SAND2023-10994 O

MELCOR Computer Code Manuals

Vol. 2: Reference Manual
Version 2.2 r2023.0

Date Published: January 2023

Update prepared by
L.L. Humphries, B.A. Beeny, F. Gelbard, T. Haskin, D. Louie, J. Phillips, J. Reynolds, R.C. Schmidt

Sandia National Laboratories
Albuquerque, NM 87185-0748

Operated for the U.S. Department of Energy

S. Campbell, Nuclear Regulatory Commission Project Manager

Prepared for Division of Systems Analysis
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555-0001

Abstract

MELCOR is a fully integrated, engineering-level computer code that models the progression of severe accidents in light water reactor nuclear power plants. MELCOR is being developed at Sandia National Laboratories[1] for the U.S. Nuclear Regulatory Commission as a second-generation plant risk assessment tool and the successor to the Source Term Code Package. A broad spectrum of severe accident phenomena in both boiling and pressurized water reactors is treated in MELCOR in a unified framework. These include thermal-hydraulic response in the reactor coolant system, reactor cavity, containment, and confinement buildings; core heatup, degradation, and relocation; core-concrete attack; hydrogen production, transport, and combustion; fission product release and transport behavior. Current uses of MELCOR include estimation of severe accident source terms and their sensitivities and uncertainties in a variety of applications.

This publication of the MELCOR computer code manuals corresponds to MELCOR 2.2. Volume 1 contains a primer that describes MELCOR's phenomenological scope, organization (by package), and documentation. The remainder of Volume 1 contains the MELCOR User's Guides, which provide the input instructions and guidelines for each package. Volume 2 contains the MELCOR Reference Manuals, which describe the phenomenological models that have been implemented in each package. Volume 3 of this publication presents a portfolio of test and sample problems consisting of both analyses of experiments and of full plant problems.

---

1 *Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.*

Contents

Volume 2: Reference Manual

The following chapters are found in Volume 2 of this code manual set.

Executive Summary

MELCOR is a fully integrated, engineering-level computer code whose primary purpose is to model the progression of accidents in light water reactor nuclear power plants. A broad spectrum of severe accident phenomena in both boiling and pressurized water reactors is treated in MELCOR in a unified framework. Current uses of MELCOR include estimation of fission product source terms and their sensitivities and uncertainties in a variety of applications.

The MELCOR code is composed of an executive driver and several major modules, or packages, that together model the major systems of a reactor plant and their generally coupled interactions. Reactor plant systems and their response to off-normal or accident conditions include:

- thermal-hydraulic response of the primary reactor coolant system, the reactor cavity, the containment, and the confinement buildings,
- core uncovering (loss of coolant), fuel heat-up, cladding oxidation, fuel degradation (loss of rod geometry), and core material melting and relocation,
- heat-up of reactor vessel lower head from relocated fuel materials and the thermal and mechanical loading and failure of the vessel lower head, and transfer of core materials to the reactor vessel cavity,
- core-concrete attack and ensuing aerosol generation,
- in-vessel and ex-vessel hydrogen production, transport, and combustion,
- fission product release (aerosol and vapor), transport, and deposition,
- behavior of radioactive aerosols in the reactor containment building, including scrubbing in water pools, and aerosol mechanics in the containment atmosphere such as particle agglomeration and gravitational settling, and,
- impact of engineered safety features on thermal-hydraulic and radionuclide behavior.

The various code packages have been written using a carefully designed modular structure with well-defined interfaces between them. This allows the exchange of complete and consistent information among them so that all phenomena are explicitly coupled at every step. The structure also facilitates maintenance and upgrading of the code.

Initially, the MELCOR code was envisioned as being predominantly parametric with respect to modeling complicated physical processes (in the interest of quick code execution time and a general lack of understanding of reactor accident physics). However, over the years as phenomenological uncertainties have been reduced and user expectations and

demands from MELCOR have increased, the models implemented into MELCOR have become increasingly best estimate in nature. The increased speed (and decreased cost) of modern computers (including PCs) has eased many of the perceived constraints on MELCOR code development. Today, most MELCOR models are mechanistic, with capabilities approaching those of the most detailed codes of a few years ago. The use of models that are strictly parametric is limited, in general, to areas of high phenomenological uncertainty where there is no consensus concerning an acceptable mechanistic approach.

Current uses of MELCOR often include uncertainty analyses and sensitivity studies. To facilitate these uses, many of the mechanistic models have been coded with optional adjustable parameters. This does not affect the mechanistic nature of the modeling, but it does allow the analyst to easily address questions of how particular modeling parameters affect the course of a calculated transient. Parameters of this type, as well as other numerical parameters such as convergence criteria and iteration limits, are coded in MELCOR as sensitivity coefficients, which may be modified through optional code input.

MELCOR modeling is general and flexible, making use of a "control volume" approach in describing the plant system. No specific nodalization of a system is forced on the user, which allows a choice of the degree of detail appropriate to the task at hand. Reactor-specific geometry is imposed only in modeling the reactor core. Even here, one basic model suffices for representing either a boiling water reactor (BWR) or a pressurized water reactor (PWR) core, and a wide range of levels of modeling detail is possible. For example, MELCOR has been successfully used to model East European reactor designs such as the Russian VVER and RMBK-reactor classes.

The MELCOR 2.2 code manuals are contained in three volumes. Volume 1 contains a primer that describes MELCOR's phenomenological scope, organization (by package), and documentation. The remainder of Volume 1 contains the MELCOR User's Guides, which provide the input instructions and guidelines for each package. Volume 2 contains the MELCOR Reference Manuals, which describe the phenomenological models that have been implemented in each package. Volume 3 contains a portfolio of sample demonstration problems. These problems are a combination of experiment analyses, which illustrate code model performance against data, and full plant analyses showing MELCOR's performance on larger realistic problems.

ACCUMULATOR (ACC) Package

The MELCOR ESF package models the physics for the various engineered safety features (ESFs) in a nuclear power plant. The Accumulator (ACC) package constitutes a subpackage within the ESF package and provides a simplified model to calculate liquid injection from a user specified accumulator. This reference manual gives a description of the physical models and numerical solutions implemented in the ACC package.

User input for running MELGEN and MELCOR with the ACC package activated is described separately in the Accumulator section of the Users' Guide.

Contents

1. Introduction

The MELCOR ESF package models the thermal-hydraulic behavior of various engineered safety features (ESFs) in nuclear power plants. The accumulator is a passive cooling system designed to provide water to the reactor coolant system in the event of a sudden drop in primary pressure, such as from a primary break. These systems are pressurized with a nitrogen cover gas and contain borated water. When the pressure in the primary system is low enough, the nitrogen forces the water out of the tank and into the coolant system. This can be modeled through control volumes, flow paths, and control functions, but was recently added as a system object to alleviate possible numerical challenges and to improve code performance during times of system injection.

2. Model Description

The accumulator is an engineered safety feature for injecting coolant into the RCS in a depressurization event such as a large pipe break. The tanks contain borated water with a nitrogen cover gas that provides the force for injecting the contents into the reactor core, once the reactor pressure falls below a set point. The accumulator and the reactor system are connected by a surge line which connects to the cold leg volume of the RCS which is specified on user input. The systems are separated by two check valves in series that open at the low pressure.

This simple model for the accumulator initiates a source of water mass and enthalpy to the cold leg (CV specified by input) when the pressure in the cold leg (P_{CVH}) drops below the accumulator pressure (P_{acc}), $\Delta P = P_{acc} - P_{CVH} > 0$. When such a condition is met, the velocity of the flow through the surge line is calculated:

$$u = \sqrt{\frac{2\Delta P}{k_{eff}\rho}}, \qquad k_{eff} = k + \frac{4fL}{D} \tag{2-1}$$

Where

u = the velocity of pool in the surge line (m/s),

L = the surge line length (m),

D = the surge line diameter (m),

f = friction coefficient,

$\rho = \text{fluid density } (\text{kg/m}^3)$,

k = form loss coefficient, and

k_{eff} = effective loss coefficient.

The friction coefficient is based on the Colebrook-White equation for turbulent flow:

$$\frac{1}{\sqrt{f}} = 3.48 - 4.\cdot log_{10}\left(2.\cdot\frac{e}{D} + \frac{9.35}{Re\sqrt{f}}\right) \quad (2\text{-}2)$$

where e is the the surface roughness (m).

Equation (2-2) is solved for f using the Newton method.

If a lag control component is specified by the user, the “lag” velocity is defined as:

$$Y^{n+1} = \frac{Y^n(1 - \frac{\Delta t}{2A_2}) + A_1(u^n + u^{n+1})\frac{\Delta t}{2A_2}}{1 + \frac{\Delta t}{2A_2}} \quad (2\text{-}3)$$

The volume of water ejected is calculated from this velocity, as well as the mass and enthalpy of the water which is then sourced into the control volume:

$$\Delta V = A_S \cdot u \cdot \Delta t \quad (2\text{-}4)$$

$$\Delta m = \rho \cdot \Delta V \quad (2\text{-}5)$$

$$\Delta H = H_{specific} \cdot \Delta m \quad (2\text{-}6)$$

Where the area, A_S, is calculated from the area of the surge line, which is calculated from the diameter provided by the user.

The calculated mass and enthalpy are added to the control volume connected to the accumulator just as any other mass and energy source would be.

Finally, the volume of the water in the accumulator is reduced and the pressure in the accumulator is recalculated based on either an adiabatic approximation or an isothermal approximation:

Adiabatic approximation for diatomic gas:

$$P_{N_2} \cdot {V_{N_2}}^{7/5} = const = P_{ACC,0} \cdot {V_{N_2,0}}^{7/5} \quad (2\text{-}7)$$

$$P_{N_2} = P_{ACC,0} \cdot \left(\frac{V_{N_2}}{V_{N_2,0}}\right)^{-7/5} = P_{ACC,0} \cdot \left(\frac{V_{N_2,0} + \Delta V}{V_{N_2,0}}\right)^{-7/5} = P_{ACC,0} \cdot \left(\frac{\Delta V}{V_{N_2,0}} + 1\right)^{-7/5} \quad (2\text{-}8)$$

$$P_{acc} = P_{N_2} + P_{H_2O}$$

$$P_{H_2O} << P_{N_2} \Rightarrow P_{ACC} = P_{N_2} = P_{ACC,0} \cdot \left(\frac{\Delta V}{V_{N_2,0}} + 1\right)^{-7/5} \quad (2\text{-}9)$$

Isothermal approximation:

$$P_{N_2} \cdot V_{N_2} = const = P_{ACC,0} \cdot V_{N_2,0} \quad (2\text{-}10)$$

$$P_{N_2} = P_{ACC,0} \cdot \left(\frac{V_{N_2}}{V_{N_2,0}}\right) = P_{ACC,0} \cdot \left(\frac{V_{N_2,0} + \Delta V}{V_{N_2,0}}\right) \quad (2\text{-}11)$$

$$P_{acc} = P_{N_2} + P_{H_2O}$$

$$P_{H_2O} << P_{N_2} \Rightarrow P_{ACC} = P_{N_2} = P_{ACC,0} \cdot \left(\frac{\Delta V}{V_{N_2,0}} + 1\right) \quad (2\text{-}12)$$

This package needs references and a much clearer writeup that includes definitions of all variables.

Burn (BUR) Package

The Burn (BUR) package models the combustion of gases in control volumes. The models consider the effects of burning on a global basis without modeling the actual reaction kinetics or tracking the actual flame front propagation. The BUR package models are based on the deflagration models in the HECTR 1.5 code. The diffusion flame model, also derived from HECTR 1.5, was added to the BUR package in the MELCOR 1.8.5 release.

This Reference Manual describes the models employed in the BUR package. Detailed descriptions of the user input requirements can be found in the BUR Package Users' Guide.

Contents

1. Introduction

The Burn (BUR) package models the combustion of gases in control volumes. These models consider the effects of burning off premixed gases without modeling the actual reaction kinetics or tracking the actual flame front propagation. The models in the BUR package are based on the deflagration models in the HECTR 1.5 code [1]. The only significant modifications made were to provide more direct user control of the models through the implementation of sensitivity coefficients and to include optional model parameters that are used to override the nominal parameters in control volumes in which direct containment heating (DCH) is occurring.

A diffusion flame model became available in MELCOR 1.8.5, also based on HECTR 1.5. The diffusion flame model allows more realistic modeling of DCH phenomena without having to make major adjustments to the nominal bulk burn parameters.

Briefly, a burn is initiated if certain criteria are satisfied in a control volume, causing the reactants (hydrogen, carbon monoxide and oxygen) to be converted during the burn to steam and carbon dioxide. The conversion occurs over a time interval called the burn duration. The reaction may or may not be complete, depending on the conditions in the control volume. After a burn is initiated in a control volume, it can be propagated to adjoining control volumes if a second set of criteria is satisfied. These criteria, as well as the duration and completeness of the burns, are discussed in Section 2. The modeling follows the recommendations of the MELCOR Assessment on Combustible Gas Treatment [2]. The default values and correlations used to calculate burn effects are those used in Reference 1.

