom,rom1,rom0,alpi,alp1,
     1            drdpm,drdpm1,drdpm0,drdhi,vi,depi,dehi,dmpi,dmhi,
     &            dmhi1,mi)
c-----------------------------------------------------------------
c     calculate coefficients of energy and mass equations
c-----------------------------------------------------------------
c     return variables:  depi,dehi,dmpi,dmhi,dmhi1,mi
c
      data dpi/1.e4/
      real mi,mip
c
      hi=hm
      roi=rom
      drdpi=drdpm
      if(dmi.lt.0.) go to 5
      hi1=hm1
      roi1=rom1
      drdpi1=drdpm1
      go to 6
5     hi1=hm0
      roi1=rom0
      drdpi1=drdpm0
6     dh=hi-hi1
      if(dh.eq.0.) go to 100
      if(hi.lt.hfp.and.hi1.lt.hfp) go to 10
      if(hi.gt.hfp.and.hi1.lt.hfp) go to 20
      if(hi.lt.hfp.and.hi1.gt.hfp) go to 30
      call vp2(pi,hi,hi1,dh,roi,roi1,alpi,alp1,mi,ei)
      call vp2(pi+dpi,hi,hi1,dh,roi,roi1,alpi,alp1,mip,eip)
40    dmdpi=(mip-mi)/dpi
      dedpi=(eip-ei)/dpi
      go to 50
20    call vp1(pi,hi,hi1,dh,roi,roi1,alpi,mi,ei)
      call vp1(pi+dpi,hi,hi1,dh,roi,roi1,alpi,mip,eip)
      go to 40
30    call vp1(pi,hi1,hi,dh,roi1,roi,alp1,mi,ei)
      call vp1(pi+dpi,hi1,hi,dh,roi1,roi,alp1,mip,eip)
      go to 40
c
c     single-phase condition
c
10    mi=0.5*(roi+roi1)
      ei=((2.*hi+hi1)*roi+(2.*hi1+hi)*roi1)/6.
      dmdpi=0.5*(drdpi+drdpi1)
      dedpi=((2.*hi+hi1)*drdpi+(2.*hi1+hi)*drdpi1)/6.
50    depi=vi*(dedpi-hi*dmdpi-1.)
      dmdhi=(roi-mi)/dh
      dmdhi1=(mi-roi1)/dh
      dmpi=vi*dmdpi
      dmhi=vi*dmdhi
      dmhi1=vi*dmdhi1
      dehi=vi*0.5*(roi+roi1)
      return
```

```
100   mi=roi
      ei=hi*roi
      dmdpi=drdpi
      dedpi=hi*drdpi
      dmdhi=drdhi
      dedhi=hi*drdhi+roi
      dmpi=vi*dmdpi
      depi=vi*(dedpi-hi*dmdpi-1.)
      dmhi=vi*dmdhi
      dehi=vi*0.5*(roi+roi1)
      return
      end
```

```
      subroutine vp1(pi,hi,hi1,dh,roi,roi1,alpi,mi,ei)
c-----------------------------------------------------
c     calculate volume averaged density and specific
c     energy under the condition: hi1 < hfp < hi
c-----------------------------------------------------
c     output variables:  mi,ei
c
      real mi
      hfp=hf(pi,dum)
      hgp=hg(pi,dum)
      rof=rol(hfp,pi,dum,dum1)
      rog=rov(hgp,pi,dum,dum1)
      romi=alpi*rog+(1.-alpi)*rof
      if(alpi.eq.0.) romi=roi
      beta=(hfp-hi1)/dh
      vi=1./romi
      vf=1./rof
      dvi=vi-vf
      mi=0.5*beta*(roi1+rof)+(1.-beta)*alog(vi/vf)/dvi
      ei=beta*((2.*hfp+hi1)*rof+(2.*hi1+hfp)*roi1)/6.+(1.-beta)
     1   *((hi-hfp)+(hfp*vi-hi*vf)*alog(vi/vf)/dvi)/dvi
      return
      end
```

```
      subroutine vp2(pi,hi,hi1,dh,roi,roi1,alpi,alp1,mi,ei)
c-------------------------------------------------------------
c     calculate volume averaged density and specific energy
c     under the condition:  hfp < hi, and hfp < hi1
c-------------------------------------------------------------
c     return variables:  mi,ei
c
      real mi
      hfp=hf(pi,dum)
      hgp=hg(pi,dum)
      rof=rol(hfp,pi,dum,dum1)
      rog=rov(hgp,pi,dum,dum1)
      romi=alpi*rog+(1.-alpi)*rof
      rom1=alp1*rog+(1.-alp1)*rof
      vi=1./romi
      vi1=1./rom1
      dvi=vi-vi1
      mi=alog(vi/vi1)/dvi
      ei=(dh+(hi1*vi-hi*vi1)*alog(vi/vi1)/dvi)/dvi
      return
      end
```

```
      subroutine hflow(pi,hi,admi,ari,di,hfg,hfp,hgp,rof,rog,drfdp,
     1     drgdp,sig,hfli,tempi,dtdp,dtdh,alpi,xi,roi,drdpi,drdhi,
     2     dhfdhm,jcomp)
c-------------------------------------------------
c     calculate mixing-cup enthalpy, void fraction,
c     mixture density and their partial derivatives
c-------------------------------------------------
c     return variables:  hfli,tempi,alpi,xi,roi,
c                        drdpi,drdhi,dhfdhm
c
c     jcomp:  1 = core
c             2 = hot leg
c             3 = steam generator
c             4 = cold leg
c             5 = core inlet or exit plenum
c
c     variables:
c
c         hfli   = mixing-cup enthalpy
c         dhfdhm = partial derivative of hlfi wrt hi
c         dhpdhm = dhfdhm, in two-phase condition
c                = 0, in single-phase condition
c         c0     = void concentration factor
c         ugj    = drift velocity
c         xi     = flow quality
c         y      = static quality
c
      if(hi.le.hfp) go to 500
      if(jcomp.gt.4) go to 100
c
c     two-phase condition:  drift flux model
c
      rgf=rog/rof
      y=(hi-hfp)/hfg
      if(y.gt.1.) y=1.
      alpi=rof*y/((rof-rog)*y+rog)
      roi=(1.-alpi)*rof+alpi*rog
      drdpi=(1.-alpi)*drfdp+alpi*drgdp
      drdhi=0.
      tempi=tl(hfp,pi,dtdp,dtdh)
      call drift(ugj,c0,rof,rog,sig,pi,di,jcomp)
      xi=alpi*(c0*rgf+rog*ugj*ari/admi)/(1.-c0*(1.-rgf)*alpi)
      if(xi.gt.1.) xi=1.0
      hfli=hfp+xi*hfg
      denom=1.-(1.-c0)*(1.-1./rgf)*((hi-hfp)/hfg)
      dhfdhm=(c0+rog*ugj*ari/(admi*rgf))/(denom*denom)
      dhpdhm=dhfdhm
      return
c
c     perfect mixing model (two-phase mixture)
c
100   xi=(hi-hfp)/hfg
      if(xi.gt.1.) xi=1.0
```

```
      alpi=xi
      roi=xi*rog+(1.-xi)*rof
      drdpi=xi*drgdp+(1.-xi)*drfdp
      drdhi=0.
      tempi=tl(hfp,pi,dtdp,dtdh)
      hfli=hi
      dhfdhm=0.
      return
c
c     single-phase condition
c
500   hfli=hi
      dhfdhm=1.
      roi=rol(hi,pi,drdpi,drdhi)
      tempi=tl(hi,pi,dtdp,dtdh)
      alpi=0.
      xi=0.
      return
      end
```

```
      subroutine heater(pi,qh,tlp,wlevel,dt,isw)
c-----------------------------------------------------------
c     determine proportional and backup heater power output
c-----------------------------------------------------------
c     output variables:  qh
c
c     parameters:
c         iswp  = proportional heater switch
c                 (0/1) (on/off)
c         iswb  = backup heater switch
c                 (0/1) (on/off)
      common /heat/tau1,wlmin,qp,qb,pph,ppl,pbh,pbl,vh,ah,hh
      qpi=qp
      qbi=qb
      if(qh.eq.0.) th=tlp
      if(isw.eq.0) go to 5
      qh=0.0
      return
c
c     water level test
c
5     if(wlevel.gt.wlmin) go to 10
      iswp=1
      iswb=1
      go to 50
c
c     pressure test
c
10    if(pi.lt.ppl) go to 20
      iswp=1
      iswb=1
      go to 50
20    iswp=0
      iswb=0
50    if(iswp.eq.1) qpi=0.0
      if(iswb.eq.1) qbi=0.0
      qh=qpi+qbi
c      qi=qpi+qbi
c      ha=hh*ah
c      rcp=vh*ross(th)*cpss(th)
c      th=((qi+ha*tlp)*dt+rcp*th)/(rcp+ha*dt)
c      qh=ha*(th-tlp)
c      if(qh.lt.0.) qh=0.
      return
      end
```

```
      subroutine spray(pi,wspray,wlevel,hc,dpl,roc,ros)
c-------------------------------------
c     calculate spray mass flow rate
c-------------------------------------
c     Parameters:
c         hsv   = elevation head of the spray valve
c         hlv   = elevation head of the water level
c
      common /sprays/wspmax,asp,hesp,hisp,psh,psl
      common /spv/cvmax
      wspray=0.0
      if(pi.lt.psl) return
      if(pi.lt.psh) go to 10
      if(pi.ge.psh) cv=cvmax
      go to 20
10    pipc=(pi-psl)/(psh-psl)
      cv=cvmax*pipc
      cv=cvmax*pipc
      if(cv.le.0.) return
20    hsv=roc*9.81*hesp
      hlv=ros*9.81*(hisp+wlevel)
      dpch=hc-dpl
      dpsp=dpch+hlv-hsv
      if(dpsp.lt.0.) return
      wspray=sqrt(asp*asp*dpsp/cv)
      return
      end
```

```
      subroutine charge(wcharg,wld,hcl,hcharg,wmass,qithp,wv0,
     &            rol,dt,nloop)
c-------------------------------------------------
c     calculate charge or letdown mass flow rate
c-------------------------------------------------
c     water level is controlled to track the set-
c     value defined by power level
c
c     output variables:  wcharg,wd,hcharg
c
      common /charge/gpmc,gpml,hch0
      hcharg=hch0
c
c     pressurizer water volume vs. power set-point
c
      vwp=57.5+0.497*(qithp-14.5)
      if(qithp.gt.100.) vwp=100.
      if(qithp.lt.14.5) vwp=57.5
      vwi=wv0*vwp/100.
      wmi=vwi*rol
      dwmi=wmi-wmass
      wcharg=dwmi/dt
      if(wcharg.gt.0.) go to 10
c
c     let-down
c
      if(abs(wcharg).gt.gpml) wld=gpml
      wcharg=0.
      return
c
c     charge
c
10    if(wcharg.gt.gpmc) wcharg=gpmc
      hcharg=hch0
      wld=0.
      return
      end
```

```
      subroutine valve(pi,dt,arv,asv1,asv2,asv3)
c-----------------------------------------------------------------
c     calculate valve openings of the safety and relief valves
c-----------------------------------------------------------------
c     return variables:  arv,asv1,asv2,asv3
c
      common /valves/amax,prv,psv1,psv2,psv3
      if(pi.ge.prv) go to 10
      arv=0.0
11    asv1=0.0
12    asv2=0.0
13    asv3=0.0
      return
10    arv=amax
      if(pi.ge.psv1) go to 20
      go to 11
20    asv1=amax
      if(pi.ge.psv2) go to 30
      go to 12
30    asv2=amax
      if(pi.ge.psv3) go to 40
      go to 13
40    asv3=amax
      return
      end
```

```
      subroutine fulpin(dt,qi,qith,tfa,tca,hcr,hf,dmic,pi,
     &     cpl,isw)
c----------------------------------------
c     calculate average fuel rod temperature
c----------------------------------------
c
c     return variables:  tfa, qith
c
      common/fuel/fuell,cdlo,dcli,dfo,rodn,fmass,clmass,ftd,fpuo2,
     &             hgap,afuel
      if(isw.gt.0) go to 100
      call fresis(tfa,tclo,cpmr,urod,trod,qi,tca,hcr,hf,pi,dmic,
     &     cpl,dt)
      qith=afuel*urod*(tfa-tca)
      return
c
c     calculate steady-state average fuel rod temperature
c
100   i=0
      tclo=tfa
110   call fresis(tfa,tclo,cpmr,urod,trod,qi,tca,hcr,hf,pi,dmic,
     &     cpl,dt)
      ftfa=(tfa-tca)-qi/(afuel*urod)
      tfai=tfa+10.
      call fresis(tfai,tclo,cpmr,urod,trod,qi,tca,hcr,hf,pi,dmic,
     &     cpl,dt)
      ftfai=(tfai-tca)-qi/(afuel*urod)
      ft=(ftfai-ftfa)/10.
      dtfa=ftfa/ft
      tfa=tfa-dtfa
      if(abs(dtfa).lt.10.) return
      i=i+1
      if(i.gt.10) go to 200
      go to 110
200   write(6,201)
201   format(//,' Warning!!!  Fuel Temperature Iteration Did Not',
     &        ' Converge.')
      return
      end
```