For user convenience, the BUR package also prints messages to warn the user when the detonability criteria are satisfied in a control volume. A *detonation* is combustion in which the flame front travels at supersonic speeds, whereas a *deflagration* travels at subsonic speeds. In the BUR package, only deflagrations are modeled; detonations are merely flagged and no other action is taken.

The gases hydrogen (H_2), carbon monoxide (CO), carbon dioxide (CO_2), and oxygen (O_2) must be defined in the NonCondensible Gas (NCG) package whenever the BUR package is active. Steam (H_2O) is automatically present for all MELCOR calculations, so no special action need be taken to include it in a calculation.

The BUR package currently has a limited capability to burn deuterium gas (D_2). For purposes of combustion, D_2 is treated as equivalent to H_2 *on a mole-for-mole basis*. Therefore, one mole of D_2 combines with one-half mole of O_2 to produce one mole of H_2O (<u>not</u> D_2O), and mass is not conserved. Some equivalence must be assumed in the absence of a D_2O equation of state comparable in quality to the equation of state used in MELCOR for H_2O. Equivalence on a molar basis was chosen because the equations of state of D_2O and H_2O are much more similar on a molar basis than on a mass basis,

particularly in the gas phase. In addition, the former gives a more accurate value for the heat of combustion.

The same mole-for-mole equivalence is assumed in ignition, detonation, and completeness calculations, and input (or default) data for H_2 are applied to D_2 and H_2/ D_2 mixtures. We consider that the error is small: for example, the ideal combustion limits for D_2 are 5.0 to 95.0 mole percent compared to 4.0 to 94.0 mole percent for H_2.

2. Detailed Models

In the following equations, variables that are defined by user input are referred to by the same names as described in the Burn Package Users' Guide. Thus, there is a direct correspondence between the variables in the Users' Guide and those in the Reference Manual.

2.1 Burn Model Logistics

A burn is initiated in a control volume if the ignition criteria discussed in Section 2.2 are satisfied. As soon as a burn is initiated, calculations (described in Sections 2.3 and 2.4) are performed to determine the completeness of the burn and its duration. During subsequent timesteps, the reactants are converted to the products of combustion in that control volume according to the reactions.

$$H_2 + \frac{1}{2}O_2 \rightarrow H_2O \qquad (2\text{-}1)$$

and

$$CO + \tfrac{1}{2}O_2 \rightarrow CO_2. \qquad (2\text{-}2)$$

The rate of burning varies during the burn duration to account for change in composition (e.g., due to inter-compartment flow and gas sources), as described in Section 2.5.

After a burn is initiated in a control volume, it can be propagated to adjoining control volumes if a second set of criteria is satisfied. These criteria are discussed in Section 2.6. After a burn propagates into a control volume, the same steps as outlined above for ignition are followed to calculate the burn effects.

2.2 Ignition Criteria

A deflagration is initiated in a control volume if the mole fraction composition satisfies the criteria described in this section. In addition, control volumes that are specified to contain igniters are tested against different criteria than control volumes without igniters, and a separate criteria may be specified for use when direct containment heating (DCH) is

occurring in a control volume. For all cases, LeChatelier's formula (for the effective combustion mole fraction for a mixture containing more than one combustible gas) is used to determine the threshold of ignition. Ignition occurs when the following criterion is satisfied:

$$X_{H2} + X_{CO}\left({}^{L_{H2,ign}}\!/_{L_{CO,ign}}\right) \geq L_{H2,ign} \qquad (2\text{-}3)$$

where

X_{H2} = hydrogen mole fraction in the control volume;

X_{CO} = carbon monoxide mole fraction in the volume;

$L_{H2,ign}$ = XH2IGN, if there are no igniters in the volume and DCH is not occurring,

or

XH2IGY, if there are igniters in the volume and DCH is not occurring,

or

XH2DCH, if DCH is occurring in the volume;

$L_{CO,ign}$ = XCOIGN, if there are no igniters in the volume and DCH is not occurring,

or

XCOIGY, if there are igniters in the volume and DCH is not occurring,

or

XCODCH, if DCH is occurring in the volume;

XH2IGN = hydrogen mole fraction limit for ignition without igniters, when DCH is not occurring, input on record BUR_IGN (default = 0.10);

XH2IGY = hydrogen mole fraction limit for ignition with igniters, when DCH is not occurring, input on record BUR_IGN (default = 0.07);

XH2DCH= hydrogen mole fraction limit for ignition during DCH, input on record BUR_IGN (default = XH2IGY);

XCOIGN = carbon monoxide mole fraction limit for ignition without igniters, when DCH is not occurring, input on record BUR_IGN (default = 0.167);

XCOIGY = carbon monoxide mole fraction limit for ignition with igniters, when DCH is not occurring, input on record BUR_IGN (default = 0.129);

XCODCH = carbon monoxide mole fraction limit for ignition during DCH, input on record BUR_IGN (default = XCOIGY).

The preceding tests are made only for the presence of sufficient combustible gases. Tests are also made to determine whether there is sufficient oxygen and to determine whether the amount of steam and carbon dioxide is below the inerting level. The same values are used when igniters are present as when there are no igniters, but separate values may be specified for use during DCH. The ignition and inerting criteria are

$$X_{O2} \geq \text{XO2IG (or XO2DCH during DCH)} \tag{2-4}$$

and

$$X_{H2O} + X_{CO2} < \text{XMSCIG (or XINDCH during DCH)} \tag{2-5}$$

where

X_{O2} = oxygen mole fraction in the control volume;

X_{H2O} = steam mole fraction in the control volume;

X_{CO2} = carbon dioxide mole fraction in the volume;

XO2IG = minimum oxygen mole fraction for ignition, input on record BUR_IGN (default = 0.05);

XO2DCH = minimum oxygen mole fraction for ignition during DCH, input on record BUR_IGN (default = XO2IG);

XMSCIG = maximum diluent mole fraction for ignition, input on record BUR_IGN (default = 0.55);

XINDCH = maximum diluent mole fraction for ignition during DCH, input on record BUR_IGN (default = XMSCIG).

If all three tests are satisfied (Equations (2-3) through (2-5)) (i.e., there is enough hydrogen and/or carbon monoxide, enough oxygen, and not too much steam and/or carbon dioxide), a burn is initiated. The burn duration and combustion completeness are discussed in Sections 2.3 and 2.4. If too much steam and carbon dioxide is present, the control volume is considered to be inert, and is identified as such in the printed edits. A

message is printed to the output file and to the special message file and a plot dump is written (if specified by the user) when a deflagration begins and ends in any control volume.

2.3 Combustion Completeness

In MELCOR, deflagrations are not required to be complete; that is, all of the combustible gases present in a control volume at the start of a deflagration are not required to be burned during the deflagration. The *combustion completeness* is used to determine the amounts of combustible gases that should be present in a control volume at the end of an incomplete burn. In the BUR package, the combustion completeness, CC, is defined as

$$CC = 1 - {}^{Y_{min}}/_{Y_{max}} \tag{2-6}$$

where Y is given by the LeChatelier formula,

$$Y = X_{H2} + X_{CO}(\mathrm{YH2CC/YCOCC}) \tag{2-7}$$

and

Y_{max} = value of LeChatelier formula evaluated at the start of the burn (initial amount of combustibles);

Y_{min} = value of LeChatelier formula that is desired at the end of the burn (final amount of combustibles);

YH2CC = XH2CC, if DCH is not occurring, or

= XH2CCD, if DCH is occurring;

YCOCC = XCOCC, if DCH is not occurring, or

= XCOCCD, if DCH is occurring;

XH2CC = hydrogen mole fraction for calculating combustion completeness, input on record BUR_COM (default = 0.08);

XH2CCD= hydrogen mole fraction for calculating combustion completeness during DCH, input on record BUR_COM (default = XH2CC);

XCOCC = carbon monoxide mole fraction for calculating combustion completeness, input on record BUR_COM (default = 0.148);

XCOCCD = carbon monoxide mole fraction for calculating combustion completeness during DCH, input on record BUR_COM (default = XCOCC).

The combustion completeness is first evaluated by the method described below, then it is used to determine the value for Y_{min} for the current deflagration in the control volume. The burning rate is adjusted as necessary (see Section 2.5) to achieve this value at the end of the burn.

The combustion completeness can be input as a constant value, calculated from a user-specified control function, or calculated from a correlation. The default correlation for combustion completeness, which was obtained from the HECTR 1.5 code [1], and derived from experimental data, is dependent on the mole fraction of combustible gases present at the start of the burn, Y_{max}, and is given by

$$CC = 0.0 \ \textit{for} \ Y_{\max} \le 0.03746 \tag{2-8}$$

$$= 23.4116(Y_{\max} - 0.03746) \ \textit{for} \ Y_{\max} > 0.03746 \tag{2-9}$$

The constants in this correlation have been implemented in sensitivity coefficient array 2202.

2.4 Burn Duration

The burn duration is calculated by dividing a user-specified characteristic dimension by the *flame speed*. The flame speed can be input as a constant value, calculated from a user-specified control function, or calculated from a correlation. Optional input can be specified to determine the flame speed with a different constant, control function or correlation when DCH is occurring in the control volume. The default correlation, obtained from the HECTR 1.5 code, was derived from experimental data. However, few data were available regarding the effect of large amounts of diluents (steam and carbon dioxide) on flame speed, so the correlation is questionable in mixtures with high diluent concentration. For these mixtures, sensitivity studies should be conducted to bound the expected pressure rises. The default correlation for the flame speed, V, is

$$V = V_{base} x \ C_{dil} \tag{2-10}$$

where

$$V_{base} = 59.2Y_{\max} + 1.792 \text{ if } 0.0 \le Y_{\max} \le 0.1, \tag{2-11}$$

$$= 172.88Y_{\max} - 9.576 \text{ if } 0.1 < Y_{\max} \le 0.2, \tag{2-12}$$

$$= 50.Y_{max} + 15. \quad \text{if } 0.2 < Y_{max} \le 0.3, \tag{2-13}$$

$$= -50.Y_{max} + 45. \quad \text{if } 0.3 < Y_{max} \le 0.4, \tag{2-14}$$

$$= -75.Y_{max} + 55. \quad \text{if } 0.4 < Y_{max} \le 0.6, \tag{2-15}$$

$$= -64.3Y_{max} + 48.58 \quad \text{if } 0.6 < Y_{max} \le 1.0 \tag{2-16}$$

$$C_{dil} = \max(0.05, 1.0 - 4.53XD + 5.37XD^2) \quad \text{if } 0.0 \le Y_{max} \le 0.2 \tag{2-17}$$

$$\begin{aligned} &= \max(0.05, 1.0 - 4.53XD) + 5.37XD^2)(0.3 - Y_{max})/0.1 \\ &+ \max(0.0, 1.0 - 1.29XD)(Y_{max} - 0.2)/0.1 \quad \text{if } 0.2 < Y_{max} \le 0.3 \end{aligned} \tag{2-18}$$

$$= max(0.0, 1.0 - 1.29XD) \text{ if } 0.3 \le Y_{max} \tag{2-19}$$

XD = diluent concentration $(X_{H2O} + X_{CO2})$.

The constants in this correlation have been implemented in sensitivity coefficient array 2200. The burn duration time, t_{comb}, is calculated by dividing the flame speed into a user-specified characteristic dimension of the control volume, CDIM (or CDDH when DCH is occurring), input on record BUR_BRT:

$$\begin{aligned} t_{comb} &= CDIM / V \text{ if DCH is not occurring or} \\ &= CDDH / V \text{ if DCH is occurring.} \end{aligned} \tag{2-20}$$

2.5 Combustion Rate

The combustion rate (amount of hydrogen, carbon monoxide, and oxygen converted to steam and carbon dioxide per timestep) is not constant during a burn. Rather, it is adjusted at each timestep to account for inter-compartment flows and gas sources in an effort to match the desired final conditions. In other words, the combustion rate is adjusted so that the mole fractions corresponding to the calculated combustion completeness and the desired burn duration are simultaneously achieved. At each timestep, the burn rate, YRATE, is calculated as

$$YRATE = (Y(t) - Y_{min}(t))/(t_o + t_{comb} - t) \tag{2-21}$$

where

t_o = time that burn was initiated, and

t = current time in calculation.

Once the rate is calculated, it is used to determine the decrease in the inventory of the combustible gases for the current MELCOR system timestep:

$$DELH2 = X_{H2}(t) \cdot YRATE \cdot \frac{\mathrm{DT}}{Y(t)} \quad (2\text{-}22)$$

$$DELCO = X_{CO}(t) \cdot YRATE \cdot \frac{\mathrm{DT}}{Y(t)} \quad (2\text{-}23)$$

where

DELH2 = decrease in hydrogen moles in the control volume during the timestep from combustion,

DELCO = decrease in carbon monoxide moles in the control volume during the timestep from combustion, and

DT = MELCOR system timestep (s).

At the end of the burn, the value Y_{min} would be reached exactly if there were no flow or sources. These values are updated on every timestep to reflect the changing conditions. DELH2 and DELCO are constrained to prevent burning more moles of either gas than are present in the control volume.

The energies of formation are included in the water and noncondensible gas equations of state. With this formulation, simply changing the relative masses of the reactants and products automatically results in the appropriate pressure and temperature increase. Thus, it is not necessary to calculate a combustion energy release to a control volume. The total mass and energy of a control volume are not changed by the BUR package, but the masses of individual species are changed to reflect the reactions listed in Section 2.1. (That is, DELH2, DELCO, and 0.5 · DELH2 + 0.5 · DELCO moles of hydrogen, carbon monoxide, and oxygen are subtracted from the control volume while DELH2 and DELCO moles of steam and carbon dioxide are added to the control volume.) Because the specific enthalpy of each species properly accounts for the energy of formation, the conversion of the reactants to the products increases the temperature and pressure of the control volume with combustion, even though the total energy remains unchanged.

2.6 Propagation Criteria

Propagation of combustion from a control volume to connected control volumes is allowed after a user-controlled time period has elapsed. This delay is intended to account for the time it would take for a flame to reach the edge of a control volume if a flame front were being modeled. Different delay periods may be specified depending upon whether DCH

is occurring in the control volume. Propagation occurs if the propagation criteria are satisfied in the connected control volume. The propagation delay, t_{prp}, is calculated to be

$$t_{prp} = FRAC \cdot t_{comb} \qquad (2\text{-}24)$$

where

FRAC = TFRAC, if DCH is not occurring in the control volume, or

= TFDH, if DCH is occurring in the control volume; and

TFRAC = propagation time fraction input on record BUR_BRT (default = 0)

TFDH = override value of TFRAC during DCH, input on record BUR_BRT (default = TFRAC).

Note that if TFRAC equals zero, propagation is possible as soon as a control volume begins burning. If TFRAC equals 1.0, propagation is only considered at the end of the control volume burn.

For propagation, LeChatelier's formula is still applicable if appropriate values are used for the L parameters. Propagation is allowed if the following inequality is satisfied

$$X_{H2} + X_{CO}\left(L_{H2,prp} / L_{CO,prp}\right) \geq L_{H2,prp}$$

where

$L_{H2,prp}$ = XH2PUP, for upward propagation, or

= XH2PHO, for horizontal propagation, or

= XH2PDN, for downward propagation;

$L_{CO,prp}$ = XCOPUP, for upward propagation, or

= XCOPHO, for horizontal propagation, or

= XCOPDN, for downward propagation;

XH2PUP = hydrogen mole fraction limit for upward propagation, input on record BUR_COM (default = 0.041).

XH2PHO= hydrogen mole fraction limit for horizontal propagation, input on record BUR_COM (default = 0.06).

XH2PDN= hydrogen mole fraction limit for downward propagation, input on record BUR_COM (default = 0.09).

XCOPUP = carbon monoxide mole fraction limit for upward propagation, input on record BUR_COM (default = 0.125).

XCOPHO = carbon monoxide mole fraction limit for horizontal propagation, input on record BUR_COM (default = 0.138).

XCOPDN = carbon monoxide mole fraction limit for downward propagation, input on record BUR_COM (default = 0.15).

The propagation direction is determined directly from the flow path input using the *from* and *to* elevations (see the FL Package Users' Guide). If a flow path is not open, or if the flow path is covered by water, propagation is not allowed. Note that the presence of a check valve is not taken into account when determining whether a flow path is open.

A message is printed to the output file and to the special message file and a plot dump is written (if specified by the user) when a deflagration due to propagation begins in any control volume.

2.7 Detonation

MELCOR does not contain a detonation model. However, tests are performed in each control volume, and a warning message is written indicating the possibility of a detonation if all of the following mole fractions limits are satisfied:

$$X_{H2} > \mathrm{XH2DET} \tag{2-25}$$

$$X_{O2} > \mathrm{XO2DET} \tag{2-26}$$

$$X_{H2O} < \mathrm{XH2ODT} \tag{2-27}$$

where

XH2DET = minimum hydrogen mole fraction for detonable mixture, input on record BUR_DET (default = 0.14),

XO2DET = minimum oxygen mole fraction for detonable mixture, input on record BUR_DET (default = 0.09), and

XH2ODT = maximum steam mole fraction for detonable mixture, input on record BUR_DET (default = 0.30).

No detonation calculation is performed when a detonable mixture is detected. The warning message is written, but the calculation continues under the control of the deflagration model. The detonation model is mainly intended as a user convenience to flag potentially dangerous conditions that may require separate analysis.

2.8 Diffusion Flame Model

The diffusion flame model is intended to model the burning of hydrogen entering a control volume under DCH conditions. Under such conditions, the hydrogen enters accompanied by hot melt particles which act as igniters, so that the conditions for ignition and burning are quite different from those for a bulk burn. The diffusion flame implementation is a simple model that burns combustible gas passing through a flow path and entering a control volume containing oxygen, subject to ignition criteria. These are defined on the BUR_DIF input records (see BUR Users Guide), new in MELCOR 1.8.5. This general approach is used both in MELCOR and in CONTAIN [3].

As implemented in MELCOR, a combustion completeness criterion is used as specified on the BUR_CF cards and described in the BUR Users Guide. No flame speed or duration calculation is performed, and the ignition criteria are the same as for deflagration. Additionally, the airborne DCH debris temperature must be greater than a lower limit specified by the C2203 sensitivity coefficient (default = 600K). See BUR Users Guide.

The ignition limits for the diffusion flame model are set to insure virtually complete combustion with any oxygen present in the receiving volume even if large amounts of inert gases are present. This is done to simulate the expected effect of hot DCH debris in the incoming gas on hydrogen recombination and is similar to the model used in calculating DCH with CONTAIN. The diffusion flame implementation assumes burning occurs whenever the ignition criteria are met. The effects of flashback or blowout are not considered. (Note: flashback occurs when the flame is swallowed back into the combustible gas source; blowout occurs when the flame front moves away from the gas source so rapidly that it is extinguished).

3. Timestep Control

When a burn first occurs, the Burn Package requests a fallback after which the calculation continues with the timestep value specified by the BUR_TIM record. In addition, as the burn approaches completion, tests are included to prevent excessive overshoot of the originally desired burn-completeness values. In particular, a timestep is repeated if the originally desired burn-completeness values are crossed during that timestep and either (a) the combustible gas concentration is more than 0.5% different, or (b) the diluent concentration is more than 1% different from the originally desired burn completeness values. These maximum overshoots can be adjusted through sensitivity coefficient C2201.

4. References

1 S. E. Dingman, et al., HECTR Version 1.5 User's Manual, SAND86-0101, NUREG/CR-4507 (April 1986).

2 G. G. Weigand, ed., Thermal-Hydraulic Process Modeling in Risk Analysis: An Assessment of the Relevant Systems, Structures, and Phenomena, SAND84-1219, NUREG/CR-3986 (August 1984).

3 K. K. Murata et al., Code Manual for CONTAIN 2.0: A Computer Code for Nuclear Reactor Containment Analysis, SAND97-1735, NUREG/CR-6533 (December 1997).

Cavity (CAV) Package

The MELCOR Cavity (CAV) package models the attack on the base-mat concrete by hot (often molten) core materials. The effects of heat transfer, concrete ablation, cavity shape change, and gas generation are included, using models taken from the CORCON-Mod3 code. The coding of the models is identical to that in CORCON-Mod3, but interfaces have been modified for integration into the MELCOR framework. This integration couples the Cavity package models to thermal-hydraulic boundary conditions in the Control Volume Hydrodynamics (CVH) package, to sources of core debris from the Core (COR) and/or Fuel Dispersal Interactions (FDI) package, and to the standard MELCOR input, output, plotting, and restart capabilities. The fission-product release models in CORCON-Mod3—originally developed as the separate VANESA code—are included in MELCOR as part of the RadioNuclide (RN) package.

This Reference Manual provides an overview of modeling in the CAV package. User input for running MELGEN and MELCOR with the CAV package activated is described in the CAV Package Users' Guide. The fission-product release models (VANESA) and available input are described in the RN Reference Manual and Users' Guide, respectively.

Contents

List of Figures

List of Tables

1. Introduction

The Cavity (CAV) package in MELCOR models the attack on the base-mat concrete by hot, often molten, core materials. The effects of heat transfer, concrete ablation, cavity shape change, gas generation, and debris/gas chemistry are included. The package consists of models taken from the CORCON-Mod3 code [1] together with all necessary interfaces to the MELCOR database and to other packages in MELCOR.

Before the initial release version of CORCON-Mod3 [2] was incorporated into MELCOR and into CONTAIN [3], a number of modifications were made to the coding that had no effect on results calculated by the stand-alone code, but allowed the direct use of all routines containing phenomenological models and properties data *without modification* in the systems codes. These changes involved a restructuring of the internal database and of the interfaces to input and output routines (including diagnostics and plotting) and to routines that provide boundary conditions for the CORCON models.

Boundary conditions for temperature and pressure used by the cavity models are obtained from an associated CVH control volume, rather than from user input as in the stand-alone CORCON. Any overlying coolant (water) pool is considered part of the boundary condition rather than part of the cavity model and is modeled by CVH. Heat and evolved gases are delivered as sources to the associated CVH volume.

Debris from the Core (COR) package, the Fuel Dispersal Interactions (FDI) package, or the External Data File (EDF) package is ordinarily deposited into the cavity through the Transfer Process (TP) package. However, initial contents may also be defined in CAV input and arbitrary addition rates may be prescribed by input to the TP package. When debris is deposited, no spreading calculation is performed because it is assumed to spread instantaneously to the maximum area permitted by the cavity geometry.

The CAV packages uses the CORCON-Mod3 properties routines, which are currently independent of the general Materials Properties (MP) package in MELCOR.

The phenomena modeled by the CAV package may be treated in more than one location in a MELCOR calculation. Transfer of material between cavities is allowed based on three tests: axial rupture, radial rupture, or a transfer triggered by a Control Function. Each of the three types of rupture (axial, radial, and triggered) can overflow to a separate cavity, but only “one-way” transfers are allowed. That is, if material can overflow from cavity 1 to cavity 2, it is not permitted to flow from cavity 2 back to cavity 1, either directly or through intermediate cavities. These ruptures can be used to model such phenomena as failure of the pedestal in a BWR Mk I or of the diaphragm slab in a BWR Mk II. Triggered transfers may also simulate (in a qualitative way) the effects of the spreading of debris across a flat floor.

The VANESA model [4] was integrated into CORCON-Mod3 to calculate the release of fission products and the generation of aerosols from debris in the cavity. The structure

of MELCOR requires that radionuclides associated with debris in the cavity be treated by the RadioNuclide (RN) package, which maintains time-dependent inventories for each RN class in each cavity. The relevant subroutines from CORCON-Mod3 were therefore made part of the RN package. They are identical to the routines in the latest stand-alone version of CORCON-Mod3 and in CONTAIN; an interface is provided through a utility entry in the RN package that duplicates the functionality in the stand-alone code. See the RN package Reference Manual for more details.

Several options for direct user input of internal heating of the debris by fission products are allowed, but this heating is ordinarily calculated by the RN and DCH (Decay Heat) packages, based on RN inventories. Therefore, the effects on internal heating of relocation of debris into or between cavities, as well as the effects of RN releases within each cavity, are automatically accounted for.

2. Phenomenology

This section gives a qualitative description of the processes modeled in the CAV package in MELCOR, and the physical picture on which the models are based. The information is largely derived from Section 2.1 of the CORCON-Mod3 Manual [1]. Interfaces to other MELCOR packages are noted in the discussion.

The attack of core debris on concrete in a light water reactor is primarily thermal and may be considered quasi-steady for much of the period of a reactor accident. Decay heat and heat from chemical reactions is generated in the debris and is transferred either through its top surface or to the concrete floor. Boundary conditions at the surface, including temperature and the presence or absence of water, are obtained from the associated control volume in CVH. Heat lost from the cavity top surface is treated by CVH as a source into that control volume.

The quasi-steady partition of the heat transfer to the concrete floor and through the debris top surface is determined by the ratio of the corresponding thermal resistances. Thus, debris behavior and concrete ablation are dominated by conservation of energy, with heat transfer relations providing the most important constitutive relations.