```
      subroutine fresis(tfa,tclo,cpmr,urod,trod,qi,tca,hcr,hf,pi,
     &          dmic,cpl,dt)
c-----------------------------------------------
c     calculate fuel-clad thermal resistance
c-----------------------------------------------
c     output variables:  tfa,tclo,trod,urod
c
c         trod   = fuel-clad heat transfer time constant
c         cpmr   = fuel-clad thermal inertia
c         tclo   = clad outside surface temperature
c
      common/fuel/fuell,dclo,dcli,dfo,rodn,fmass,clmass,ftd,fpuo2,
     &           hgap,afuel
      common/prime/vip,vcr,vep,vhl,vsg,vcl,
     &     arip,arcr,arep,arhl,arsg,arcl,
     &     dip,dcr,dep,dhl,dsg,dcl
      data pei/3.1416/
      call mpf(tfa,ftd,fpuo2,cpf,cndf)
      call mpc(tfa,cpcl,cndcl)
      fm=fmass/afuel
      clm=clmass/afuel
      cpfm=fm*cpf
      cpclm=clm*cpcl
      rm=fm+clm
      cpmr=cpfm+cpclm
c----------------------------------------------
c     calculate heat transfer coefficients
c----------------------------------------------
      ucl=1.*cndss(tfa)/(dclo*alog(dclo/dcli))
      ug=hgap*dcli/dclo
      uf=4.*cndf/dclo
      ut=1./(1./ucl+1./ug+1./uf)
      f1=ut/ucl
      f2=ut/ucl+ut/ug
      fr=0.5*(cpclm*f1+cpfm*(f2+1.))/cpmr
      if(hcr.le.hf)
     & call htco(tca,dmic,arcr,pi,hcr,cpl,dcr,1,htc)
      if(hcr.gt.hf)
     & call htco((tclo-tca),dmic,arcr,pi,hcr,cpl,dcr,2,htc)
      urod=1./(fr/ut+1./htc)
      trod=cpmr/urod
c---------------------------------
c     calculate new temperatures
c---------------------------------
      dtfa=(dt*qi/(afuel*cpmr)-dt*(tfa-tca)/trod)/(1.+dt/trod)
      tfa=tfa+dtfa
      tclo=tfa-fr*qi/urod
      return
      end
```

```
      subroutine mpf (tem, ftd, fpuo2, rcp, cond)
c-----------------------------------------------------------------------
c  calculates heat capacity and conductivity of U02 and PU02 fuels as
c    functions of temperature, fraction of theoretical density, and
c    plutonium content
c-----------------------------------------------------------------------
c  arguments
c    input      tem      temperature (deg K)
c               ftd      fraction of theoretical density
c               fpuo2    plutonium fraction by volume
c    return     rcp     heat capacity (J/m**3-deg K)
c               cond     conductivity (W/m-deg K)
c
c  This subroutine is based on expressions used in MATPRO; see
c    TREE-NUREG-1005, Appendix A. Those expressions have been approxi-
c    mated by polynomial fits whose maximum errors are about one
c    standard deviation in experimental data.
c       rcp error =  2 per cent       300 < tem < 3000 deg K
c       cond error = 10 per cent       400 < tem < 2500 deg K
c
      dimension rc(4), rcm(4), cn(3), cnm(3)
      data  rc /1.78e6, 3.62e3, -2.61, 6.59e-4/
      data rcm /1.81e6, 3.72e3, -2.57, 6.13e-4/
      data  cn /10.8, -8.84e-3, 2.25e-6/
      data cnm /9.88, -8.44e-3, 2.25e-6/
      data rouo2,ropuo2/1.097e4,1.146e4/
c
      rof=ftd*((1.-fpuo2)*rouo2+fpuo2*ropuo2)
      if (fpuo2.gt.1.e-7)  go to 20
c------------
c  U02 fuel
c------------
   10 rcp = ftd*( rc(1)+ tem*(rc(2) +tem*(rc(3) +tem*rc(4))) )
      if(tem.gt.3.e3) rcp=ftd*6.943e6
      rcp=rcp/rof
      bt = 2.74 - tem * 5.8e-4
      if(bt.lt.0.) bt=0.
      por = 1.- bt*(1.- ftd)
c--the factor  /(1.-bt*(1.-.95)) is incorporated in the fit cn(3)
      cond = por*( cn(1)+ tem*(cn(2)+ tem*cn(3)) )
      go to 100
c--------------------
c  mixed oxide fuel
c--------------------
   20 rcp = ftd *(1.+.045+fpuo2) *
     *         (rcm(1)+ tem*(rcm(2)+ tem*(rcm(3)+ tem*rcm(4))) )
      rcp=rcp/rof
      bt = 2.74 - tem * 5.8e-4
      por = ftd / (1.+ bt*(1.-ftd))
c         the factor (1.+bt*(1.-.96))/.96 is incorporated in cnm(3)
      cond = por*( cnm(1)+ tem*(cnm(2)+ tem*cnm(3)) )
c
c
```

```
100 continue
    return
    end
```

```
      subroutine mpc (tem, rcp, cond)
c------------------------------------------------------------
c  calculates heat capacity and conductivity of zircaloy as
c  a function of temperature
c------------------------------------------------------------
c  arguments
c    input      tem      temperature (deg K)
c    return     rcp     heat capacity (J/m**3-deg K)
c               cond     conductivity (W/m-deg K)
c
c  This subroutine is based on data in TREE-NUREG-1005, Appendix B.
c    Conductivity is used unchanged. Heat capacity has been fit
c    linearly in the alpha phase (tem < 1190), by a constant in the
c    beta phase (tem > 1254), and by an inverted vee in the transition.
c    Error is 5 per cent in the alpha phase, 300 < tem < 1190 deg K.
c
      dimension cn(4)
      data cn /7.51, 2.09e-2, -1.45e-5, 7.67e-9 /
      data roz/6.5514e3/
c----------------
c  heat capacity
c----------------
      if (tem.gt.1090.)  go to 20
c  alpha phase:    (0 < tem < 1090 deg K, usual case)
      rcp = 1673456. + tem * 721.6
      rcp=rcp/roz
      go to 50
   20 if (tem.ge.1254.)  go to 30
        rcp = 5346400. - 36080.*abs(tem-1170.)
        rcp=rcp/roz
        go to 50
   30   rcp = 2315680.
        rcp=rcp/roz
   50 continue
c---------------
c  conductivity
c---------------
      cond = cn(1)+ tem*(cn(2)+ tem*(cn(3)+ tem*cn(4)))
      return
      end
```

```
      subroutine htco(t,dmi,aflow,pi,hli,cpl,de,ic,htc)
c-----------------------------------------
c     calculate heat transfer coefficient
c-----------------------------------------
c     output variable:  htc
c         htc  = heat transfer coefficient (W/m2-K)
c
c         re   = Reynold number
c         pr   = Prandl number
c         ic   = 1:  Dittus-Bolter
c                2:  Thom
c
      if(ic.eq.2) go to 10
c
c     Dittus-Boelter correlation
c
      visli=visl(hli,pi)
      re=abs(dmi)*de/(aflow*visli)
      cndli=cndl(hli,pi)
      pr=cpl*visli/cndli
      htc=0.023*(cndli/de)*re**0.8*pr**0.4
      return
c
c     Thom correlation
c
10    htc=949.*t*exp(2.*pi/8.7e6)
      return
      end
      function ross(tw)
c----------------------------------------
c     density of stainless steel, kg/m3
c----------------------------------------
      data a0,a1,a2,t0,r0/1.7887e-5,2.3977e-9,3.2692e-13,298.15,
     1      7.8e3/
      at=a0+tw*(a1+tw*a2)
      ross=r0/(1.+at*(tw-t0))**3
      return
      end
      function cpss(tw)
c-------------------------------------------------
c     thermal capacity of stainless steel, J/Kg-K.
c-------------------------------------------------
      data c0,c1,c2,c3/380.962,0.535104,-6.10413e-4,3.02469e-7/
      cpss=c0+tw*(c1+tw*(c2+tw*c3))
      return
      end
      function cndss(tw)
c------------------------------------------
c     thermal conductivity of stainless steel
c------------------------------------------
      data c0,c1,c2,c3/9.01748,1.62997e-2,-4.80329e-6,2.18422e-9/
      cndss=c0+tw*(c1+tw*(c2+tw*c3))
      return
```

```
      end
      function rcin(tk)
c-----------------------------------------
c     determines the product of density and
c     thermal capacity of Inconel 600
c-----------------------------------------
c     rcin [=] J/m3-K
c     data base is within the range:
c     422 K < tk < 755 K
c
      data ce1,ce2/1.3677e3,3.3663e6/
      rcin=ce1*tk+ce2
      return
      end
```

```
      subroutine state(pi,hl,hv,hfi,hgi,tsat,dtsdp,tli,tvi,roli,
     1  rovi,drldp,drldh,drvdp,drvdh,rof,rog,drfdp,drgdp,dhfdp,dhgdp,
     2  cpli,cpvi,visli,visvi,cndli,cndvi,sig)
c-----------------------------------------------------
c     determine properties and their derivatives
c-----------------------------------------------------
c     input:  pi,hl,hv
c
      hfi=hf(pi,dhfdp)
      hgi=hg(pi,dhgdp)
      tsat=tl(hfi,pi,dtsdp,dtsdh)
      tli=tl(hl,pi,dtldp,dtldh)
      roli=rol(hl,pi,drldp,drldh)
      rovi=rov(hv,pi,drvdp,drvdh)
      rof=rol(hfi,pi,drfdp,dum)
      rog=rov(hgi,pi,drgdp,dum)
      cpli=cpl(hl,pi)
      cpvi=cpv(hv,pi)
      tvi=tsat+(hv-hgi)/cpvi
      visli=visl(hl,pi)
      visvi=visv(hv,pi,tvi,rovi)
      cndli=cndl(hl,pi)
      cndvi=cndv(hv,pi,tvi,rovi)
      sig=surten(tli)
      return
      end
      function hf(pi,dhfdp)
c-----------------------------------------
c     enthalpy of saturated liquid, J/Kg.
c-----------------------------------------
      data a0,a1,a2,a3,a4,a5/5.7474e5,0.209206,-2.8051e-8,
     1      2.38098e-15,-1.0042e-22,1.6587e-30/
      hf=a0+pi*(a1+pi*(a2+pi*(a3+pi*(a4+pi*a5))))
      dhfdp=a1+pi*(2.*a2+pi*(3.*a3+pi*(4.*a4+5.*pi*a5)))
      return
      end
      function hg(pi,dhgdp)
c----------------------------------------
c     enthalpy of saturated vapor, J/Kg.
c----------------------------------------
      data b0,b1,b2,b3,b4/2.7396e6,3.7588e-2,-7.164e-9,
     1      4.2002e-16,-9.8507e-24/
      hg=b0+pi*(b1+pi*(b2+pi*(b3+pi*b4)))
      dhgdp=b1+pi*(2.*b2+pi*(3.*b3+4.*pi*b4))
      return
      end
      function tl(hl,pi,dtldp,dtldh)
c------------------------------------------
c     temperature of compressed liquid, K.
c------------------------------------------
      data c00,c01,c10,c11,c20,c21,c30,c31/2.7291e2,-1.5954e-7,
     1      2.3949e-4,-5.1963e-13,5.966e-12,1.2064e-18,-1.3147e-17,
     2      -5.6026e-25/
```