Under the conditions visualized by the CORCON developers, the heat flux to the concrete floor is sufficient to decompose it, releasing water vapor (from both adsorbed water and hydroxides) and carbon dioxide (from carbonates) and to melt the residual oxides. The surface of the concrete is typically ablated at several centimeters per hour and molten oxides and molten steel from reinforcing bars in the concrete are added to the debris pool. The decomposition gases are strongly oxidizing at debris temperatures and are reduced, primarily to hydrogen and carbon monoxide, on contact with metals in the debris. Ultimately, the reacted and unreacted gases enter the atmosphere above the debris pool, where they may or may not burn immediately. (Modeling of these containment phenomena is not included in CORCON.) These gases (with appropriate enthalpies) are

treated as sources in the associated control volume in CVH. The possibility that the combustible gases might burn is considered by the BUR package.

The full concrete response is extremely complicated, with elements of ablation, transient conduction, decomposition of hydroxides and carbonates in advance of the ablation front, and transport of gases and liquid water through the pores of the concrete. Further, the length scale of the temperature profile is often comparable to the size of the coarse aggregate in concrete, making any assumption of homogeneous properties questionable.

In CORCON and in CAV, concrete response is modeled as quasi-steady ablation. The thermal diffusivity of concrete is extremely small, a few times 10^{-7} m^2/s. Over the time scale of interest in cavity phenomena (hours), the amount of heat that can be transferred into concrete (by transient conduction) under nonablative conditions is usually small compared to the amount of heat that must be removed from core debris through other mechanisms to maintain its temperature below the ablation temperature. Therefore, if the debris temperature is below the ablation temperature, the concrete floor surface is modeled as an adiabatic boundary.

Gas released at the bottom of the debris pool is assumed to rise through it as bubbles. Gas released at the side of the pool may also form bubbles that rise to the surface. At sufficiently high gas release rates, a stable gas film may form at either the bottom or side interfaces. Gas bubbles rising through the debris pool increase its volume. This “level swell” increases the depth of the pool and area of its radial interface with the concrete floor.

The rising bubbles also promote the production of aerosols containing fission products stripped from the fuel debris. The processes involved, reactive vaporization and bubble bursting, are treated by the VANESA model [4] in the RN package in MELCOR. This model calculates the removal and relocation of fission products and the resulting sources of aerosols for the MAEROS aerosol physics model (also part of the RN package). All necessary data concerning the temperature and bulk composition of the debris and the gas generation rates are passed by CAV to a utility entry in RN; the fission product inventories themselves are part of the RN database. The subroutines that implement VANESA in the RN package are identical to those that implement it in CORCON-Mod3.

Experimental evidence (cited in Reference 1) shows that the various oxidic species in the melt are highly miscible, as are the metallic species, but that the two groups are mutually immiscible. Previous versions of CORCON assumed that the core debris would stratify into distinct layers based on the relative densities of the phases. The passage of gas bubbles through the interface between layers can overcome this separation if the gas flux is high or the density difference is small by entraining droplets of the lower (denser) material and mixing them into the upper one. If entrainment occurs, the degree of mixing achieved is determined by a balance between entrainment and reseparation as the denser droplets settle out under the influence of gravity. The debris may therefore be

fully stratified, partially mixed, or fully mixed, and the state may change as the densities and gas fluxes change during a debris-concrete interaction.

There is a possibility that an overlying coolant layer (water) could interact with molten debris so as to break it up and form a coolable debris bed. In the MAAP code [5], this breakup and quenching is assumed to occur; it is not considered in CORCON, nor is it included in the current version of the MELCOR CAV model.

As the core-concrete interaction progresses, the debris pool grows as concrete oxides are added to it; its surface area increases, and internal heating decreases. Therefore, debris temperatures and heat fluxes decrease, and the possibility of refreezing arises. Substantial freezing of the metal phase may occur. However, the large internal heating and small thermal conductivity of the oxidic phase prevent the formation of steady, solid crusts thicker than a few centimeters. Therefore, unless the debris is spread over an extremely large area, the interior of the oxidic phase remains molten for a long time, probably for weeks.

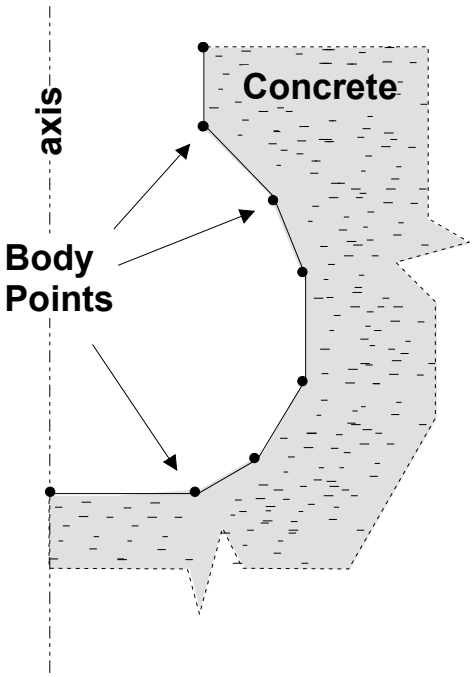
3. Models

Documentation of CORCON-Mod3 [1] remains the primary reference for most of the submodels in the Cavity package. The following subsections briefly summarize the material contained there, while noting modifications made for incorporation into MELCOR.

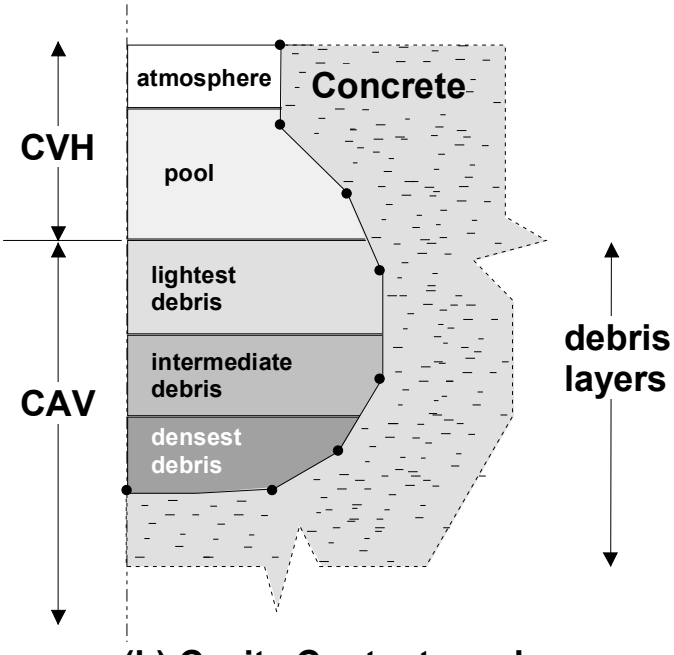
3.1 System Components

The physical system considered by the Cavity package consists of an axisymmetric concrete cavity, a multilayered debris pool, and a set of boundary conditions (provided by CVH) at the top surface of the debris, as illustrated in Figure 3.1.

The shape of the concrete cavity is described by a series of so-called *body points* lying in a vertical cross-section of the concrete surface. The initial shape is defined by user input. The concrete itself is described by specifying an average chemical composition; its thermochemical properties are then obtained from an internal database of properties for the component species. A number of standard compositions are available by name as built-in defaults, or the user may define composition and melting temperatures through input.



(a) Cavity Geometry



(b) Cavity Contents and Boundary Conditions

Figure 3.1 Cavity System Components

The modeling assumes that all oxidic species in the debris are mutually miscible, as are all metallic species, but that oxides are not miscible with metals. If the densities of the phases are different, the debris tends to separate into distinct oxidic and metallic phases under the influence of gravity, but this stratification may be partially or completely overcome by the stirring effect of gas bubbles. If the density difference is sufficiently small and the gas bubbles sufficiently large, droplets of a lower (denser) layer can be entrained across the interface to mix with a lighter layer above it.

The debris pool is modeled as a number of layers filling some part of the concrete cavity. Pure-phase and mixed-phase layers may be included, and the ordering of the layers is assumed to be determined by their densities, with the densest on the bottom and the lightest on top. Many configurations are possible, as discussed in Section 3.2. Layer volumes, including the swelling effects of gas bubbles, determine the elevations of layer interfaces and of the debris surface.

3.2 Debris Layering and Mixing

Five possible types of debris layers are considered in CORCON; each has a conventional three-letter designation in the associated documentation. In order of increasing density they are:

LOX: Pure oxide, less dense than the metallic phase;
LMX: Mixed phases, less dense than the metallic phase;
MET: Pure metal;
HMX: Mixed phases, more dense than the metallic phase; and
HOX: Pure oxide, more dense than the metallic phase.

If only oxides are present, the debris is called LOX by convention. The possibility of creating mixed-phase layers was introduced as part of the enhanced modeling in CORCON-Mod3. The major assumptions concerning these mixed layers is very specific:

> The LMX layer is formed by entrainment of metal from MET or HMX into a previously existing LOX layer, and consists of a suspension of discrete droplets of metal in a less-dense continuous oxidic phase. The mixing is assumed to be complete so that the LOX layer is converted to an LMX layer in the process; LMX and LOX cannot exist simultaneously. The entrainment competes with settling of the denser metal droplets from LMX back into the lower layer (or to form a new MET layer if there is no lower metal-containing layer present).
>
> The HMX layer is formed by entrainment of oxides from HOX into a previously existing MET layer, and consists of a suspension of discrete droplets of oxide in a less-dense continuous metallic phase. The mixing is assumed to be complete so that the MET layer is converted to an HMX layer in the process; HMX and MET cannot exist simultaneously. The entrainment competes with settling of the denser

oxide droplets from HMX back into HOX (or to form a new HOX layer if there is none present).

Under these assumptions, there are 15 possible configurations of the debris. These can be summarized as follows:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LOX				X	X	X					X				X
LMX							X	X	X	X		X	X		
MET	X			X			X				X	X		X	
HMX		X	X		X	X			X	X					
HOX	X	X		X	X		X	X	X						

where “X” denotes the presence of the layer.

Three options are available for the treatment of layering and mixing of debris in CORCON. They are (1) enforcement of complete mixing, (2) enforcement of complete stratification, and (3) mechanistic modeling of the entrainment and separation processes. The first of these (complete mixing) is the default in the CAV package in MELCOR, but the user may specify any of the options by input of MIXING on the CAV_U record.

3.2.1 Enforced Mixing

This is the simplest of the options, with the debris always considered to form a single layer. If both metals and oxides are present, the layer is HMX or LMX (configuration 3 or 13), depending on the relative densities of the phases. If there is only a single phase, it is either MET or LOX (configuration 14 or 15). As noted previously, this is the default treatment in MELCOR.

3.2.2 Enforced Stratification

This was the only option available in CORCON in versions prior to Mod3, in which the possible creation of heterogeneous mixtures of metals and oxides was not considered. It was, therefore, the only option available in versions of MELCOR prior to 1.8.3.

When this option is specified, the possibility of two oxidic layers, physically separated by a metallic layer, is allowed for. If the initial oxide phase is sufficiently rich in UO_2 (fuel) to be more dense than the initial metallic phase, it is assumed to form an oxidic layer beneath the one containing the metals. An oxide slag, rich in concrete and steel oxides and less dense than the metals, accumulates on top of the metal layer. Thus, the most general structure of the debris pool is a light oxide layer (LOX), over a metallic layer (MET), over a heavy oxide layer (HOX).

This three-layer configuration (configuration 4) can persist until dilution by less dense concrete oxides renders the HOX layer less dense than the MET layer. The configuration is then (instantaneously) converted to one containing only MET and LOX (configuration 11), with the latter layer combining the previous contents of HOX and LOX. Addition of UO_2-rich debris to a debris pool in the LOX-over-MET configuration can result in an oxide mixture that is denser than the contents of MET. When this occurs, the LOX is eliminated and the configuration is (instantaneously) converted to one of MET over HOX (configuration 1). These changes in configuration are effected by checking the relative densities of adjacent layers at every step of the calculation, and relocating and/or combining the layers as appropriate.

3.2.3 Mechanistic Mixing

The most general option uses mechanistic models for entrainment and separation developed by Green [6, 7, 8] to predict the occurrence and extent of mixing. One consequence of this modeling is to eliminate the instantaneous change in debris configuration (often referred to as “layer flip”) resulting from an insignificant change in the relative densities of the debris phases. Instead, the phases become increasingly strongly mixed whenever their densities approach equality (unless there is no gas flow to drive the mixing).

The entrainment model assumes that if the bubbles are large enough, as they pass through the interface between two layers, these bubbles may carry material from the lower layer into the upper layer. The critical bubble diameter to include entrainment depends on density ratios and on the surface tension of the liquid-liquid interface; above the threshold, a correlation is used to determine the volume of condensed-phase material entrained by each gas bubble. The separation model is based on the terminal velocity of falling droplets of a size corresponding to the critical Weber number for the onset of droplet oscillations.