```
      c0=c00+c01*pi
      c1=c10+c11*pi
      c2=c20+c21*pi
      c3=c30+c31*pi
      tl=c0+hl*(c1+hl*(c2+hl*c3))
      dtldp=c01+hl*(c11+hl*(c21+hl*c31))
      dtldh=c1+hl*(2.*c2+3.*hl*c3)
      return
      end
      function hl(tl,hl,pi)
c-----------------------------------------------------------
c     calculate liquid enthalpy given liquid temperature,
c     pressure, and initial guess using Newton iteration
c-----------------------------------------------------------
      data c00,c01,c10,c11,c20,c21,c30,c31/2.7291e2,-1.5954e-7,
     1     2.3949e-4,-5.1963e-13,5.966e-12,1.2064e-18,-1.3147e-17,
     2     -5.6026e-25/
      n=0
      hl=hl
      c0=c00+c01*pi
      c1=c10+c11*pi
      c2=c20+c21*pi
      c3=c30+c31*pi
10    f=tl-c0-hl*(c1+hl*(c2+hl*c3))
      dfdh=-c1-hl*(2.*c2+3.*hl*c3)
      dh=f/dfdh
      if(abs(dh).lt.10.) return
      hl=hl-dh
      n=n+1
      if(n.gt.10) return
      go to 10
      end
      function rol(hl,pi,drldp,drldh)
c--------------------------------------------
c     density of compressed liquid, Kg/m3.
c--------------------------------------------
      data f10,f11,f20,f21,f30,f31,f40,f41,f50,f51,f60,f61,hl0/
     1     999.65,4.9737e-7,-2.5847e-10,6.1767e-19,1.2696e-22,
     2     -4.9223e-31,1488.64,1.3389e-6,1.4695e9,8.85736,
     3     3.20372e6,1.20483e-2,6.513e5/
      if(hl.gt.hl0) go to 10
      f1=f10+pi*f11
      f2=f20+pi*f21
      f3=f30+pi*f31
      rol=f1+hl*hl*(f2+hl*hl*f3)
      drldp=f11+hl*hl*(f21+hl*hl*f31)
      drldh=2.*hl*(f2+2.*hl*hl*f3)
      return
10    f4=f40+pi*f41
      f5=f50+pi*f51
      f6=f60+pi*f61
      rol=f4+f5/(hl-f6)
      drldp=f41+(f51*(hl-f6)+f5*f61)/((hl-f6)*(hl-f6))
```

```
      drldh=-(rol-f4)/(hl-f6)
      return
      end
      function rov(hv,pi,drvdp,drvdh)
c-------------------------------------------
c     density of superheated vapor, Kg/m3.
c-------------------------------------------
      data g00,g01,g02,g10,g11,g12/-5.1026e-5,1.1208e-10,
     1      -4.4506e5,-1.6893e-10,-3.398e-17,0.23058/
      g0=g00+pi*g01+g02/pi
      g1=g10+pi*g11+g12/pi
      dg0=g01-g02/(pi*pi)
      dg1=g11-g12/(pi*pi)
      rov=1./(g0+g1*hv)
      drvdp=-(rov*rov)*(dg0+hv*dg1)
      drvdh=-(rov*rov)*g1
      return
      end
      function cpl(hl,pi)
c--------------------------------------------------
c     specific heat of compressed liquid, J/m3-K.
c--------------------------------------------------
      data a00,a01,a10,a11,a20,a21/2.3949e-4,-5.1963e-13,
     1      1.1932e-11,2.4127e-18,-3.9441e-17,-1.6808e-24/
      a0=a00+pi*a01
      a1=a10+pi*a11
      a2=a20+pi*a21
      cpl=1./(a0+hl*(a1+hl*a2))
      return
      end
      function cpv(hv,pi)
c--------------------------------------------------
c     specific heat of supreheated vapor, J/m3-K.
c--------------------------------------------------
      data b00,b01,b02,b10,b11,b12/-5.2569e-4,-3.4406e-11,
     1      7.0081e-19,3.2441e-10,3.7348e-18,-2.9134e-26/
      b0=b00+pi*(b01+pi*b02)
      b1=b10+pi*(b11+pi*b12)
      cpv=1./(b0+hv*b1)
      return
      end
      function visl(hl,pi)
c---------------------------------------------
c     viscosity of compressed liquid, N-s/m2.
c---------------------------------------------
      data c0,c1,c2,c3,c4,d0,d1,d2,d3/1.2995e-3,-9.264e-4,
     1      3.8105e-4,-8.2194e-5,7.0224e-6,-6.5959e-12,6.763e-12,
     2      -2.8883e-12,4.4525e-13/
      data e00,e01,e02,e03,e10,e11,e12,e13/1.4526e-3,-6.9881e-9,
     1      1.521e-14,-1.2303e-20,-3.8064e-11,3.9285e-16,
     2      -1.2586e-21,1.286e-27/
      data f0,f1,f2,f3,f4,h1,h2/3.026e-4,-1.8366e-4,7.5671e-5,
     1      -1.6479e-5,1.4165e-6,2.76e5,3.94e5/
```

```
      data p1,g0,g1,g2,g3,g4,g5/6.8946e5,8.5813e-6,4.2659e4,
     1     6.4845e-6,5.5359e4,3.8921e-6,4.0147e5/
      if(hl.ge.h2) go to 10
      if(hl.gt.h1) go to 20
      x=g0*(hl-g1)
      e=g2*(hl-g3)
      visl=c0+x*(c1+x*(c2+x*(c3+x*c4)))-(pi-p1)*(d0-e*(d1+
     1     e*(d2+e*d3)))
      return
10    z=g4*(hl-g5)
      visl=f0+z*(f1+z*(f2+z*(f3+z*f4)))
      return
20    e0=e00+hl*(e01+hl*(e02+hl*e03))
      e1=e10+hl*(e11+hl*(e12+hl*e13))
      visl=e0+e1*(pi-p1)
      return
      end
      function visv(hv,pi,tvi,rovi)
c--------------------------------------------------
c     viscosity of supreheated vapor, N-s/m2.
c--------------------------------------------------
      data a0,a1,b0,b1,c0,c1,c2,c3/4.07e-8,8.04e-6,1.858e-7,
     1     5.9e-10,-2.885e-6,2.427e-8,-6.7893e-11,6.317e-14/
      data d0,d1,d2,d3,e0,e1,e2,t1,t2/1.76e+2,-1.6,4.8e-3,
     1     -4.7407e-6,3.53e-8,6.765e-11,1.021e-14,300.,375./
      tv1=tvi-273.15
      if(tv1.ge.t2) go to 10
      if(tv1.gt.t1) go to 20
      visv=a0+a1*tv1-rovi*(b0+b1*tv1)
      return
10    visv=a0+a1*tv1-rovi*(e0+rovi*(e1+rovi*e2))
      return
20    visv=a0+a1*tv1+rovi*(c0+tv1*(c1+tv1*(c2+tv1*c3)))+rovi*
     1     (d0+tv1*(d1+tv1*(d2+d3*tv1)))*(e0+rovi*(e1+rovi*e2))
      return
      end
      function cndl(hl,pi)
c--------------------------------------------------
c     conductivity of compressed liquid, W/m-K.
c--------------------------------------------------
      data a0,a1,a2,a3,h0/0.57374,0.25361,-0.14547,1.3875e-2,
     1     5.815e5/
      x=hl/h0
      cndl=a0+x*(a1+x*(a2+x*a3))
      return
      end
      function cndv(hv,pi,tvi,rovi)
c--------------------------------------------------
c     conductivity of superheated vapor, W/m-K.
c--------------------------------------------------
      data c,a0,a1,a2,a3,b0,b1,b2/2.1482e5,1.76e-2,5.87e-5,
     1     1.04e-7,-4.51e-11,1.0351e-4,4.198e-7,-2.771e-11/
      tv1=tvi-273.15
```

```
      x=a1+tv1*(a1+tv1*(a2+tv1*a3))
      z=b0+tv1*(b1+tv1*b2)
      cndv=x+rovi*(z+(c*rovi)/tv1**4.2)
      return
      end
      function surten(tl)
c--------------------------------------
c     surface tension of liquid water
c--------------------------------------
c     in units of 9.80665 Kg-m/s**2-m
c     also equal to surface tension/gravitational accelaeration
c     in units of Kg/m
c
c     error of approximation = 2 % for 373 < tl < 623 K
c     value at 250 C =0.0026
c
      surten=(80.72-tl*0.126)/(5140.+tl)
      if(surten.lt.0.) surten=0.
      return
      end
```

```
      subroutine time(dt,dtmax,vcr,rocr,rof,dmic,alp)
c
c     calc. transport time through the core
c
      save
      dt=vcr*rocr/dmic
      if(alp.lt.0.98) go to 10
      if(dt.gt.dtmax) dt=dtmax
10    ti=ti+dt
      return
      end
```

```
      subroutine power(qi,ti,tscram,ppow,dt,isw)
c---------------------------
c     calculate decay power
c---------------------------
c
c  Parameters:
c
c          beta(i)  = fraction yield for group i
c          decay(i) = decay constant for group i
c          cd(i)    = normalized decay power for group i
c          dcd(i)   = change in cd(i)
c          dpow     = normalized total decay power
c          pown     = normalized total nuclear power
c          fd       = fraction of decay power at steady state
c
      common /pfull/qi0,dmi0,pi0,thot0,tcold0,delt0
      common /force/psfac(30),tps(30),qfac(30),tq(30),np,nq
      dimension beta(7),decay(7),cd(7),dcd(7)
      save
      data beta(1),beta(2),beta(3),beta(4),beta(5),beta(6),beta(7)
     &      /0.097,0.22,0.237,0.187,0.132,0.072,0.055/
      data decay(1),decay(2),decay(3),decay(4),decay(5),decay(6),
     &     decay(7)/1.28,0.152,1.93e-2,1.88e-3,1.43e-4,1.25e-5,2.2e-7/
      data fd/0.0668/
      fact=1.0
      if(nq.gt.0) call table(fact,ti,qfac,tq,nq)
      qii=qi0*fact*ppow/100.
      if(isw.eq.1) go to 30
      go to 50
30    do 40 i=1,7
40    cd(i)=1.0
      qi=qii
      return
50    if(ti.lt.tscram) go to 16
      powr=0.
      if(iflag.gt.0) go to 17
      write(6,15) ti
15    format(//,' *** REACTOR IS SCRAMMED AT ',f8.2,' SEC ***')
      iflag=1
      go to 17
c   -------------------------
c     calculate decay power
c   -------------------------
16    powr=fact
17    dpow=0.
      do 60 i=1,7
      dcd(i)=dt*decay(i)*(powr-cd(i))/(1.+dt*decay(i))
      cd(i)=cd(i)+dcd(i)
      if(cd(i).lt.1.e-30) cd(i)=0.
      dpow=dpow+cd(i)*beta(i)
60    continue
      pown=(1.-fd)*powr+fd*dpow
      qi=pown*ppow*qi0/100.
```

```
      return
      end
      function pow(expi)
      if(expi.lt.35.) go to 10
      pow=0.
      return
10    pow=exp(-expi)
      return
      end
```

```
      subroutine sgpow(qisg,dmi,pi,hfp,hfg,hhl,hsg,hcl,thl,tsg,
     &     tcl,dthldp,dtsgdp,dthldh,dtsgdh,cpl,tsec,pisec,arsg,
     &     xhl,xsg,strl1,strl2,dqhl,dqsg,dqsgdp,boil,isw)
c--------------------------------------
c     calculate heat transfer rate in
c      a U-tube steam generator
c--------------------------------------
c     output variables:  qisg,strl1,strl2
c
c  boil   = boiling length
c  hi     = tube-side heat transfer coefficient
c  ho     = shell-side heat transfer coefficient
c  rff    = fauling factor
c  uo     = overall heat transfer coefficient
c           (based on shell-side area)
c  strl1  = single-phase heat transfer length
c  strl2  = two-phase heat transfer length
c
c     single-phase heat transfer correlation:  Dittus-Bolter
c     two-phase heat transfer correlation:     Thom
c
      common/steam/do,di,ntube,tbl,zl,voltm
      save
      data pei/3.1416/
      data eps/0.1/
      tout=tsg
      tin=thl
      hout=hsg
      hin=hhl
      if(dmi.lt.0.) tin=tcl
      if(dmi.lt.0.) hin=hcl
      ta=0.5*(tin+tout)
      tw=0.5*ta+tsec
      ha=0.5*(hout+hin)
      if(tin.eq.tout) go to 100
      if(tout.le.tsec) go to 5
      go to 7
100   dta=tin-tsec
      go to 10
c
c     calculate log-mean temperature
c
7     if(tin.lt.tsec.and.tout.gt.tsec) go to 8
      t1=tin-tout
      t2=tin-tsec
      t3=tout-tsec
      tln=alog(t2/t3)
      dta=t1/tln
      go to 6
c
c     calculate average heat transfer temperature
c
8     dta=ta-tsec
```