Competition between these processes defines the net rate of mixing or separation at the various layer interfaces. The model considers entrainment of oxides from HOX into HMX or LMX, or into MET to form HMX, and of metal from MET or HMX into LMX or into LOX to form LMX. It also considers the possibility that a mixed layer is unstable and separates to produce a new HOX layer below HMX or a new MET layer below LMX.

After release of the initial version of CORCON-Mod3 [2], the numerical implementation of the models into MELCOR was modified to provide numerical stability with reasonable timesteps. The entrainment rate depends primarily on the gas flux; therefore, over a finite timestep,

$$\dot{m}_e(t) \approx \dot{m}_e(0) \tag{3-1}$$

However, the separation rate is proportional to the mass of the discontinuous phase in the mixed layer, and has the form

$$\dot{m}_s(t) = \frac{M_D(t)\ v_{settle}}{L_M} \tag{3-2}$$

where v_{settle} is the settling velocity, M_D is the mass of droplets suspended in the mixed layer, and L_M is thickness of that layer.

The mass of suspended droplets therefore satisfies

$$\frac{dM_D}{dt} = \dot{m}_e(t) - \dot{m}_s(t) \approx \dot{m}_e(0) - \frac{M_D(t)}{\tau_s} \tag{3-3}$$

where

$$\tau_s \equiv \frac{L_M}{v_{settle}} \tag{3-4}$$

is the time constant for separation. Equation (3-3) has the analytic solution.

$$M_D(t) = M_D(0) + [\dot{m}_e(0)\tau_s - M_D(0)]\left(1 - e^{-t/\tau_s}\right) \tag{3-5}$$

Equation (3-5) expresses the fact that entrainment and separation approach a balance where the mass of suspended droplets M_D^{ss} is

$$M_D^{ss} = \dot{m}_e\tau_s \tag{3-6}$$

with a characteristic time τ_s. Independent treatment of the competing processes is numerically unstable unless the timestep, Δt, is less than τ_s, and the results are dependent on timestep unless Δt is *much* less than τ_s. Because the time constant may be relatively short compared to the rates at which conditions are changing, the revised version of CORCON-Mod3 applies the analytic solution given by Equation (3-5) over a timestep. This requires moving a net mass

$$\Delta M_e^{net} = (\dot{m}_e^0\tau_s - M_D^0)\left(1 - e^{-\Delta t/\tau_s}\right) \tag{3-7}$$

from the lower layer to the upper layer during the timestep, where superscript 0 denotes evaluation at the start of the step. If the net move is positive, it must be limited to the contents of the lower layer. If it is negative, it cannot—by its very form—exceed the mass of droplets initially suspended in the upper layer. This change in numerical implementation has eliminated almost all of the instabilities observed in layer mixing in the initially released version of CORCON-Mod3.

3.3 Energy Generation and Heat Transfer

The fuel/concrete interaction is driven primarily by decay heat power generated within the debris pool, with heat from oxidation reactions also contributing. In stand-alone CORCON, the decay heating is calculated by an internal model based on an initial fission

product inventory and fits to the decay powers for each of the 27 elements in CORCON. In MELCOR, this heating is calculated by the RN and DCH packages; the model is conceptually very similar to that in CORCON (see the RadioNuclide (RN) and Decay Heat (DCH) Package Reference Manuals and Reference 1), but the CAV database contains no information on the location—or relocation—of the fission products. (The exact model used in stand-alone CORCON is therefore not available in MELCOR even as an option.) Heat sources based on control functions and/or tabular functions are also permitted, primarily for simulation of experiments.

For the calculation of energy conservation, each debris layer is treated as a lumped mass with a single (average) temperature. The heat flux between the interior of each layer and each of its interfaces (with another layer, with concrete, or with the pool or the atmosphere in the bounding control volume) is treated separately. Continuity of the heat flux determines the temperature of each interface.

The possible heat transfer regimes within each debris layer are conduction and natural convection, based on conventional correlations, and bubble-enhanced convection based on Kutateladze [9] and surface renewal [10] models. The correlations are implemented in such a way that they reproduce correlations for convective heat transfer in internally heated fluid layers (in the absence of gas flows) developed by Kulacki and co-workers [11, 12, 13] with a maximum error of 30 percent and an average error closer to 10 percent. An enhancement factor developed by Farmer [14] is applied at the top surface of the debris (adjacent to the coolant or the atmosphere) to account for the greater surface area of the unstable surface.

The modeling includes the possibility that the interior of a layer may be fluid, with heat transfer by convection, while one or more of its axial and radial surfaces is covered by a solid crust, with heat transfer by conduction [15]. In all cases, only one-dimensional effects are considered, and the situation is assumed to be quasi-steady.

Losses from the surface are calculated, based either on radiation and convection in the absence of overlying water or on a complete pool boiling curve in its presence. The representation of the boiling curve is the one used in CORCON [1], and includes convection, nucleate boiling, transition boiling, and film boiling regimes. In the film boiling regime, the effects of coolant subcooling and of gas barbotage (injection of noncondensible gas at the coolant interface), both of which can greatly increase both the film boiling heat flux and the temperature at which the film collapses (the Leidenfrost point), are also included.

The concrete surface is treated using a quasi-steady ablation model. If concrete is ablating, it presents a constant temperature boundary condition defined by the ablation temperature, T_a. This temperature is obtained either from internal data or user input. Under quasi-steady conditions, changes in the sensible heat content of the preheated region in advance of the ablation front may be neglected. (As mentioned in Section 2, the thermal diffusivity of concrete is extremely small. The *total* heat content of this region

is therefore small and is neglected.) The rate of ablation (in kg/m^2-s) is then proportional to the heat flux (W/m^2) from the debris to the concrete surface. Their constant of proportionality is simply the inverse of the heat of ablation h_a.

If the heat flux to a concrete surface at an assumed temperature of T_a would be negative, no ablation can be taking place, and heat transfer can affect only the thermal boundary layer in the concrete. Under these conditions, change in the heat content of this boundary layer is neglected and the concrete surface is treated as an adiabatic boundary. Further decomposition of concrete in advance of ablation is also neglected.

An additional thermal resistance is included between the debris and the concrete. CORCON-Mod3 allows this resistance to be calculated using either a gas film or a slag film model. In each case, separate models are provided for the bottom and side surfaces of the debris.

The gas film models are based on the assumption of a gas film between the debris and the concrete. An analog of Taylor-instability-bubbling film boiling is used on nearly horizontal surfaces [16], and an analog of attached-flow film boiling is used on strongly inclined surfaces. A transition from bubbling to flow is made over a range of inclination angles. Details of the model are presented in Reference 1.

A detailed slag film model was developed by Bradley [17], based on a picture of transient growth and removal. He found that when the resulting thermal resistance of the slag film was combined with the resistance within the debris layer, the net heat transfer coefficient between the interior of the debris and the concrete surface could be adequately represented as a constant multiple (0.29) of the latter coefficient over a wide range of conditions. The heat transfer coefficient for the slag film model is therefore calculated in CORCON as 0.41 times the heat transfer coefficient between the interior of the debris and its surface, for either the bottom and side surfaces of the debris, so that the net heat transfer coefficient is 1.0*0.41/(1.0+0.41) = 0.29 times the internal heat transfer coefficient.

The model to be used may be selected independently for the bottom and side surfaces of the debris. The default in MELCOR 1.8.3 and later versions is to use the gas film model in both places, consistent with previous versions of MELCOR. The user may specify which model is to be used on the bottom and/or side surfaces by input of GFILMBOTT or GFILMSIDE on the CAV_U record, as described in the CAV Package Users' Guide. (There is no default for the choice of models in stand-alone CORCON-Mod3, and the Manual [1] provides no recommendation.)

3.4 Concrete Ablation and Cavity Shape Change

In steady-state ablation, the incident heat flux and the ablation rate are directly proportional; the ratio is simply the volumetric ablation enthalpy. Therefore, the heat flux to the concrete at each body point in the cavity profile is used to calculate the local

ablation rate. A new position of the body point is then calculated, displaced along the local normal to the surface. To maintain calculational stability, the cavity profile is then rezoned, and the body points are interpolated back onto a series of guiding lines called rays, as illustrated in Figure 3.2. The effect of the rezone is that the body points must follow the rays, and their spacing along the cavity profile is constrained. As shown in the figure, all but one of the rays pass through a user-defined origin. The final ray lies parallel to the axis, through the outermost point on the flat bottom of the cavity, and serves to ensure that this flat bottom remains flat. The scheme evolved from the CASCET model [18] written by ACUREX/Aerotherm Corporation under contract to Sandia.

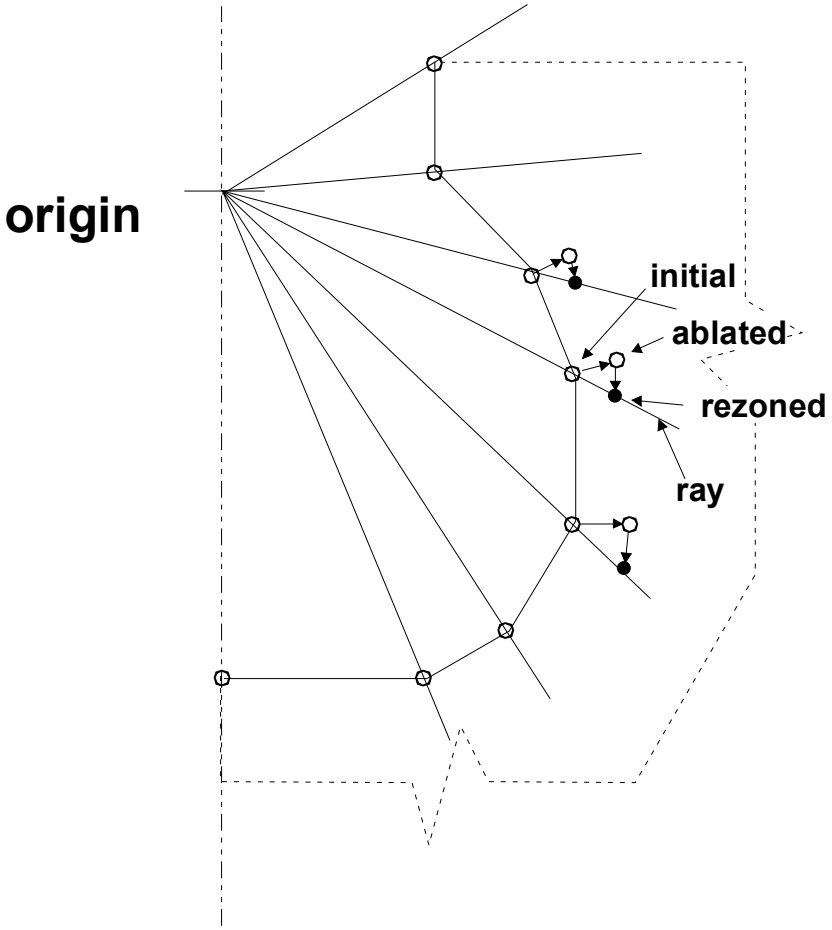


Figure 3.2 Position and Motion of Body Points

3.5 Chemistry

The chemistry considered in the Cavity package of MELCOR involves interactions between concrete decomposition products and metallic species in the debris pool. Equilibrium chemistry is assumed, without consideration of rate limiting effects. The calculational method is very general and is based on minimization of the total Gibbs

function for a metallic phase, a gaseous phase, and an oxidic phase. Each of the three phases is treated as an ideal solution; that is, the entropy of mixing is considered, but any heat-of-solution effects are ignored.

Two separate reactions are considered. The first involves reactions in the interior of the debris. For a pure metal layer, it is modeled as mutual equilibrium among the metal layer and the gas bubbles and concrete decomposition oxides passing through it. For a mixed-phase layer, the oxidic constituents of the layer are included as reactants. The primary effect is the oxidation of metals by the H_2O and CO_2 in the bubbles. However, if the metallic phase contains significant amounts of Zr, it can also reduce the concrete oxides to produce metallic Al, Ca, and Si. The user may specify that these reactions be ignored (as in older versions of CORCON) through input of CTOXYREA on record CAV_U; in this case, only the products of metal oxidation are included in the oxide phase.

The second reaction involves mutual equilibrium among the metal layer, the gas film at its radial boundary, and the products of metal oxidation. Concrete decomposition (and other) oxides are not included in this reaction.

The gaseous reactants are H_2O and CO_2, and the principal gaseous products are H_2 and CO. The full equilibrium calculation in CORCON predicts the formation of small amounts of additional gaseous species including hydrocarbons and various dissociation products such as atomic hydrogen. Most, if not all, of these species are predicted to occur in quantities insufficient to warrant their inclusion in the control volume inventories. To ignore them would violate mass conservation, and there is insufficient information to unambiguously convert them to “equivalent” amounts of significant species. The problem can be avoided by imposing constraints in minimization of the Gibbs function to eliminate consideration of any gaseous species other than H_2O, CO_2, H_2, and CO. This option was added to stand-alone CORCON-Mod 3 after its initial release, and is used in MELCOR. The results conserve mass and represent a restricted equilibrium state consistent with the modeling of atmosphere chemistry in MELCOR. If it were desired to include additional gases such as methane, only a trivial change to coding would be required. This is because the Gibbs function to be minimized has not been changed, but only the domain over which it is minimized.