```
      dtin=0.5
      dtout=0.5
      go to 6
c
c     calculate modified log-mean temperature
c
5     if(tin.lt.tsec) go to 7
c      eps=0.01*(tin-tout)
      t1=tin-tout-eps
      if(t1.lt.0.) go to 8
      t2=tin-tsec
      t3=tin-tout
      t4=t2-eps
      tln=alog(t2/eps)
      dtlm=t1/tln
      eta=(dtlm-eps)/t4
      dta=eta*t2+(1.-eta)*(tout-tsec)
      dtin=(1.-1./tln/t2)/tln
      dtout=-1./tln
      dein=(dtin-(dtlm-eps)/t4)/t4
      deout=dtout/t4
      dtin=t3*dein+eta
      dtout=t3*deout+1.-eta
6     if(isw.eq.0) go to 10
c
c     determine initial fauling factor
c
      asg=pei*ntube*do*tbl
      call htco(ta,dmi,arsg,pi,ha,cpl,di,1,hi)
      z2=2.265e-2*sqrt(asg)*exp(-pisec/8.7e6)
      ho=sqrt(qisg)/z2
      rff=asg*dta/qisg-do/(hi*di)-0.5*do*alog(do/di)/cndin(tw)
     &    -1./ho
      uo=qisg/(asg*dta)
      strl1=abs(dmi)*cpl*tbl/(asg*uo)
      strl2=strl1
      dqhl=0.
      dqsg=0.
      dqsgdp=0.
      boil=0.
      return
c
c     single-phase condition for entire tube side
c
10    if(xhl.gt.0.) go to 20
      call htco(ta,dmi,arsg,pi,ha,cpl,di,1,hi)
      asg=pei*ntube*do*tbl
      z1=do/(hi*di)+0.5*do*alog(do/di)/cndin(tw)+rff
      z2=2.265e-2*sqrt(asg)*exp(-pisec/8.7e6)
      z3=asg*dta
      if(dta.lt.0.) go to 11
      qisgr=0.5*(-z2+sqrt(z2*z2+4.*z1*z3))/z1
      qisg=qisgr*qisgr
```

```
      go to 12
11    qisgr=0.5*(-z2-sqrt(abs(z2*z2+4.*z1*z3)))/z1
      qisg=-qisgr*qisgr
12    uo=1./(z1+z2/sqrt(abs(qisg)))
      strl1=abs(dmi)*cpl/(pei*do*ntube*uo)
      strl2=strl1
      if(tout.le.tsec) go to 15
      if(tin.le.tsec) go to 15
c
c     implicit form
c
      dtin=(tln-t1/t2)/(tln*tln)
      dtout=(t1/t3-tln)/(tln*tln)
15    dqhl=asg*uo*dtin*dthldh
      dqsg=asg*uo*dtout*dtsgdh
      dqsgdp=asg*uo*(dtout*dtsgdp+dtin*dthldp)
      boil=0.
      return
20    if(xsg.eq.0.) go to 30
c
c     two-phase condition for entire tube side
c
      asgo=ntube*pei*do*tbl
      ho=abs((tin-tsec))*exp(2.*pisec/8.7e6)/2.265e-2
      hi=abs((tin-tsec))*exp(2.*pi/8.7e6)/2.265e-2
      uo2=1./(do/di/hi+0.5*do*alog(do/di)/cndin(tw)+1./ho+rff)
      strl2=abs(dmi)*hfg/(pei*ntube*do*uo2*tin)
      strl1=strl2
      qisg=asgo*uo2*(tin-tsec)
      dqhl=0.
      dqsg=0.
      dqsgdp=0.
      boil=tbl
      return
c
c     S/G power from two-phase portion
c
30    ho=abs((tin-tsec))*exp(2.*pisec/8.7e6)/2.265e-2
      hi=abs((tin-tsec))*exp(2.*pi/8.7e6)/2.265e-2
      uo2=1./(do/di/hi+0.5*do*alog(do/di)/cndin(tw)+1./ho+rff)
      strl2=abs(dmi)*hfg/(pei*ntube*do*uo2*abs(tin-tsec))
      boil=abs(tbl*(hin-hfp)/(hin-hout))
      if(boil.gt.tbl) boil=tbl-0.1
      asgo=ntube*pei*do*boil
      qisg2=asgo*uo2*(tin-tsec)
c
c     S/G power from single-phase portion
c
      call htco(ta,dmi,arsg,pi,ha,cpl,di,1,hi)
      z1=do/(hi*di)+0.5*do*alog(do/di)/cndin(tw)+rff
      asgo=ntube*pei*do*(tbl-boil)
      z2=2.265e-2*sqrt(asgo)*exp(-pisec/8.7e6)
      z3=asgo*dta
```

```
      qisgr1=0.5*(-z2+sqrt(z2*z2+4.*z1*z3))/z1
      qisg1=qisgr1*qisgr1
      uo=1./(z1+z2/sqrt(qisg1))
      str11=abs(dmi)*cp1/(pei*do*ntube*uo)
      qisg=qisg1+qisg2
      dqh1=0.
      dqsg=0.
      dqsgdp=0.
      return
      end
      function cndin(tw)
c
c     conductivity of Inconel 600
c
c     the range of data base is within
c     473 K < tw , 673 K
c
      data cd0,cd1/9.632,0.016/
      cndin=cd0+cd1*tw
      return
      end
```

```
      subroutine sgbh(bsg,dmi,cpl,pi,rof,rog,rohl,rosg,xhl,xsg,xcl,
     &            strl1,strl2,arsg,dsg,sig,bl)
c-------------------------------------------------------
c     calculate bouyancy head in steam generator
c-------------------------------------------------------
c     output variable:  bsg
c
c         bsg  = steam generator bouyancy head
c         bl   = boiling length
c
      common/steam/do,di,ntube,tbl,zl,voltm
      if(dmi.lt.0.) go to 5
      xin=xhl
      xout=xsg
      go to 6
5     xin=xcl
      xout=xsg
6     if(xin.gt.0.) go to 10
c
c     single-phase condition
c
      c1=zl/strl1
      c2=(tbl-zl)/strl1
      c3=tbl/strl1
      bsg=(rosg-rohl)*strl1*(1.-pow(c1)-pow(c2)+pow(c3))/
     &    (1.-pow(c3))
      if(dmi.lt.0.)
     & bsg=(rosg-rohl)*strl1*(1.-pow(c1)-pow(c2)+pow(c3))/
     &    (1.-pow(c3))
      return
c
c     two-phase condition
c
10    call drift(ugj,c0,rof,rog,sig,pi,dsg,3)
      a=c0*(1.-rog/rof)
      b=c0*rog/rof+rog*ugj*arsg/abs(dmi)
      a1=(rof-rog)/a-rof
      a2=(rof-rog)*b/(a*a)
      if(bl.eq.tbl) go to 40
      if(bl.lt.zl) go to 20
      if(bl.lt.(tbl-zl)) go to 30
c
c     (tbl-zl) < bl < tbl
c
      bsg2i=strl2*(a1*xin*zl/bl+
     &      a2*alog((a*xin+b)/(a*xin*zl/bl+b)))
      bsg2j=strl2*(a1*xin*(1.-(tbl-zl)/bl)+
     &      a2*alog((a*xin*(1.-(tbl-zl)/bl)+b)/b))
      c1=(tbl-bl)/strl1
      a3=1.-pow(c1)
      bsg1=rof*(tbl-zl)+(rosg-rof)*(tbl-bl+strl1*a3)/a3
      bsg=bsg2i+bsg2j+bsg1
      return
```

```
c
c     0 < bl < zl
c
20    bsg2=strl2*(al*xin-a2*alog((a*xin+b)/b))
      cl=(zl-bl)/strl1
      bsgl=bl*(rosg-rof*pow(cl))/(1.-pow(cl))
      bsg=bsg2+bsgl
      return
c
c     zl < bl < (tbl-zl)
c
30    bsg2=strl2*(al*xin*zl/bl+
     &      a2*alog((a*xin+b)/(a*xin*zl/bl+b)))
      cl=(tbl-zl-bl)/strl1
      c2=(tbl-bl)/strl1
      bsgl=rof*zl+(rosg-rof)*(zl-strl1*(pow(cl)-pow(c2)))/
     &      (1.-pow(c2))
      bsg=bsg2+bsgl
      return
c
c     bl > tbl
c
40    bsg2i=strl2*(al*xin*zl/bl+
     &      a2*alog((a*xin+b)/(a*xin*zl/bl+b)))
      bsg2j=strl2*(al*(xin*(1.-(tbl-zl)/tbl)-xout)
     &      +a2*alog((a*xin*(1.-(tbl-zl)/tbl)+b)/(a*xout+b)))
      bsg=bsg2i+bsg2j
      return
      end
```

```
      subroutine hpsi(wsi,pi)
c
c     determines the HPSI mass flow rate
c
      common /safe/whpsi,psih,psil,hsi
      wsi=0.
      if(pi.gt.psih) return
      if(pi.lt.psil) return
      wsi=whpsi
      return
      end
```

```
      subroutine critic(wbr,dwbdp,pi,psec,x,rom,drldp,drfdp,drgdp,
     &             dhfdp,dhgdp,hfg,rof,rog,arbr,fbr)
c
c     calculates critical mass flow rate
c
      if(x.gt.0.) go to 100
c
c     single-phase liquid
c
      root=sqrt(2.*rom*(pi-psec))
      wbr=fbr*0.61*arbr*root
      dwbdp=fbr*0.61*arbr*((pi-psec)*drldp+rom)/root
      return
c
c     homogeneous equilibrium model (HEM)
c
100   vf=1./rof
      vg=1./rog
      vfg=vg-vf
      dvfdp=-drfdp/rof/rof
      dvgdp=-drgdp/rog/rog
      gcsq=-1./((1.-x)*dvfdp+x*dvgdp-vfg*((1.-x)*dhfdp+
     &     x*dhgdp-1./rom)/hfg)
      wbr=arbr*sqrt(gcsq)
      dwbdp=0.
      return
      end
```

```
      subroutine drift(ugj,c0,rof,rog,sig,pi,di,jcomp)
c--------------------------------------
c     determine drift flux parameters
c--------------------------------------
c     return variables:  ugj,c0
c
      data g,pa/9.80665,2.209e7/
      pr=pi/pa
      go to (100,200,300,200) jcomp
c
c     Zuber correlation for bubbly and slug flow
c
100   ugj=1.41*(sig*g*g*(rof-rog)/(rof*rof))**0.25
c  -- Zuber correlation for rectangular duct --
      c0=1.4-0.4*pr
      return
c
c     horizontal flow:  Zuber correlation for tubes
c
200   ugj=0.
      if(di.gt.0.05) go to 210
      if(pr.gt.0.5) go to 220
      c0=1.2
      return
220   c0=1.2-0.4*(pr-0.5)
      return
210   c0=1.5-0.5*pr
      return
c
c     inverted U-tube
c
300   c0=1.2-0.2*sqrt(rog/rof)
      ugj=1.41*(sig*g*g*(rof-rog)/(rof*rof))**0.25
      return
      end
```

```
      subroutine pump(dmi,dmic,wi,ti,dt,roip,rocr,roep,rohl,rosg,rocl,
     &   xip,xcr,xep,xhl,xsg,xcl,pi,
     &   dpb,dpl,hc,trip,vis)
c    ****************************************************
c    * solve the momentum and pump-rotor speed equations *
c    ****************************************************
c
c      the algorithm includes a procedure for
c      anti-reverse rotation of pump rotor
c
c      output variables:  dmi,dmic,wi,dpl
c
      common /input/t,fdt,tscram,ppow,ind,nloop,nh,nc,rn,pi1,
     &         tcold1,wlevel,wi1,dmi1,pisec1,n,m,isw
      common /pfull/qi0,dmi0,pi0,thot0,tcold0,delt0
      common /finert/za1,za2,za3,za4,za5,za6,za7,za8,za9,za10,
     &        zal,zalrv
      common /dimens/ar1,ar2,ar3,ar4,ar5,ar6,ar7,ar8,ar9,ar10,
     &        d1,d2,d3,d4,d5,d6,d7,d8,d9,d10,v1,v2,v3,v4,v5,v6,
     &        v7,v8,v9,v10,z1,z2,z3,z4,z5,z6,z7,z8,z9,z10,zs,zc0,
     &        zcl,zep
      common /loss/sk1e,sk1c,cv1,cv2,ep,sk2,sk4,sk5e,sk5c,sk6,
     &        sk8e,sk9e,sk9c,sk10,sk90,skv,nvh,nvc,ns,nbh,nbc
      common /pres2/hfp,hgp,hfg,rof,rog,hlp,hvp,alp,rolp,rovp,
     &         drvdp,drvdh,drldp,drldh,vlp,vvp,
     &         wmass,smass,wlevel,tlp,tvp,twi,tw
      common /pumps/tr,tinert,ter,taue,phcr,ror,twbr,dmirr,wirr,fh
      common /head/hc0,hc1,hc2,hc3
      common /torq/tb0,tb1,tb2,tb3
      dimension dmi(4),wi(4),rohl(4),rosg(4),rocl(4),xhl(4),xsg(4),
     &           xcl(4),dpb(4),dpl(4),trip(4),iflag(4),wint(4)
      dimension b11(4),b12(4),b13(4),
     &           b21(4),b22(4),h1(4),h2(4),ddmi(4),dwi(4),hc(4)
c  pump head capacity coefficients for Wi < 0
      data hc0n,hc1n,hc2n,hc3n/0.5,0.51,-0.26,0.25/
c  pump BHP torque coefficients for wi < 0
      data tb0n,tb1n,tb2n,tb3n/-0.65,1.9,-1.28,0.54/
      data wint(1),wint(2),wint(3),wint(4)/0.1,0.1,0.1,0.1/
      save
      dwr=1./wirr
      dmr=1./dmirr
      rn2=rn*rn
      do 1000 i=1,m
      j=0
10    dmir=dmi(i)/dmirr
      ar=rocl(i)/ror
      admi=abs(dmi(i))
      admic=abs(dmic)
      dmi2=dmi(i)*admi
      dmic2=dmic*admic
      if(wi(i).ne.0.0)go to 30
      wr=0.
      go to 110
```