The equilibrium calculation sometimes predicts the “coking” reaction in which CO_2 is fully reduced to condensed carbon (rather than simply to CO), primarily in the presence of metallic Zr. Because simulant experiments have not provided overwhelming evidence either for or against the occurrence of coking, the user is permitted to specify whether this reaction is permitted in CORCON. The default in MELCOR is to suppress the production of condensed carbon, but the user may enable this production by input of the COKE parameter on the CAV_U record.

In stand-alone CORCON, the chemistry includes an extremely simplified consideration of fission products; in MELCOR, this calculation is entirely replaced by the VANESA model [4] in the RN package.

3.6 Mass Transfer and Associated Heat Effects

The processes involved include the injection of concrete decomposition products (condensed and gaseous) into the debris pool, the addition of core and structural materials from other packages through the use of TP, the addition of debris from rupture or overflow of another cavity, and the production of condensed-phase materials from chemical reactions. Also involved is the transport of all these materials to their proper locations, whether within a debris layer in CAV or in a CVH volume. These processes modify both the mass inventories and the energy contents of the various debris layers and determine the mass source delivered to CVH and its associated enthalpy.

The masses and enthalpies of all debris layers are updated for mass transfer and associated heat transfer in two passes. These passes follow the paths of gaseous and condensed-phase concrete decomposition products, and of the products of chemical reactions involving these materials. The updating procedure is designed to account for successive interactions of transported materials, from the location where they are born to the location where they reside at the conclusion of the advancement procedure.

The first pass, upward through the debris pool, follows the rising gases and rising condensed-phase materials from concrete decomposition or melt/gas reactions. (The direction of motion of condensed-phase materials is determined by its density relative to the density of the local layer material.) The materials are thermally equilibrated with any layers they pass through, and their mass and energy are ultimately added to the layer where they end up (condensed phases) or to the associated CVH volume (gases). For condensed-phase materials, this final layer is assumed to be the first layer encountered that already contains that phase: HMX, MET, or LMX for metals, and any layer *but* MET for oxides. A new LOX layer may be formed to accommodate rising oxides from concrete ablation or metal oxidation or none already exists. Similarly, a new MET layer may be created to accommodate steel from melting reinforcing bars in concrete if the pool contains only a dense oxide layer.

Melt/gas chemical reactions are evaluated during this upward pass, following rising bubbles and flowing films. The composition of the layer involved is modified to reflect the effects of the reaction and, if the reaction takes place in the pure metal layer (MET), the condensed phase oxidic products are added to the rising inventory. The gas composition is modified appropriately, and the heat of reaction is assumed to remain with the layer in which the reactions occur.

The second pass, downward through the debris pool, is similar; it follows any material entering from above (from another cavity or from a TP), and any sinking reaction and/or concrete ablation products. If the mechanistic mixing model is used, mixing calculations are done during the downward pass. This differs from the initially released version of CORCON-Mod3 [2]. The change was made because separation of a mixed layer can create a new pure-phase layer *below* it, and the revised order of calculations greatly simplifies the logic in treating this possibility.

3.7 Water Ingress and Melt Eruption Models

3.7.1 Description

Two new models have been added to CAV as a result of observations in the OECD MACE/MCCI experiments.[19, 20] These are water ingression into the top crust, and the possibility of melt eruptions through the top crust into the water, forming an overlying debris layer.

The water ingression model is based on a model by Epstein[21] following work on water ingress into molten lava by Lister.[22] The basis of the model is the observation that water can progress into a crust via a cracking mechanism for a long ways, effectively limiting the possible thickness of the conduction zone in the crust.

The melt eruption model is based on the observation that under some circumstances, melt can erupt through the top crust into the overlying water layer, forming a debris bed on top of the crust.

3.7.2 General Implementation

The new models are implemented in CAV by adding two new layers to the CORCON model, a crust layer and a debris layer. These layers sit on top of the existing melt layers in CORCON. Mass is transferred from the melt to the crust layer by a dynamic crust model, replacing the present static top crust model in the melt layer. Mass can be transferred to the debris layer through the crust from the melt layer via a melt ejection model as detailed below. Equations for the new models generally follow those in CORQUENCH (CQ).[23]

3.7.3 Water Ingress Model

The water ingression model allows water into the crust layer if the top heat flux is less than a dryout flux q''_{dry}. The test is applied at the top of the conduction zone in the crust layer. The dryout flux is given as

$$q''_{dry} = C_{dry}\left(\left(h_{lv}\frac{(\rho_l-\rho_v)g}{\nu_v}\right)^5\left(\frac{N_{dry}k_c^2\Delta e_{sat}^2}{C_p\Delta e_{cr}}\right)^4\right)^{1/13}\left(\alpha_T(T_{cr}-T_{sat})\right)^{15/13} \quad (3\text{-}8)$$

where

q''_{dry} = dryout heat flux (W/m^2)

C_{dry} = dimensionless empirical constant

h_{lv} = heat of vaporization of water (J/kg)

ρ_l = density of water (kg/m^3) (960 kg/m^3)

ρ_v = density of steam (kg/m^3) (0.59 kg/m^3)

g = gravitational constant = 9.8 m/s^2

ν_v = dynamic viscosity of steam (m^2/s) (1.29e-5 m^2/s)

N_{dry} = numerical constant = 0.1 K-m$^{1/2}$

k_c = thermal conductivity of crust (W/m/K)

Δe_{sat} = change in specific enthalpy from melt temperature to saturation temperature (J/kg)

C_p = specific heat capacity of melt (J/kg-K)

Δe_{cr} = change in specific enthalpy from crack temperature to saturation temperature (J/kg)

α_T = coefficient of thermal expansion for melt (1/K)

T_{cr} = crack temperature (K)

T_{sat} = saturation temperature of steam (K)

The crack temperature is estimated from Lister's formula as

$$T_{cr} = T_s - \frac{\sigma_{tens}}{\alpha_T E} \tag{3-9}$$

where

σ_{tens} = tensile strength of crust (Pa) (6.77e7 Pa)

E = Young's modulus for crust (Pa) (1.25e11 Pa)

Some of these quantities are not calculated in CORCON so are set to constant values; the values are taken from a CQ run of CCI-3.

The thickness of the conduction region is estimated as

$$\delta_{cond} = k_c \frac{T_B - T_{sat}}{q''_{dry}} \tag{3-10}$$

The actual solution given by Epstein is

$$\delta_{cond} = \frac{\alpha\rho\left(h_{fs} + C_p(T_S - T_{sat})\right)}{q''_{dry} - q''_B} ln \frac{h_{fs} + C_p(T_S - T_{sat})}{h_{fs} + (q''_B/q''_{dry})C_p(T_S - T_{sat})} \quad (3\text{-}11)$$

3.7.4 Melt Eruption Model

Melt eruption is implemented in CAV as a transfer of mass from the melt layer to the debris layer. The rate of transfer is proportional to the gas sparging rate:

$$j_{melt} = K_{ent} j_{gas} \quad (3\text{-}12)$$

where

j_{melt} = melt ejection rate (m/s)

K_{ent} = entrainment coefficient

j_{gas} = gas sparging rate (m/s)

The entrainment coefficient is calculated using the Ricou-Spalding correlation [24] as

$$K_{ent} = E_{ent}\left(\frac{\rho_{gas}}{\rho_{melt}}\right)^{1/2} \quad (3\text{-}13)$$

where

E_{ent} = user input entrainment constant (default = 0.06)

ρ_{gas} = gas density (kg/m^3)

ρ_{melt} = melt density (kg/m^3)

There is also a condition whether or not melt is ejected, based on a minimum gas flow rate and the crust permeability, given as

$$j_{min} \frac{\kappa(\rho_c - \rho_m)g}{\mu_g} \quad (3\text{-}14)$$

where

j_{min} = minimum gas flow rate (m/s)

κ = crust permeability (m^2)

ρ_c = crust density (kg/m^3)

μ_g = gas viscosity (Pa-s)

This equation is for a crust that is assumed to be floating on the melt, hence the difference term with the crust and melt densities.

Permeability is calculated based on the dryout flux above, q''_{dry}, using an expression from Jones et al. [25]:

$$\kappa = \frac{2\mu_v q''_{dry}}{\rho_v h_{lv}(\rho_l - \rho_v)g} \tag{3-15}$$

3.8 Debris Spreading

A new melt spread model for debris in the cavity has been added to MELCOR 2.2 and is the new default model treatment. Before the new debris spread, debris would spread uniformly and instantaneously across the full width of any cavity into which it is deposited. Optionally, users may provide the radius of the debris bed specified by a tabular function, a control function, or a channel in an external data file. In most MELCOR calculations, debris does not appear in the cavity until after the reactor vessel fails, and the time of this event is not known in advance. Additionally, defining debris spreading requires the user to have some knowledge of the melt progression and melt properties in order to input the spreading information. The new default internal model in MELCOR allows a formalized treatment of debris spreading using the internal MELCOR melt properties, thus modeling a more realistic spreading of the debris.

Ramacciotti Model for Two-Phase Viscosity

One advantage to using the internal model is the use of internal models for molten debris viscosity as a function of the melt solid fraction. There are two models for calculating the enhancement of viscosity due to solid/liquid phases, the Ramacciotti model and the Kunitz model. Recent advancements in modeling debris spreading [26-27] suggest the use of the Ramacciotti model [28]:

$$\mu = \mu_0 \cdot \exp\left(2.5 \cdot C \cdot \phi\right) \tag{3-16}$$

where C ranges from 4 to 8 depending on the experiment simulated, μ_0 is given the following melt metal and melt oxide layers of the debris, and ϕ is the melt solid fraction.

For melt metal,

$$\mu_0 = 1.076 \times 10^{-3} \exp^{3313/T} \cdot M_{user} \tag{3-17}$$

Where T is the temperature of the melt layer, and M_{user} is the user supplied multiplier. Note that the correlation implemented in the code is representative of steel.

For oxide, μ_0 is calculated with a complicated function as defined using Kendell-Monroe or Shaw correlations. A maximum value of the two correlations is used in MELCOR. However, if the viscosity calculated is not greater than zero, the use of Basalt viscosity formulation is used:

$$\mu_0 = 1.94 \times 10^{-5} \exp^{20950/T} \cdot M_{user} \tag{3-18}$$

Where T is the temperature of melt layer, and M_{user} is the user supplied multiplier.

Kunitz Model for Two-Phase Viscosity

CORCON-MOD3 [1] incorporates the Kunitz model for enhanced viscosity due to solid suspended in a molten slurry. The predicted viscosity enhancement is a function of the solid volume fraction and is 1 for when the solid fraction is insignificant and becomes infinite as the solid fraction approaches 1.0

$$\mu = \mu_0 \cdot \frac{1 + 0.5 \cdot \phi}{(1 - \phi)^4} \tag{3-19}$$

Also note that ϕ is computed based on the temperatures:

$$\phi = \mathrm{Max}\left(0, \mathrm{MIN}\left(1, \left(\frac{T_{liq} - T_{ave}}{T_{liq} - T_{sol}}\right)\right)\right) \tag{3-20}$$

where T_{liq} is the liquidus temperature, T_{sol} is the solidus temperature and T_{ave} is the average liquid temperature, which is given by:

$$T_{ave} = \frac{T_Z + T_R}{2} \tag{3-21}$$

Where T_Z is the Z average liquid temperature and T_R is the R average liquid temperature. Z and R are the measures of the thickness and radius of the melt, respectively. Note that maximum value of ϕ is set in the code as 0.9. Figure 3.3 plots both the Kunitz and Ramacciotti viscosity multipliers as functions of ϕ. For the Ramacciotti correlation, several values of C are plotted in this figure.

The analytical melt spread model assumes a right circular cylinder of debris with radius, R, height, H, volume $V=\pi R^2 H$, density, ρ, and viscosity, μ. Assuming that the debris spreading process is driven by gravitational forces and opposed by viscous forces, and that the flow is laminar, the balance between forces can be expressed in terms of pressures by

$$\mu \frac{v}{H^2} R \propto \rho g H \tag{3-22}$$

where g and v (dR/dt) are the gravity and characteristic velocity of the flow, respectively.