```
30    wr=wi(i)/wirr
      wr2=wr*wr
      a=dmir/wr
      x=a/ar
      dxdw=-dmir/(wr2*wirr*ar)
      dxdm=dmr/wr/ar
      if(wint(i).gt.0.1) go to 110
      if(dmi(i).lt.0.) wint(i)=wr
c-----------------------------------------
c     determine windage and bearing torque
c-----------------------------------------
110   if(wi(i).ge.23.51)go to 20
      if(wi(i).eq.0.) go to 35
      twb=0.035*twbr
      dtwb=0.
      go to 40
20    twb=twbr*wr**2
      dtwb=2.*twbr*wr/wirr
      go to 40
35    twb=0.5*twbr
      dtwb=0.
40    continue
c--------------------------
c     determine BHP torque
c--------------------------
      if(abs(wr).le.wint(i)) go to 200
c
c     BHP Torque for wr > 0.1 (Maine Yankee)
c
      tbr=tb0+x*(tb1+x*(tb2+tb3*x))
      dtbrdx=tb1+x*(2.*tb2+3.*tb3*x)
      tbhp=tr*wr2*tbr
      dtbdw=tr*(wr2*dxdw*dtbrdx+2.*wr*tbr/wirr)
      dtbdm=tr*wr2*dxdm*dtbrdx
      go to 81
c
c     linear interpolation between -0.1 >= wr >= 0.1
c
c-------------------------------------------
c     BHP Torque for wr = 0.1 (Main Yankee)
c
200   db=(wint(i)-abs(wr))/(0.1+wint(i))
      dbdw=-dwr/(0.1+wint(i))
      wri=wint(i)
      wr2i=wri*wri
      xi=dmir/wri/ar
      dxdmi=dmr/wri/ar
      tbrp=wr2i*(tb0+xi*(tb1+xi*(tb2+tb3*xi)))
      dtbrp=wr2i*dxdmi*(tb1+xi*(2.*tb2+3.*tb3*xi))
c
c     BHP Torque for wr = -0.1 (Rust)
c
      wrn=-0.1
```

```
      xn=dmir/wrn/ar
      dxdmn=dmr/wrn/ar
      tbrn=wr2i*(tb0n+xn*(tb1n+xn*(tb2n+tb3n*xn)))
      dtbrn=wr2i*dxdmn*(tb1n+xn*(2.*tb2n+3.*tb3n*xn))
c
c     linear interpolation for wi(i) >= 0
c
      tbhp=tr*((1.-db)*tbrp+db*tbrn)
      dtbdm=tr*((1.-db)*dtbrp+db*dtbrn)
      dtbdw=tr*(tbrn-tbrp)*dbdw
c      dtbdw=0.
81    continue
c-----------------------------------
c     determine pump head capacity
c-----------------------------------
      if(wi(i).le.0.) go to 120
      if(wr.le.wint(i)) go to 300
c
c     pump head in first quadrant, W>0 h>0 (Maine Yankee)
c
      hr=hc0+x*(hc1+x*(hc2+hc3*x))
      dhrdx=hc1+x*(2.*hc2+3.*hc3*x)
      hc(i)=phcr*wr2*hr
      dhcdw=phcr*(2.*wr*hr/wirr+wr2*dhrdx*dxdw)
      dhcdm=phcr*wr2*dhrdx*dxdm
      go to 130
c
c     linear interpolation for 0.1 >= wr >= 0
c
c---------------------------------------------------
c
c     pump head for wr = 0.1 (Maine Yankee)
c
300   db=(wint(i)-abs(wr))/wint(i)
      dbdw=-dwr/wint(i)
      wri=wint(i)
      wr2i=wri*wri
      xi=dmir/wri/ar
      dxdmi=dmr/wri/ar
      hrp=wr2i*(hc0+xi*(hc1+xi*(hc2+xi*hc3)))
      dhrp=wr2i*dxdmi*(hc1+xi*(2.*hc2+3.*hc3*xi))
c
c     pump head for wi(i) = 0
c
      hrn=-fh*2.01e-2*dmi2/(ar7*ar7*phcr)
      dhrn=-fh*2.0*2.01e-2*dmi(i)/(ar7*ar7*phcr)
      hc(i)=phcr*((1.-db)*hrp+db*hrn)
      dhcdm=phcr*((1.-db)*dhrp+db*dhrn)
c      dhcdw=phcr*(hrn-hrp)*dbdw
      dhcdw=0.
      go to 130
c
c     pump head for wi(i) = 0
```

```
c
120   hc(i)=-fh*2.01e-2*dmi2/(ar7*ar7)
      dhcdw=0.
      dhcdm=-fh*2.0*2.01e-2*dmi(i)/(ar7*ar7)
130   continue
      q1=tbhp*wi(i)
      q2=dmi(i)*hc(i)/rocl(i)
      if(q1.lt.q2) go to 310
      go to 140
310   tbhp=dmi(i)*hc(i)/(wi(i)*rocl(i))
      dtbdm=hc(i)/(wi(i)*rocl(i))
      dtbdw=-dmi(i)*hc(i)/(wi(i)*wi(i)*rocl(i))
140   continue
c--------------------------
c     check pump trip time
c--------------------------
      ta=ti-trip(i)
      if(ti.lt.trip(i)) go to 85
      if(iflag(i).gt.0) go to 85
      write(6,84) i,ti
84    format(1h1,//,' *** PUMP ',i1,' IS TRIPPED AT ',f8.2,' SEC ***')
      iflag(i)=1
85    if(ta.lt.0.) ta=0.
c-------------------------------
c     calculate electric torque
c-------------------------------
      tie=2.*ta/taue
      if(tie.gt.50.) tie=50.
      tele=ter*wr*exp(-tie)
c-------------------------------------
c     pump torque functional relation
c-------------------------------------
      gwi=(tele-tbhp-twb)/tinert/pf(ti)
c      if(wi(i).le.0.) go to 90
      dgdw=(-dtbdw-dtwb)/tinert/pf(ti)
      dgdm=-dtbdm/tinert/pf(ti)
c      go to 92
c90    dgdw=0.
c      dgdm=0.
c92    continue
c-------------------------------
c     calculate pressure losses
c-------------------------------
c
c  -- calc. two-phase multiplier for form loss --
c
100   tpip=tph(xip,rof,rog)
      tpcr=tph(xcr,rof,rog)
      tpep=tph(xep,rof,rog)
      tphl=tph(xhl(i),rof,rog)
      tpsg=tph(xsg(i),rof,rog)
      tpcl=tph(xcl(i),rof,rog)
c
```

```
c  -- calc. core pressure loss --
c
c  core friction loss
      call ffact(ff1,dff1,admic,vis,ar1,d1,2.5e-7,xcr,rof,rog,pi)
      dp1f=0.5*ff1*dmic2*za1/(ar1*d1*rocr)
      ddp1f=dp1f*(2./admic+dff1/ff1)
c  core entrance & exit losses
      dp1i=0.5*(sk1e+sk1c)*tpcr*dmic2/(ar1*ar1*rocr)
      ddp1i=2.*dp1i/admic
c  spacer form loss
      cv=cv1*tpcr*admic**cv2
      dcv=cv2*cv/admic
      dp1s=0.5*ns*cv*ep*ep*dmic2/(ar1*ar1*rocr)
      ddp1s=dp1s*(2./admic+dcv/cv)
      dp1=dp1f+dp1i+dp1s
      ddp1=ddp1f+ddp1i+ddp1s
c
c  -- calc. R/V exit (nozzle) pressure loss --
c
      dp2=0.5*sk2*tpep*rn2*dmi2/(ar3*ar3*roep)
      ddp2=2.*dp2/(rn*admi)
c
c  -- calc. pressure loss in hot leg --
c
      call ffact(ff3,dff3,rn*admi,vis,ar3,d3,2.5e-7,xhl(i),rof,rog,pi)
      dp3f=0.5*ff3*rn2*dmi2*za3/(ar3*d3*rohl(i))
      ddp3f=dp3f*(2./(rn*admi)+dff3/ff3)
c  form loss due to 90 deg bend
      dp3s=0.5*sk90*tphl*nbh*rn2*dmi2/(ar3*ar3*rohl(i))
      ddp3s=2.*dp3s/(rn*admi)
      dp3=dp3f+dp3s
      ddp3=ddp3f+ddp3s
c
c  -- calc. S/G inlet plenum pressure loss --
c
      dp4=0.5*sk4*tphl*rn2*dmi2/(ar3*ar3*rosg(i))
      ddp4=2.*dp4/(rn*admi)
c
c  -- calc. S/G tube pressure loss --
c
c  tube friction loss
      call ffact(ff5,dff5,rn*admi,vis,ar5,d5,2.5e-7,xsg(i),rof,rog,pi)
      dp5f=0.5*ff5*rn2*dmi2*za5/(ar5*d5*rosg(i))
      ddp5f=dp5f*(2./(rn*admi)+dff5/ff5)
c  S/G tube entrance & exit & bend losses
      dp5s=0.5*(sk5c+sk5e+2.*sk90)*tpsg*rn2*dmi2/(ar5*ar5*rosg(i))
      ddp5s=2.*dp5s/(rn*admi)
      dp5=dp5f+dp5s
      ddp5=ddp5f+ddp5s
c
c  -- calc. S/G exit plenum pressure loss --
c
      dp6=0.5*sk6*tpcl*dmi2/(ar7*ar7*rosg(i))
```

```
      ddp6=2.*dp6/admi
c
c  -- calc. pressure loss in cold leg --
c
c  friction loss
      call ffact(ff7,dff7,admi,vis,ar7,d7,2.5e-7,xcl(i),rof,rog,pi)
      dp7f=0.5*ff7*dmi2*za7/(ar7*d7*rocl(i))
      ddp7f=dp7f*(2./admi+dff7/ff7)
c  form loss due to 90 deg bend
      dp7s=0.5*sk90*tpcl*nbc*dmi2/(ar7*ar7*rocl(i))
      ddp7s=2.*dp7s/admi
      dp7=dp7f+dp7s
      ddp7=ddp7f+ddp7s
c
c  -- calc. R/V entrance & 90 deg turn losses --
c
      dp8=0.5*sk8e*tpcl*dmi2/(ar7*ar7*rocl(i))
      ddp8=2.*dp8/admi
c
c  -- calc. R/V thermal shield pressure loss --
c
c  friction loss
      call ffact(ff9,dff9,admic,vis,ar9,d9,2.5e-7,xip,rof,rog,pi)
      dp9f=0.5*ff9*dmic2*za9/(ar9*d9*roip)
      ddp9f=dp9f*(2./admic+dff9/ff9)
c  down-comer entrance & exit losses
      dp9s=0.5*(sk9c+sk9e)*tpip*dmic2/(ar9*ar9*roip)
      ddp9s=2.*dp9s/admic
      dp9=dp9f+dp9s
      ddp9=ddp9f+ddp9s
c
c  -- calc. R/V lower plenum pressure loss --
c
      dp10=sk10*tpip*dmic2
      ddp10=2.*dp10/admic
c
c  -- calc. pressure losses due to stop valves --
c
      dpvhl=0.5*skv*tphl*dmi2*nvh/(ar3*ar3*rohl(i))
      ddpvh=2.*dpvhl/admi
      dpvcl=0.5*skv*tpcl*dmi2*nvc/(ar7*ar7*rocl(i))
      ddpvc=2.*dpvcl/admi
      dpv=dpvhl+dpvcl
      ddpv=ddpvh+ddpvc
c---------------------------------------
c     overall pressure loss in the RCS
c---------------------------------------
      dpl(i)=dp3+dp4+dp5+dp6+dp7+dpv
      ddpl=ddp2+ddp3+ddp4+ddp5+ddp6+ddp7+ddp8+ddpv
      dprv=dpl+dp2+dp8+dp9+dp10
      ddprv=ddpl+ddp9+ddp10
c-----------------------------------------------------
c     determine differential function of momentum eq.
```