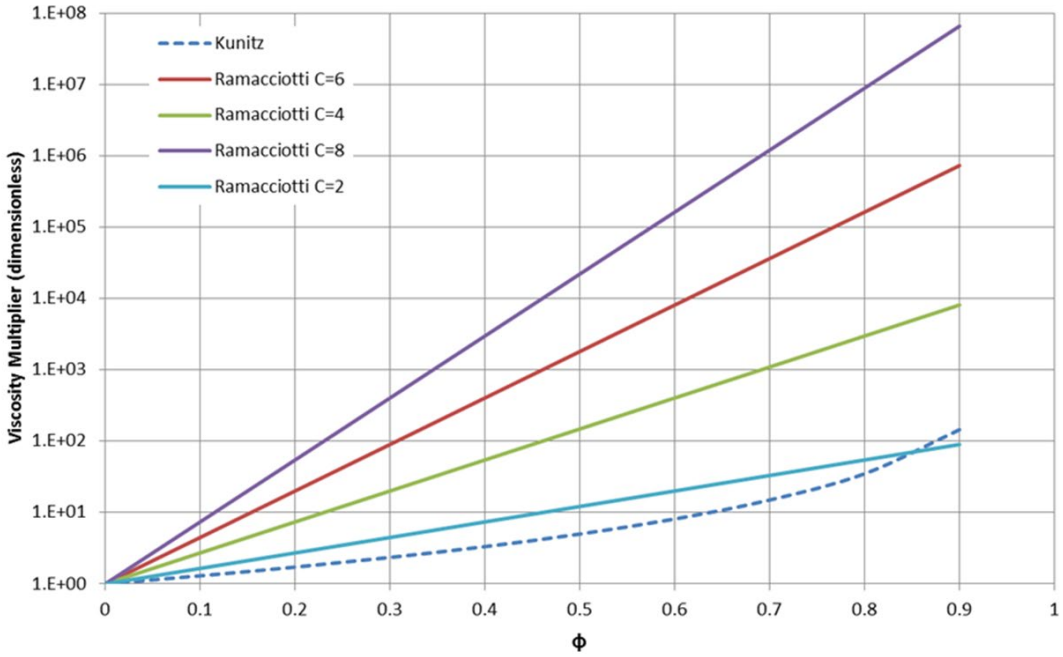


Figure 3.3 Viscosity Multiplier Correlations as a Function of ϕ

Equation (3-22) can be rewritten as

$$\frac{dR}{dt} \propto \frac{\rho\, g}{\mu} \frac{H^3}{R} \tag{3-23}$$

Assuming that the volume of debris is approximately constant Equation (3-23) can be rewritten to eliminate the height, H, by expressing it in terms of the volume, V:

$$\frac{dR}{dt} = C_1 \frac{\rho\, g}{\mu} \left(\frac{V}{\pi R^2}\right)^3 \frac{1}{R} \quad \text{or} \quad C_1 \frac{\rho\, g}{\mu \pi^3} \frac{V^3}{R^7} \tag{3-24}$$

where C_1 is the constant of proportionality. The solution for Equation (3-24) is given as:

$$R(t) = \sqrt[8]{R(t_0)^8 + C_1 \cdot \frac{\rho\, g}{\mu\, \pi^3} V^3 (t - t_0)} \tag{3-25}$$

Huppert [29] has suggested a value for C_1 = 0.136.

Spreading is computed only when liquid remains in the layers. Therefore, it is necessary to sum up the liquid portion of each layer in the debris.

The volume of each layer, including both liquid and solid is computed as

$$V_{lay} = \frac{m_{lay}}{\rho_{lay}} \quad (3\text{-}26)$$

where $V, m \text{ and } \rho$ are the volume, mass and density of the layer, respectively. Once the volume is calculated, the layer height, H_{lay} is given as

$$H_{lay} = V_{lay}/(\pi \cdot R^2) \quad (3\text{-}27)$$

where R is calculated previously for the debris radius. The liquid height for each layer, $H_{lay,liq}$, is computed as

$$H_{lay,liq} = H_{lay} - \delta_{lay,bot} - \delta_{lay,top} \quad (3\text{-}28)$$

where δ is the crust for the bottom and top of the layer. Similarly, the liquid radius for each layer, $R_{lay,liq}$ is calculated as

$$R_{lay,liq} = R - \delta_{lay,rad} \quad (3\text{-}29)$$

where δ is the radial crust. Based on both the radius and height of liquid in each layer, the volume of the liquid in each layer is calculated, assuming it is a right cylinder. Once this volume is computed, the liquid volume fraction is computed:

$$F_{lay,liq} = V_{lay,liq}/V_{lay} \quad (3\text{-}30)$$

The total liquid volume is computed by dividing the average density by the total liquid mass of the debris. Average properties, such as viscosity and density are then calculated. The total liquid volume, and the average viscosity and density are then used in Equation (3-24) to estimate the spreading during a time step. Equation (3-24) is then integrated using the numerical Euler integral method: $y_{n+1}=y_n+y'_n \cdot \Delta t$.

For now, a stopping logic for melt spread is used. The user may choose to change these thresholds. This selection is done through the use of SC2303(2). When SC2303(2)=0, the simple MELCOR immobilization logic is used – 0.5 of radius as crust or 0.5 of the melt thickness as crust (default). These two values are accessible in SC2303(3) and (4), respectively.

3.9 Energy Conservation

CORCON uses a formulation for the energy equation for debris in the cavity in which temperature-driven heat transfers between layers are treated semi-implicitly, as described in Reference 1. Numerical difficulties associated with addition of new debris were observed during incorporation into MELCOR. The implementation of the equation was substantially modified to improve stability by including modifications made to previous versions of CORCON in the CAV package of MELCOR 1.8.2. The revised numerical treatment is now included in the stand-alone code as well as in MELCOR.

3.10 Material Properties

The material properties in the CAV package are those of the stand-alone CORCON code. They include internally consistent specific heats, enthalpies, and chemical potentials for a large number of condensed and gaseous species, based on fits to JANAF [30] and other data. All enthalpies are based on the JANAF thermochemical reference point. All heats of reaction are therefore implicitly contained in the enthalpy data. Also included are data on thermal expansivity and density, thermal conductivity, viscosity, and surface tension.

The list of materials for which properties are defined is contained in Appendix A. These data are independent of the MELCOR data contained in the Water (H_2O), NonCondensible Gas (NCG), and Material Properties (MP) packages. They are retained both for consistency with the stand-alone CORCON code and to facilitate incorporating future upgrades to CORCON modeling into MELCOR. Appropriate adjustments to enthalpies are made whenever materials are passed into or out of the Cavity package.

Additional models are included for evaluating the properties of mixtures. Details of the material properties models, and further references, are contained in Reference 1. Most are quite conventional, but two deserve further discussion in this Reference Manual.

In determining the enthalpy of a mixture as a function of temperature, a submodel is used to determine its melting range as defined by solidus and liquidus temperatures. Below the solidus temperature of the mixture, properties for the solid phase of each species—extrapolated, if necessary—are used. Similarly, liquid phase properties (possibly extrapolated) are used above the liquidus temperature. Between solidus and liquidus, the enthalpy is interpolated as a linear function of temperature (corresponding to a constant specific heat).

The melting range for the metallic phase is determined from a fit to the ternary phase diagram for Cr-Fe-Ni; other elements (Zr, C) are simply ignored. If the metal phase contains no Cr, Fe, or Ni, however, the melting point of Zr is used. The melt range for an oxidic phase is determined by reference to a pseudo-binary phase diagram based on an ideal solution model for the liquid and solid phases. One component is high melting and is assumed to consist of fuel (UO_2 and ZrO_2); the second component is low melting and includes everything else. The corresponding melting temperatures and effective latent heats are taken from internal data for fuel for the first component and from the properties of the concrete oxides for the second.

Also modeled is the effect of SiO_2 content on the viscosity of oxidic mixtures, based on a modification of a correlation derived by Shaw [31]. The original correlation was fit to a database containing geologic data for materials with relatively high silica contents; no consideration was given to application of the correlation outside of the range of compositions included in the original database. As implemented in CORCON and in MELCOR, the original correlation has been modified to avoid nonphysical extrapolation

characteristics. It is coupled to a conventional Kendell-Monroe [32] mixture model in such a way that the viscosity is a continuous function of composition over an unrestricted range of compositions. Details are given in Reference 33.

4. Comparison to Stand-Alone CORCON

The Cavity (CAV) package in MELCOR consists primarily of the CORCON-Mod3 code [1]. The calculational routines are identical to those in the stand-alone code, but input, output, and interfaces to boundary conditions are different. In addition, the MELCOR implementation includes several sensitivity coefficients to allow user control of submodels in CORCON. The sensitivity coefficients currently available are:

1. an additive modification to the concrete ablation enthalpy, and

2. coefficients in many heat transfer relations.

In future versions, we expect to expand this list to allow access to more of the so-called "user flexibility" options available in CORCON-Mod3.

Additional similarities and differences with respect to CORCON-Mod 3 are summarized in Table 4.1.

Table 4.1 Comparison of Stand-Alone CORCON-Mod3 [1] and MELCOR Cavity Package

Feature	CORCON-Mod3	MELCOR
Concrete Cavity, Layered Debris, Debris/Concrete Heat Transfer, Concrete Ablation	Treatment identical	
Ablation Delay	Not permitted	Optional control function
Overlying Water	Simple equilibration of rising gases	Part of CVH SPARC bubble model
Atmosphere and Surroundings	User-input tabular boundary conditions	Boundary conditions from CVH Package
Debris/Water or Debris/Atmosphere Interface	Models and correlations identical; numerics of solution modified for MELCOR	

Table 4.1 Comparison of Stand-Alone CORCON-Mod3 [1] and MELCOR Cavity Package

Feature	CORCON-Mod3	MELCOR
Fission Product (F.P.) Inventories	Six “pseudo-species” (coarse grouping) included in CORCON; separate detailed inventory for VANESA	Treated in detail by RN package (not part of CAV package inventory)
Internal Heating	Internal model based on F.P. inventories or input table	From DCH package, based on fission product inventories or input table
Fission Product Release	Models and correlations identical; numerics of solution modified by location of model in RN package in MELCOR	
Debris/Gas Chemistry	General equilibrium gases, metals, oxides	Same model, minor gas species suppressed
Cavity Rupture/ Debris Overflow	Not modeled	Mechanistic melt-through or “triggered” failure; overflow to lower cavity
Debris Addition	User-input table	Through TP package, from other MELCOR package or table input, or from other cavity overflow or rupture
Debris Spreading	Parametric model; requires user-input table vs. time	Same model, but allows calculation using control functions
Associated F.P. Addition	Based on added UO_2, or user-input table	Calculated by RN for package providing debris source, or table input
User Control of Modeling	Provided through “user flexibility” options [2]	Provided by user input and sensitivity coefficients; not all “user flexibility” options are enabled
Restart/Fallback Capability	Not available	Provided as part of MELCOR structure

Table 4.1 Comparison of Stand-Alone CORCON-Mod3 [1] and MELCOR Cavity Package

Feature	CORCON-Mod3	MELCOR
User Input	Fixed format	MELCOR free-field format
Printed Output	Controlled by CORCON input	Essentially identical; controlled by MELCOR input
Plotted Output	Latest version allows use of HISPLTM	Plots available in normal MELCOR manner

The differences between the MELCOR Cavity package and stand-alone CORCON-Mod3 listed in Table 4.1 fall into three distinct groups:

1. Coupling of Phenomena

These differences include the use of calculated boundary conditions such as temperature, pressure, and debris addition rates rather than user-supplied tabular data generated from some independent source, and the provision to allow debris to be relocated between two or more locations when cavity boundaries fail. CAV allows the use of tabular boundary conditions by defining time-specified volumes in CVH and/or tabular debris addition rates through TP and EDF. Both CORCON-Mod3 and MELCOR can calculate internal decay heating based on fission product inventories, with these inventories based on fission product release rates calculated using VANESA. In MELCOR, the decay heat is based directly on the detailed inventories calculated by VANESA; in CORCON-Mod3, these inventories must be approximately mapped back into the coarse group inventories used by the CORCON decay heat model.

2. User Interface, User Convenience

These differences include revised input formats, restart and fallback capabilities, and plot capabilities, which have no effect on modeling of physical phenomena.

Although the CAV package in MELCOR and the stand-alone CORCON-Mod3 code contain identical versions of all subroutines incorporating phenomenological models and materials properties, they should be viewed as distinct entities because of the differences in treatment of interfaces and calculation of boundary conditions. However, because the basic modeling is identical, it is possible to run equivalent calculations with the two codes under appropriate choices of options and restrictions on boundary conditions.