```
c-----------------------------------------------------------
      fdmi=-dpl(i)-dprv+hc(i)+dpb(i)
      dfdmi=-ddpl+dhcdm
      if(dmi(i).le.0.) dfdmi=0.
      dfdmic=-ddprv
      dfdw=dhcdw
      if(dmi(i).le.0.) dfdw=0.
      if(isw.gt.0) go to 135
c------------------
c     setup matrix
c------------------
      b11(i)=zal-dt*dfdmi
      b12(i)=-dt*dfdw
      b13(i)=zalrv-dt*dfdmic
      b21(i)=-dt*dgdm
      b22(i)=1.-dt*dgdw
      h1(i)=dt*fdmi
      h2(i)=dt*gwi
1000  continue
      if(isw.gt.0) go to 700
      if(m.gt.1) go to 500
c
c     single-loop algorithm
c
      b31=-nloop
      b22(1)=1./b22(1)
      b21(1)=b21(1)*b22(1)
      h2(1)=h2(1)*b22(1)
      b11(1)=1./(b11(1)-b31*b13(1)-b21(1)*b12(1))
      h1(1)=b11(1)*(h1(1)-b12(1)*h2(1))
      ddmi(1)=h1(1)
      dwi(1)=h2(1)-b21(1)*ddmi(1)
      dmi(1)=dmi(1)+ddmi(1)
      wi(1)=wi(1)+dwi(1)
      if(wi(1).lt.0.) wi(1)=0.
      dmic=-b31*dmi(1)
      return
c
c     multiple-loop algorithm
c
500   b31=-1.
      b33=1.
      h3=0.
      do 530 i=1,m
      b11(i)=1./b11(i)
      b12(i)=b12(i)*b11(i)
      b13(i)=b13(i)*b11(i)
      h1(i)=h1(i)*b11(i)
      b22(i)=1./(b22(i)-b21(i)*b12(i))
      h2(i)=b22(i)*(h2(i)-b21(i)*h1(i))
      b33=b33-b31*b13(i)
      h3=h3-b31*h1(i)+b31*b12(i)*h2(i)
530   continue
```

```
      b33=1./b33
      h3=h3*b33
      ddmic=h3
      do 550 i=1,m
      dwi(i)=h2(i)
      ddmi(i)=h1(i)-b12(i)*dwi(i)-b13(i)*ddmic
      wi(i)=wi(i)+dwi(i)
      dmi(i)=dmi(i)+ddmi(i)
      if(wi(i).lt.0.) wi(i)=0.
550   continue
      dmic=0.
      do 560 i=1,m
560   dmic=dmic+dmi(i)
      return
700   dmic=nloop*dmi(1)
      write(6,800) dp1f,dp1i,dp1s,dp2,dp3f,dp3s,dp4,dp5f,dp5s,dp6,dp7f,
     &             dp7s,dp8,dp9f,dp9s,dp10,dpvhl,dpvcl,dpb(1),dpl(1),
     &             dprv
800   format(1x,//,' Initial-State Pressure Losses (Pa)',/,
     &      ' Core friction loss:               ',f8.0,/,
     &      ' Core entrance & exit losses:      ',f8.0,/,
     &      ' Spacer form loss:                 ',f8.0,/,
     &      ' R/V exit form loss:               ',f8.0,/,
     &      ' Hot leg friction loss:            ',f8.0,/,
     &      ' Hot leg form loss:                ',f8.0,/,
     &      ' S/G inlet plenum form loss:       ',f8.0,/,
     &      ' S/G tube friction loss:           ',f8.0,/,
     &      ' S/G tube form losses:             ',f8.0,/,
     &      ' S/G exit plenum form loss:        ',f8.0,/,
     &      ' Cold leg friction loss:           ',f8.0,/,
     &      ' Cold leg form losses:             ',f8.0,/,
     &      ' R/V entrance form losses:         ',f8.0,/,
     &      ' R/V down-comer friction loss:     ',f8.0,/,
     &      ' R/V down-comer form losses:       ',f8.0,/,
     &      ' R/V lower plenum form loss:       ',f8.0,/,
     &      ' Hot-leg stop valve loss:          ',f8.0,/,
     &      ' Cold-leg stop valve loss:         ',f8.0,//,
     &      ' Bouyancy pressure rise:           ',f8.0,/,
     &      ' Total losses in the loop:         ',f8.0,/,
     &      ' Total losses in the R/V:          ',f8.0)
      return
c-----------------------------------------------------
c     perform Newton-Raphson iteration
c     to find initial pump speed and mass flow rate
c-----------------------------------------------------
135   fx=dfdmi
c      fy=dfdw
c      gx=dgdm
c      gy=dgdw
c      f=-fdmi
c      g=-gwi
c      fx=1./fx
c      fy=fy*fx
```

```
c        f=f*fx
c        gy=1./(gy-fy*gx)
c        g=g*gy
c        dy=g
c        dx=f-fy*dy
      dx=-fdmi/dfdmi
      dmi(i)=dmi(i)+dx
c        wi(i)=wi(i)+dy
      j=j+1
c        if(abs(dx).lt.(1.e-6*dmirr).and.abs(dy).lt.(1.e-6*wirr))
c     &  go to 1000
      if(abs(dx).lt.(1.e-6*dmirr)) go to 1000
      if(j.le.20) go to 10
      write(6,101)
101   format(/,1x,'WARNING!!!  Initial Pump and Momentum Eq.',1x,
     &        ' Iterations',/,1x,'did Not Converge to the Limits.')
      go to 1000
      end
```

```
      subroutine ffact(ff,dff,dmi,vis,ar,dh,ek,x,rof,rog,pi)
c-------------------------------
c     calculate friction factor
c-------------------------------
c
c     return variables:  ff,dff
c
c         ff   = friction factor
c         dff  = partial derivative of ff wrt dmi
c         ek   = roughness value (mm)
c         re   = Reynold number
c         phi2 = M-N-J two-phase multiplier
c
      data el/-1.9/
      data pconv,gconv/1.45e-4,737.3/
c
c     Colebrook-White Eq.
c
      dre=dh/(ar*vis)
      re=dmi*dre
c      if(re.le.2300.) pause
      alpha=0.2703*ek/dh+5.74/re**0.9
      beta=alog10(alpha)
      ff=0.25/(beta*beta)
      dal=-5.166*re**el*dre
      dff=-0.2172*dal/(beta*beta*beta*alpha)
      if(x.eq.0.) return
c
c     Martinelli-Nelson-Jones Correlation
c
      gbrit=gconv*dmi/ar
      pbrit=pconv*pi
      gr=1.e-6*gbrit
      if(gr.gt.0.7) go to 10
      r=1.36+5.e-4*pbrit+0.1*gr-7.14e-4*pbrit*gr
      go to 20
10    r=1.26-4.e-4*pbrit+0.119/gr+2.8e-4*pbrit/gr
20    phi2=r*1.2*(rof/rog-1.)*x**0.824+1.
      ff=phi2*ff
      dff=phi2*dff
      return
      end
      function tph(x,rof,rog)
c
c     Homogeneous two-phase multiplier
c
      tph=1.0
      if(x.eq.0.) return
      tph=1.+x*(rof/rog-1.)
      return
      end
```

```
      subroutine output(ti,pi,qi,qisg,qith,wi,dmi,dmic,tfa,tca,tip,
     &     tcr,tep,thl,tsg,tcl,tsec,alpip,alpcr,alpep,alphl,alpsg,
     &     alpcl,zc,deltr,tsat,tlp,tvp,wlevel,qh,alp,wlr,wfl,wro,wwc,
     &     wsv,wrv,wsp,wcharg,wsu,wsum,dpl,dpb,hc,jsw,jflag,iflag,
     &     wbr,wsi,m,n,isw)
c     *****************************
c     * print calculation results *
c     *****************************
      dimension qisg(4),wi(4),dmi(4),thl(4),tsg(4),tcl(4),alphl(4),
     &     alpsg(4),alpcl(4),zc(4),thlc(4),tsgc(4),tclc(4),qisgp(4),
     &     dpl(4),hc(4),dpb(4),wbr(4),wsi(4)
      data tabs/273.16/
      tfac=tfa-tabs
      tipc=tip-tabs
      tcrc=tcr-tabs
      tepc=tep-tabs
      tcac=tca-tabs
      tsecc=tsec-tabs
      tlpc=tlp-tabs
      tvpc=tvp-tabs
      qhp=qh/1.e3
      qithp=qith/1.e6
      qip=qi/1.e6
      pip=pi/1.e6
      wbr0=0.
      wsi0=0.
      do 10 i=1,m
      wbr0=wbr0+wbr(i)
      wsi0=wsi0+wsi(i)
      thlc(i)=thl(i)-tabs
      tsgc(i)=tsg(i)-tabs
      tclc(i)=tcl(i)-tabs
10    qisgp(i)=qisg(i)/1.e6
      write(6,20) ti,qithp,tvpc,qip,tlpc,pip,wlevel,tfac,alp,
     &            tsecc,qhp,wcharg,wsp,wbr0,wrv,wsi0,wsv
20    format(//,1x,'Time, sec:                              ',f10.2,
     &       10x,' Pressurizer',/,
     &       1x,'Thermal Power, MW:                ',f10.4,
     &       10x,' Vapor Temp, C:        ',f10.2,/,
     &       1x,'Nuclear Power, MW:                ',f10.4,
     &       10x,' Liquid Temp, C:       ',f10.2,/,
     &       1x,'Pressure, MPa:                    ',f10.4,
     &       10x,' Water Level, m:       ',f10.2,/,
     &       1x,'Avg Fuel Temp, C:                 ',f10.4,
     &       10x,' Void Fraction:        ',f10.4,/,
     &       1x,'Secondary Temp, C:                ',f10.4,
     &       10x,' Heater Power, KW:     ',f10.4,/,
     &       1x,'Charging Mass Flow Rate, Kg/s:  ',f10.4,
     &       10x,' Spray Mass Flow Rate:  ',f10.4,/,
     &       1x,'Break Mass Flow Rate:             ',f10.4,
     &       10x,' R/V Mass Flow Rate:    ',f10.4,/,
     &       1x,'Safety Injection Mass Flow Rate:',f10.4,
     &       10x,' S/V Mass Flow Rate:    ',f10.4)
```

```
      write(6,30)
30    format(/,1x,'Core Mass Flow',2x,'I-P Temp',3x,'Core Temp',4x,
     &       'E-P Temp',4x,'I-P Void',3x,'Core Void',4x,'E-P Void')
      write(6,40) dmic,tipc,tcrc,tepc,alpip,alpcr,alpep
40    format(7(2x,f10.4))
      write(6,50)
50    format(/,1x,'Loop',2x,'S/G Power',2x,'Mass Flow',2x,
     &       'Pump Speed',2x,'H-L Temp',
     &       3x,'S/G Temp',3x,'C-L Temp',3x,'H-L Void',3x,'S/G Void',
     &       3x,'C-L Void',3x,'Pump Hd  ',3x,'Bouy Hd')
      do 70 i=1,m
      write(6,60) i,qisgp(i),dmi(i),wi(i),thlc(i),tsgc(i),tclc(i),
     &            alphl(i),alpsg(i),alpcl(i),hc(i),dpb(i)
      write(9,111) dmi(i),wi(i),dpb(i),thlc(i),tclc(i),tsgc(i),qisg(i),
     &             dpl(i),hc(i)
60    format(1x,1x,i2,9(1x,f10.4),2x,e10.4,1x,f10.4)
70    continue
      write(9,111)dmic,ti,pi,qi,qith,tfac,tcac,tcrc,tipc,
     &           deltr,wsu,wlevel
111   format(v)
      return
      end
```

```
      subroutine plot(m,n,t,nloop)
c------------------------
c     setup plot arrays
c------------------------
      common a(1)
      common/point1/ldmic,ldmi1,ldmi2,ldmi3,ldmi4,lwi1,lwi2,lwi3,lwi4,
     &       lti,lqi,lqith,lqisg1,lqisg2,lqisg3,lqisg4,lpi,
     &       ltfa,ltca,ltcr,ltip,lthl1,lthl2,lthl3,lthl4,ltcl1,ltcl2,
     &       ltcl3,ltcl4,ltsg1,ltsg2,ltsg3,ltsg4,
     &       lzc1,lzc2,lzc3,lzc4,ldeltr,lwsu,lwlr,ldpl1,ldpl2,ldpl3,
     &       ldpl4,lhc1,lhc2,lhc3,lhc4
c-------------------------
c     set array pointers
c-------------------------
      a(1)=a(1)
      ldmic=n
      ldmi1=ldmic+n
      ldmi2=ldmi1+n
      ldmi3=ldmi2+n
      ldmi4=ldmi3+n
      lwi1=ldmi4+n
      lwi2=lwi1+n
      lwi3=lwi2+n
      lwi4=lwi3+n
      lti=lwi4+n
      lqi=lti+n
      lqith=lqi+n
      lqisg1=lqith+n
      lqisg2=lqisg1+n
      lqisg3=lqisg2+n
      lqisg4=lqisg3+n
      lpi=lqisg4+n
      ltfa=lpi+n
      ltca=ltfa+n
      ltcr=ltca+n
      ltip=ltcr+n
      lthl1=ltip+n
      lthl2=lthl1+n
      lthl3=lthl2+n
      lthl4=lthl3+n
      ltcl1=lthl4+n
      ltcl2=ltcl1+n
      ltcl3=ltcl2+n
      ltcl4=ltcl3+n
      ltsg1=ltcl4+n
      ltsg2=ltsg1+n
      ltsg3=ltsg2+n
      ltsg4=ltsg3+n
      lzc1=ltsg4+n
      lzc2=lzc1+n
      lzc3=lzc2+n
      lzc4=lzc3+n
      ldeltr=lzc4+n
```

```
      lwsu=ldeltr+n
      lwlr=lwsu+n
      ldpl1=lwlr+n
      ldpl2=ldpl1+n
      ldpl3=ldpl2+n
      ldpl4=ldpl3+n
      lhc1=ldpl4+n
      lhc2=lhc1+n
      lhc3=lhc2+n
      lhc4=lhc3+n
c--------------------------
c     plot state variables
c--------------------------
      call ploter(a(ldmic),a(ldmi1),a(ldmi2),a(ldmi3),a(ldmi4),
     &      a(lwi1),a(lwi2),a(lwi3),a(lwi4),
     &      a(lti),a(lqi),a(lqith),a(lqisg1),a(lqisg2),a(lqisg3),
     &      a(lqisg4),a(lpi),a(ltfa),a(ltca),
     &      a(ltcr),a(ltip),a(lthl1),a(lthl2),a(lthl3),a(lthl4),
     &      a(ltcl1),a(ltcl2),a(ltcl3),a(ltcl4),
     &      a(ltsg1),a(ltsg2),a(ltsg4),a(ltsg4),a(lzc1),a(lzc2),
     &      a(lzc3),a(lzc4),a(ldeltr),a(lwsu),a(lwlr),a(ldpl1),
     &      a(ldpl2),a(ldpl3),a(ldpl4),a(lhc1),a(lhc2),a(lhc3),
     &      a(lhc4),m,n,t,nloop)
      return
      end
```

```
      subroutine ploter(dmic,dmi1,dmi2,dmi3,dmi4,wi1,wi2,wi3,wi4,
     &          ti,qi,qith,qisg1,qisg2,qisg3,qisg4,pi,tfa,
     &          tca,tcr,tip,thl1,thl2,thl3,thl4,tcl1,tcl2,tcl3,tcl4,
     &          tsg1,tsg2,tsg3,tsg4,zc1,zc2,zc3,zc4,
     &          deltr,wsurge,wlr,dpl1,dpl2,dpl3,dpl4,hc1,hc2,hc3,hc4,
     &          m,n,t,nloop)
c-----------------------------------
c     plot state variable histories
c-----------------------------------
      common /pfull/qi0,dmi0,pi0,tcr0,tcl0,deltr0
      character symbol
      dimension dmic(n),dmi1(n),dmi2(n),dmi3(n),dmi4(n),
     &          ti(n),wi1(n),wi2(n),wi3(n),wi4(n),
     &          qi(n),qith(n),qisg1(n),qisg2(n),qisg3(n),qisg4(n),
     &          pi(n),tca(n),tfa(n),tcr(n),tip(n),thl1(n),thl2(n),
     &          thl3(n),thl4(n),tcl1(n),tcl2(n),tcl3(n),tcl4(n),
     &          tsg1(n),tsg2(n),tsg3(n),tsg4(n),
     &          zc1(n),zc2(n),zc3(n),zc4(n),deltr(n),wsurge(n),wlr(n),
     &          dpl1(n),dpl2(n),dpl3(n),dpl4(n),hc1(n),hc2(n),hc3(n),
     &          hc4(n)
      data pmp,pkp/1.e-6,1.e-3/
      external plot_(descriptors),plot_$setup(descriptors),
     &          plot_$scale(descriptors)
      symbol='0'
11    format(v)
      rewind(9)
      do 50 i=1,n
      do 20 j=1,m
      go to (110,120,130,140),j
110   read(9,11) dmi1(i),wi1(i),zc1(i),thl1(i),tcl1(i),tsg1(i),
     &          qisg1(i),dpl1(i),hc1(i)
      go to 20
120   read(9,11) dmi2(i),wi2(i),zc2(i),thl2(i),tcl2(i),tsg2(i),
     &          qisg2(i),dpl2(i),hc2(i)
      go to 20
130   read(9,11) dmi3(i),wi3(i),zc3(i),thl3(i),tcl3(i),tsg3(i),
     &          qisg3(i),dpl3(i),hc3(i)
      go to 20
140   if(j.gt.m) go to 20
      read(9,11) dmi4(i),wi4(i),zc4(i),thl4(i),tcl4(i),tsg4(i),
     &          qisg4(i),dpl4(i),hc4(i)
20    continue
      read(9,11) dmic(i),ti(i),pi(i),qi(i),qith(i),tfa(i),tca(i),
     &          tcr(i),tip(i),deltr(i),wsurge(i),wlr(i)
50    continue
      dmii=dmi1(1)
      dmic1=dmic(1)
      wii=wi1(1)
      do 60 i=1,n
      dmic(i)=100.*(dmic(i)/dmic1)
      pi(i)=pi(i)*pmp
      do 60 j=1,m
      go to (61,62,63,64),j
```

```
61    dmi1(i)=100.*(dmi1(i)/dmii)
      wi1(i)=100.*(wi1(i)/wii)
      dpl1(i)=dpl1(i)*pkp
      hc1(i)=hc1(i)*pkp
      go to 60
62    dmi2(i)=100.*(dmi2(i)/dmii)
      wi2(i)=100.*(wi2(i)/wii)
      dpl2(i)=dpl2(i)*pkp
      hc2(i)=hc2(i)*pkp
      go to 60
63    dmi3(i)=100.*(dmi3(i)/dmii)
      wi3(i)=100.*(wi3(i)/wii)
      dpl3(i)=dpl3(i)*pkp
      hc3(i)=hc3(i)*pkp
      go to 60
64    dmi4(i)=100.*(dmi4(i)/dmii)
      wi4(i)=100.*(wi4(i)/wii)
      dpl4(i)=dpl4(i)*pkp
      hc4(i)=hc4(i)*pkp
60    continue
      do 100 i=1,n
      qi(i)=100.*(qi(i)/qi0)
      qith(i)=100.*(qith(i)/qi0)
      qisg1(i)=100.*(qisg1(i)/qi0)
      qisg2(i)=100.*(qisg2(i)/qi0)
      qisg3(i)=100.*(qisg3(i)/qi0)
      qisg4(i)=100.*(qisg4(i)/qi0)
      deltr(i)=100.*deltr(i)
100   continue
      xmax=ti(n)
      dmimax=amax1(dmi1(1),dmi1(n),dmi2(n),dmi3(n))
      dmimin=amin1(dmi1(n),dmi2(n),dmi3(n))
      phmax=hc1(1)
      phmin=amin1(hc1(n),hc2(n),hc3(n),hc4(n))
      ppmin=amin1(dpl1(n),dpl2(n),dpl3(n),dpl4(n))
      call plot_$setup("MASS FLOW RATE HISTORY","TIME (s)",
     &     "MASS FLOW RATE %",1,1.,1,0)
      call plot_$scale(0.,xmax,dmimin,dmimax)
      call plot_(ti,dmic,n,1,0)
      do 70 j=1,m
      go to (71,72,73,74),j
71    call plot_(ti,dmi1,n,1,0)
      go to 70
72    call plot_(ti,dmi2,n,1,0)
      go to 70
73    call plot_(ti,dmi3,n,1,0)
      go to 70
74    call plot_(ti,dmi4,n,1,0)
70    continue
      write(0,12)
12    format(" ")
      read(0,11) j
      call plot_$setup("HYDR HEAD HISTORY","TIME (s)",
```

```
     &     "HEAD Pa",1,1.,0,0)
      call plot_$scale(0.,xmax,0.,2500.)
      do 80 i=1,m
      go to (81,82,83,84),i
81    call plot_(ti,zc1,n,1,0)
      go to 80
82    call plot_(ti,zc2,n,1,0)
      go to 80
83    call plot_(ti,zc3,n,1,0)
      go to 80
84    call plot_(ti,zc4,n,1,0)
80    continue
      write(0,12)
      read(0,11) j
      call plot_$setup("PUMP SPEED HISTORIES","TIME (s)",
     &     "PUMP SPEED %",1,1.,0,0)
      call plot_$scale(0.,xmax,0.,100.)
      do 160 i=1,m
      go to (161,162,163,164), i
161   call plot_(ti,wi1,n,1,0)
      go to 160
162   call plot_(ti,wi2,n,1,0)
      go to 160
163   call plot_(ti,wi3,n,1,0)
      go to 160
164   call plot_(ti,wi4,n,1,0)
160   continue
      write(0,12)
      read(0,11) j
      call plot_$setup("POWER HISTORIES","TIME (s)",
     &     "POWER %",3,10.,2,0)
      call plot_$scale(0.,xmax,0.1,100.)
      call plot_(ti,qi,n,1,0)
      call plot_(ti,qith,n,1,0)
      write(0,12)
      read(0,11) j
      call plot_$setup("R/V TEMP RISE % Power","TIME (s)",
     &     "% TEMP RISE",1,1.,0,0)
      call plot_$scale(0.,xmax,0.,100.)
      call plot_(ti,deltr,n,1,0)
      write(0,12)
      read(0,11) j
      call plot_$setup("CORE INLET & EXIT TEMP HISTORIES","TIME (s)",
     &     "TEMP C",1,1.,0,0)
      call plot_$scale(0.,xmax,250.,300.)
      call plot_(ti,tcr,n,1,0)
      call plot_(ti,tip,n,1,0)
      do 90 i=1,m
      write(0,12)
      read(0,11) j
      go to (91,92,93,94), i
91    call plot_$setup('TEMP DISTRIBUTION IN LOOP 1',
     &     'TIME (s) ', 'TEMP C',1,1.,0,0)
```

```
      call plot_$scale(0.,xmax,250.,300.)
      call plot_(ti,thl1,n,1,0)
      call plot_(ti,tcl1,n,1,0)
      call plot_(ti,tsg1,n,1,0)
      go to 90
92    call plot_$setup('TEMP DISTRIBUTION IN LOOP 2',
     &      'TIME (s) ','TEMP C',1,1.,0,0)
      call plot_$scale(0.,xmax,260.,290.)
      call plot_(ti,thl2,n,1,0)
      call plot_(ti,tcl2,n,1,0)
      call plot_(ti,tsg2,n,1,0)
      go to 90
93    call plot_$setup('TEMP DISTRIBUTION IN LOOP 3',
     &      'TIME (s) ','TEMP C',1,1.,0,0)
      call plot_$scale(0.,xmax,260.,290.)
      call plot_(ti,thl3,n,1,0)
      call plot_(ti,tcl3,n,1,0)
      call plot_(ti,tsg3,n,1,0)
      go to 90
94    call plot_$setup('TEMP DISTRIBUTION IN LOOP 4',
     &      'TIME (s) ','TEMP C',1,1.,0,0)
      call plot_$scale(0.,xmax,260.,290.)
      call plot_(ti,thl4,n,1,0)
      call plot_(ti,tcl4,n,1,0)
      call plot_(ti,tsg4,n,1,0)
90    continue
      write(0,12)
      read(0,11) j
      call plot_$setup("AVG FUEL TEMP HISTORY","TIME (s)",
     &      "TEMP C",1,1.,0,0)
      call plot_$scale(0.,xmax,200.,1000.)
      call plot_(ti,tfa,n,1,0)
      write(0,12)
      read(0,11) j
      call plot_$setup("PRESSURE HISTORY","TIME (s)",
     &      "PRESSURE MPa",1,1.,0,0)
      call plot_$scale(0.,xmax,13.,15.)
      call plot_(ti,pi,n,1,0)
      write(0,12)
      read(0,11) j
      write(0,12)
      read(0,11) j
      call plot_$setup("SURGE FLOW RATE HISTORY","TIME (s)",
     &      "MASS FLOW RATE Kg/s",1,1.,1,0)
      call plot_$scale(0.,xmax,-200.,200.)
      call plot_(ti,wsurge,n,1,0)
      write(0,12)
      read(0,11) j
      call plot_$setup("PRESSURIZER WATER LEVEL","TIME (s)",
     &      "WATER LEVEL %",1,1.,0,0)
      call plot_$scale(0.,xmax,0.,100.)
      call plot_(ti,wlr,n,1,0)
      write(0,12)
```

```
      read(0,11) j
      call plot_$setup('STEAM GENERATOR POWERS','TIME (s)',
     &     'POWER MW',1,1.,0,0)
      sgmin=qisg1(n)
      sgmax=qisg1(1)
      call plot_$scale(0.,xmax,sgmin,sgmax)
      do 150 i=1,m
      go to (151,152,153,154), i
151   call plot_(ti,qisg1,n,1,0)
      go to 150
152   call plot_(ti,qisg2,n,1,0)
      go to 150
153   call plot_(ti,qisg3,n,1,0)
      go to 150
154   call plot_(ti,qisg4,n,1,0)
150   continue
      write(0,12)
      read(0,11) j
      return
      end
```

REFERENCE

[A1] J. P. Adams, et al., "Natural Circulation Cooling Characteristics During PWR Accident Simulations," EGG-M-08782.

[A2] A. K. Agrawal, et al., "An Advanced Thermohydraulic Simulation Code for Transients in LMFBRs (SSC-L Code)," BNL-NUREG-50773 (1978).

[B1] G. Baltra Aedo, "Design of a Coordinated Plant Control System for a Marine Nuclear Propulsion Plant", N. E. Thesis, Dept. of Nucl. Eng., MIT, 1981.

[B3] B. Baggoura and W. R. Martin, "Transient Analysis of the Three Mile Island Unit 2 Pressurizer System," Nuclear Technology, Vol. 62, Aug. 1983.

[B4] R. C. Baron, "Digital Model Simulation of a Nuclear Pressurizer",Nucl. Sci. and Eng.: 52, 283-291(1973).

[B5] D. L. Batt and J. M. Carpenter, "Experiment Data Report for LOFT Anticipated Transient Experiments L6-1, L6-2, and L6-3," EGG-2067, Dec. 1980.

[B6] M. A. Bjerke, et al., "A Review of Short-Term Fission-Product-Decay Power," Nuclear Safety, Vol. 18, No. 5, 1977.

[C1] A. Chenini, "Set-Theoretic Control of A Pressurized Water Nuclear Power Plant," Eng. Thesis, Dept. of Nucl. Eng., MIT, Sept. 1980.

[C2] R. N. Clark and B. Campbell, "Instrument Fault Detection in A Pressurized Water Reactor Pressurizer," Nuclear Technology, Vol. 56, Jan. 1982, 23-32.

[C3] J. G. Collier, Convective Boiling and Condensation, 2nd ed., MeGraw-Hill, 1981.

[C4] "Maine Yankee Plant Reactor Coolant System," Combustion Engineering, Inc., System Description No. 4467-010, Rev. 0 (1971).

[C5] "Natural Circulation Test Program, San Onofre Nuclear Generating Station Unit 2 Natural Circulation Cooldown," Combustion Engineering, Inc., CEN-201(S) Supp. No. 1, Jan. 1983.

[C6] K. F. Cooper, "Stochastic Optimal Control of a Pressurized Water Reactor System," Ph. D. Thesis, University of Pittsburg, 1980.

[D1] O. L. Deutsch, et al., "Development and Testing of A Real-Time Measurement Validation Program for Sodium Flowrate in EBR-II," CSDL-R-1592, Oct. 1982.

[D2] E. Z. Drucker and D. J. Gorman, "A Method of Predicting Steam-Wurge Tank Transients Based on One-Dimensional Heat Sinks," Nucl. Sci. and Eng, 21, 1965, pp 473-480.

[E1] "Summary and Evaluation of Scoping and Feasibility Studies for Disturvance Analysis and Surveillance System (DASS)," EPRI-NP-1684, December 1980.

[F1] "Maine Yankee Atomic Power Station Final Safety Analysis Report," Docket 30309, 1971.

[G1] C. Geffray, "Nuclear Reactor (PWR) Pressurizer Real Time Modeling for Sensor Validation", S. M. Thesis, Dept. of Nucl. Eng., MIT, 1980.

[G2] D. L. Gillas and J. M. Carpenter, "Experiment Data Report for LOFT Nuclear Small Break Experiment L3-7," EGG-2049, Aug. 1980.

[G3] J. G. Guppy, et al., "Super System Code (SSC-L, Rev. 2), An Advanced Thermohydraulic Simulation Code for Transients in LMFBRs," BNL-NUREG-51650.

[I1] M. Ishii, "One-Dimensional Drift-Flux Model and Constitutive Equations for Relative Motion Between Phases in Various Two-Phase Flow Regimes," ANL-77-47 (1977).

[H1] D. L. Hetrick, et al., "Solution Methods for Simulation of Nuclear Power Systems," EPRI-NP-1928, July 1981.

[H2] F. B. Hildebrand, Introduction to Numerical Analysis, 2nd ed., McGraw-Hill, 1974.

[H3] J. H. Hopps, Jr., "Safety Parameter Display Systems: A High Information Reliability Design Concept," ASME Second International Computer Engineering Conference, San Diego, August 15-19, 1982.

[K1] J. E. Kelly, et al., "User's Guide for THERMIT-2: A Version of THERMIT for both Core-Wide and subchannel Analysis of Light Water Reactors," MIT Energy Lab. Report No. MIT-EL-81-029, Aug, 1981.

[K2] T. W. Kerlin, et al., "Theoretical and Experimental Dynamic Analysis of the H. B. Robinson Nuclear Plant," Nuclear Technology, Vol. 30, Sept. 1976.

[K3] S. N. Kim, "An Experimental Model of A PWR Pressurizer During Transients," Ph. D. Thesis, Dept. of Nucl. Eng., MIT, 1984.

[L1] R. T. Lahey and F. J. Moody, The Thermal-Hydraulics of A Boiling Water Nuclear Reactor, ANS, 1977.

[L2] M. T. Leonard, "The Effect of A Non-Condensable Gas on Pressurizer Insurge Transients," S. M. Thesis, Dept. of Nucl. Eng., MIT 1983.

[M1] G. C. Mashe, Systems Summary of a Westinghouse Pressurized Water Reactor Nuclear Power Plant, Westinghouse Electric Co., 1971.

[M2] C. H. Meijer, et al.,"On-Line Power Plant signal Validation Technique Utilizing Parity-Space Representation and Analytic Redundancy," EPRI-NP-2110, November 1981.

[M3] J. E. Meyer and E. A. Reinhard, "Numerical Techniques for Boiling Flow Stability Analyses," Trans. ASME Series C, 87, 311-312 (1965).

[M4] J. E. Meyer, "Some Physical and Numerical Considerations for the SSC-S Code," BNL-NUREG-50913 (1978).

[M5] J. E. Meyer, "Hydrodynamic Models for the Treatment of Reactor Thermal Transients,' Nucl. Sci. and Eng.: 19, 269-277 (1961).

[M6] J. E. Meyer, "Conservation Laws in One-Dimensional Hydrodynamics," WAPD-BT-20, Sept. 1960.

[M7] E. O. Moeck and H. W. Hinds, "A Mathematical Model of Steam-Drum Dynamics", AECL-5057, 1976.

[M8] K. V. Moore, et al., "RETRAN A Program for One-Dimensional Transient Thermal-Hydraulic Analysis of Complex Fluid Flow Systems," EPRI-NP-408, Jan. 1977.

[M9] R. L. Moore, Jr., "Adaptive Estimation and Control for Nuclear Power Plant Load Changes," Ph. D. Thesis, Dept. of E. E., MIT, 1971.

[M10] D. S. Miller, Internal Flow Systems, BHRA Fluid Engineering, 1978.

[N1] A. N. Nahavandi, "An Improved Pressurizer Model with Bubble Rise and Condensate Drop Dynamics", Nucl. Eng. and Design 12(1970) 135-147.

[N2] G. Y. Nakayama, "Linearized Model Real Time Reactor Simulation," S. M. Thesis, Dept. of N. E., MIT, 1981.

[N3] C. L. Nalezny, "Summary of Nuclear Regulatory Commission's LOFT Program Experiments," EGG-2248, July 1983.

[N4] H. Nagumo, et al., "Experimental and Analytical Studies on Applicability of Drift Flux Model to Various Two Phase Flow Conditions," in "Small Break LOCA LWR Analysis Conference Papers," EPRI-WS-81-201.

[P1] T. A. Porsching, "A Finite Difference Method for Thermally Expandable Fluid Transients," Nucl. Sci. and Eng.: 64, 177-186 (1977).

[P2] M. A. Pulick and S. G. Margolis, "CRIB-1 -- A Steam Generator Stability Analysis Program for the PHILCO 2000 Computer," WAPD-TM-530.

[R1] V. H. Ransom, et al., "RELAP5/MOD 1 Code Manual Volume 1: system Models and Numerical Methods," NUREG/CR-1826, Nov. 1980.

[R2] A. Ray, et al., "On-Line Signal Validation and Feedback Control in Nuclear Reactor," Fifth Power Plant Dynamics, Control and Testing Symposiom, Knoxville, TN, March 1983.

[R3] D. L. Reeder, "LOFT System and Test Description," NUREG/CR-0247, July 1978.

[R4] P. J. Roache, Computational Fluid Dynamic, Hermosa Publishers, 1982.

[R5] J. T. Robinson, "Analysis of Loss of Feed Water ATWS Transients," S. M. Thesis, Dept. of Nucl. Eng., MIT, 1982.

[R6] J. T. Robinson, "Benchmark of RETRAN Pressurizer Model," Northeast Utilities, NE-81-R-488, 1981.

[R7] J. H. Rust, Nuclear Power Plant Engineering, Haralson Publishing Co., Buchanon, GA, 1979.

[R8] W. M. Rohsenow and H. Choi, Heat, Mass and Momentum Transfer, Prentice-Hall, Inc., 1961.

[S1] W. H. Strohmayer, "Dynamic Modeling of Vertical U-Tube Steam Generators for Operational Safety Systems," Ph. D. Thesis, Dept. of Nucl. Eng., MIT, August 1982.

[S2] "San Onofre Nuclear Generating Station Units 2 & 3 Preliminary Safety Analysis Report," Southern California Edison Co. and San Diego Gas & Elec. Co.

[T1] J. L. Tyler, "Low Order Model of the Loss-of-Fluid Test (LOFT) Reactor Plant for Use in Kalman Filter-Based Optimal Estimators," EGG-2006 (1980).

[U1] "Functional Criteria for Energency Response Facilities," U. S. N. R. C., NUREG-0696, 1981.

[W1] W. L. Weaver, III, J. E. Meyer, and A. K. Agrawal, "A Few-Pressure Model for Transient Two-Phase Flows in Networks," Trans. ANS, 28, 273-274 (1978).

[W2] S. Whitaker, Fundamental Principles of Heat Transfer, Pergamon Press Inc., 1977.

[W3] K. Wong, "Computer Model of A Nuclear Reactor Primary Coolant Pump," S. M. Thesis, Dept. N. E., MIT, 1982.

[W4] "Waterford Steam Electric Station Unit No. 3 Final Safety Analysis Report," Louisiana Power & Light Co.

[Y1] "Maine Yankee Start-Up Test Report" Yankee Atomic Co., 1973.

[Z1] N. Zuber and J. A. Findlay, "Average Volumetric Concentration in Two Phase Flow Systems," J. Heat Transfer, 87, 453-468 (1965).