Appendix A: Species List for CORCON in MELCOR CAV Package

The following lists the species considered by CORCON and available for use in MELCOR, either as initial contents in the melt or as constituents of concrete (see Table in the next page):

OXIDES	METALS
SIO2	FE
TIO2	CR
FEO	NI
MNO	ZR
MGO	MN
CAO	C(C)
SRO	NA
BAO	AL
LI2O	U
NA2O	SI
K2O	UAL3
FE2O3	UAL2
AL2O3	CA
UO2	
ZRO2	
CR2O3	
NIO	
FE3O4	
MN3O4	
PUO2	
UO3	
U3O8	

CONCRETE CONSTITUENTS
CO2
H2OCHEM (chemically bound water)
H2OEVAP (evaporative water)
CACO3
CA(OH)2

The observant reader may note that several additional species are included in the corresponding list, Table 2.1, in [1]. These include the aluminates, fission products, and element “X”, which are (or were) used in internal models in stand-alone versions of CORCON and are not relevant to the implementation in MELCOR.

Note that in the concrete table there are five additional species that may be used to specify concrete compositions:

These are used only in specification of the concrete composition; in particular, CACO3 and CA(OH)2 are decomposed during initialization into CAO plus CO2 and into CAO plus H2OCHEM respectively. The difference between H2OCHEM and H2OEVAP is the binding energy that must be overcome to release the chemically-bound water from the concrete.

The list of gases in Table 2.1 of Reference 1 is not relevant to MELCOR input, as the composition of the control volume above the debris pool is determined by the CVH package. In addition, production of all trace gaseous species has been suppressed in the chemical reaction routines so that the only gases considered by CORCON in MELCOR are H2, H2O, CO, and CO2.

References

1. D. R. Bradley and D. R. Gardner, CORCON-MOD3: An Integrated Computer Model for Analysis of Molten Core-Concrete Interactions. Users Manual, NUREG/CR-5843, SAND92-0167, Sandia National Laboratories, Albuquerque, NM (October 1993).

2. K. E. Washington, Letter Report to Sudhamay Basu, USNRC, "CORCON-Mod3 Code Integration," under FIN L1484 (February 1993).

3. K. K. Murata et al., Code Manual for CONTAIN 2.0: A Computer Code for Nuclear Reactor Containment Analysis, NUREG/CR-6533, SAND97-1735, Sandia National Laboratories, Albuquerque, NM (December 1997).

4. D. A. Powers, J. E. Brockmann, and A. W. Shiver, VANESA: A Mechanistic Model of Radionuclide Release and Aerosol Generation During Core Debris Interactions with Concrete, NUREG/CR-4308, SAND85-1370, Sandia National Laboratories, Albuquerque, NM (July 1986).

5. MAAP, Modular Accident Analysis Program User's Manual, Volumes 1 and 2, IDCOR (undated).

6. G. A. Greene, "Heat, Mass, and Momentum Transfer in a Multi-Fluid Bubbling Pool," Advances in Heat Transfer, Vol. 21, pp. 270-345 (1991).

7. G. A. Greene, J. C. Chen, and M. T. Conklin, "Onset of Entrainment Between Immiscible Liquid Layers Due to Rising Gas Bubbles," International Journal of Heat and Mass Transfer, Vol. 31, p. 1309 (1988).

8. G. A. Greene, J. C. Chen, and M. T. Conklin, "Bubble-Induced Entrainment Between Stratified Liquid Layers," International Journal of Heat and Mass Transfer, Vol. 34, p. 149 (1990).

9. S. S. Kutateladze and I. G. Malenkov, "Boiling and Bubbling Heat Transfer Between a Gas-Liquid System and a Heat Exchange Element," Zhurnal Prikladnoi Khimii (Journal of Applied Chemistry of the USSR), Vol. 35, No. 11 (1962).

10. G. A. Greene and T. F. Irvine, "Heat Transfer Between Stratified Immiscible Liquid Layers Driven by Gas Bubbling Across the Interface," in ANS Proceedings of the 1988 National Heat Transfer Conference, Houston, TX, July 24-27 (1988).

11. F. A. Kulacki and R. J. Goldstein, "Thermal Convection in a Horizontal Fluid Layer with Uniform Volumetric Energy Source, "Journal of Fluid Mechanics, Vol. 55, No. 2, p. 271 (1975).

12. F. A. Kulacki and M. E. Nagle, Journal of Heat Transfer, Vol. 97, p. 204 (1975).

13. F. A. Kulacki and A. A. Emara, Trans. ANS, Vol. 22, No. 2, p. 447 (1975).

14. M. T. Farmer, J. J. Sienicki, and B. W. Spencer, "CORQUENCH: A Model for Gas Sparging-Enhanced Melt Water Film Boiling Heat Transfer," ANS Special Session on Thermal Hydraulics of Severe Accidents, November 11-15, 1990.

15. R. K. Cole, Jr., "A Crust Formation and Refreezing Model for Molten-Fuel/Concrete Interaction Codes," Paper 12.5 in Proceedings, International Meeting on Light Water Reactor Severe Accident Evaluation, Cambridge, MA (1983).

16. H. Alsmeyer and M. Reimann, "On the Heat and Mass Transport Processes of Horizontal Melting or Decomposing Layer under a Molten Pool," Nuclear Reactor Safety Heat Transfer, Winter Annual Meeting ASME, Atlanta, GA, pp. 47-53 (1977).

17. D. R. Bradley, "Modeling of Heat Transfer Between Core Debris and Concrete," in ANS Proceedings of the 1988 National Heat Transfer Conference, Houston, TX, p. 37-49 (July 1988).

18. K. C. Kwong, R. A. S. Beck, and T. C. Derbidge, CORCON Program Assistance, FR-79-10/AS, ACUREX Corporation/Aerotherm Aerospace Division, Mt. View, CA (July 1979).

19. M. T. Farmer et al., "Status and Future Direction of the Melt Attack and Coolability Experiments (MACE) Program at Argonne National Laboratory," Proceedings 9th Int. Conf. on Nucl. Eng., Nice, France, April 8-12, (2001).

20. M. T. Farmer et al., "A Summary of Findings from the Melt Coolability and Concrete Interaction (MCCI) Program," *Proceedings ICAPP '07*, Nice, France, May 13-18, (2007).

21. M. Epstein, "Dryout Heat Flux During Penetration of Water Into Solidifying Rock," J. Heat Trans. **128**, p. 847 (2006).

22. C.R. Lister, "On the Penetration of Water Into Hot Rock", *Geophys. J.R. Astron. Soc*. **39**, p.465 (1974).

23. M.T. Farmer, "OECD MCCI Project, The CORQUENCH Code for Modeling of Ex-Vessel Corium Coolability Under Top Flooding Conditions, Code Manual Version 3.03", OECD/MCCI-2010-TR03, Argonne National Laboratory, Argonne, IL (2011).

24. F. B. Ricou and D. B. Spalding, "Measurements of Entrainment of Axisymmetrical Turbulent Jets," J. Fluid Mechanics, **11,** 21 (1961).

25. S.W. Jones et al, “Dryout Heat Fluxes in Particulate Beds Heated Through the Base,” *J. Heat Trans*. **106**, 176 (1984).

26 K.R. Robb and M.W. Francis, Enhanced Ex-Vessel Analysis for Fukushima Daiichi Unit1: Melt Spreading and Core-Concrete Interaction Analyses with MELTSPREAD and CORQUENCH, ORNL/TM-2012/455, Oak Ridge National Laboratory, February 2013.

27 M.T. Farmer, Melt Spreading Code Assessment, Modification, and Applications to the EPR Core Catcher Design, ANL-09/10, Argonne National Laboratory, March 2009.

28 M. Ramacciotti, et.al, “Viscosity Models for Corium Melts, Nuclear Engineering and Design 204, 377-389, 2001.

29 H.E. Huppert, “The Propagation of Two-Dimensional and Axisymmetric Viscous Gravity Currents Over a Rigid Horizontal Surface,” J. Fluid Mech. Volume 121, page 43-58, 1982.

30. JANAF Thermochemical Tables, DOW Chemical Company, Thermal Research Laboratory, Midland, MI (1965).

31. H. R. Shaw, “Viscosities of Magmatic Silicate Liquids: An Empirical Method of Prediction,” American Journal of Science, Vol. 272, pp. 870-893 (1972).

32. J. Kendell and K. P. Monroe, “The Viscosity of Liquids, III Ideal Solution of Solids in Liquids,” Journal of the American Chem. Soc., Vol. 39, No. 8, p. 1802 (September 1917).

33. R. K. Cole, Jr., D. P. Kelly, and M. A. Ellis, “Molten Fuel/Concrete Interactions Study,” in Light Water Reactor Safety Research Program Semiannual Report, April-September 1982, NUREG/CR-3407, SAND83-1576, Nuclear Fuel Cycle Program, Sandia National Laboratories, Albuquerque, NM (October 1983).

Condenser (CND) Package

The purpose of the MELCOR CND Package is to model the effects of the Isolation Condenser System (ICS) and the Passive Containment Cooling System (PCCS), both of which use heat exchangers submerged in large water pools. Several older boiling water reactors (BWRs) and the new proposed simplified boiling water reactor (SBWR) contain isolation condensers to condense steam created in the core and return it to the primary system. Only the simplified boiling water reactor, however, contains the passive containment cooling system to provide steam suppression in the drywell in the event of a LOCA or when the depressurization valves are used to equalize the pressures of the reactor vessel and containment. This equalization is required so that water can drain to the reactor vessel from the gravity-driven cooling system pools located several meters above the top of the core. The CND Package constitutes a subpackage within the ESF Package. The removal or transport of fission product vapors and aerosols is not modeled. The Reference Manual gives a description of the subroutines used in the CND Package.

User input for running MELGEN and MELCOR with the CND Package activated is described separately in the Condenser Package Users' Guide.

Contents

List of Figures

List of Tables

Figure 2.1 The noncondensible gas mole fraction decreases rapidly when steam is released directly into the drywell atmosphere during the final stage of an SBWR reactor vessel depressurization..... 19
Figure 2.2 The drywell-to-wetwell pressure differential **delpre** and the differential pressure **reqpre** at which flow through the PCCS vent line is initiated. 20
Figure 2.3 The available PCCS heat exchanger capacity (dots) is primarily determined by the drywell noncondensible gas mole fraction (solid line)..... 23
Figure 2.4 The available PCCS (three-unit) heat exchanger capacity **pitcet** and the power **e2adic** actually utilized..... 24
Figure 2.5 The total mass flow **pltifl** through the PCCS vent line and the associated flow **pltnfl** of noncondensible gases. 25

1. Introduction

This Package describes the Passive Containment Cooling System (PCCS) and Isolation Condenser System (ICS) models originally developed at Oak Ridge National Laboratory (ORNL) for use with MELCOR. This manual is divided into three sections. Section 2 describes the PCCS model, while Section 3 describes the extension of the basic PCCS model to provide calculational capability for the ICS. Finally, the interface with MELCOR for both the PCCS and ICS models is described in Section 4.

2. PCCS Model

2.1 Introduction and Concept

The PCCS is a safety-related passive system designed to remove the core decay heat that would be introduced into the SBWR containment during a loss-of-coolant accident (LOCA). The PCCS is described in Section 6.2 of the *SBWR Standard Safety Analysis Report (SSAR)* [1].

The basic operation of the PCCS derives from the induced flow of some of the drywell atmosphere to the wetwell airspace via the PCCS whenever the drywell-to-wetwell pressure differential is sufficient to clear the water from the vent line terminus within the pressure suppression pool. The venting pathway through the PCCS includes a heat exchanger in which the gases are cooled and some (or all) of the steam vapor is condensed; the condensate is drained to the Gravity-Driven Cooling System (GDCS) pool within the drywell. The noncondensible gases and any steam carryover through the vent line are released into the pressure suppression pool, where the gas bubbles rise to the pool surface. The intermittent nature of the venting process causes the thermal-hydraulic behavior of the PCCS to be much more complex than the normally encountered heat exchanger-condenser applications for which the flow is continuous.

The PCCS model described here is based upon the concept that the MELCOR code should adequately represent the effects of the PCCS under the boundary conditions that would be imposed by accidents. It is not intended that the MELCOR calculation should attempt to predict the performance of these heat exchanger-condenser systems based upon basic physical considerations; this is done by more sophisticated thermal-hydraulic codes. Furthermore, test calculations performed with MELCOR demonstrate that attempts to use the basic code “building block” approach to connect control volumes, flow paths, and heat sink structures as necessary to directly simulate the PCCS heat exchanger-condensers result in code difficulties; these include oscillations in the predicted flows and energy exchanges, a demand for extremely small timesteps, and impractically large CPU and wall clock time consumption.

2.2 General PCCS Performance

Based upon the available information in the literature concerning the PCCS design and the results of equipment tests reported by the development consortium to date, it is clear that any PCCS component model must have the following basic attributes:

(1) Capacity limited to gravity drainage of steam condensing in the tubes until drywell pressure exceeds suppression chamber pressure by a margin [about 7.25 kPa (1.05 psid)] sufficient to overcome PCCS vent line submergence. With normal pressure suppression pool water level, the uppermost vent line exit hole lies at the depth of 0.75 m (2.5 ft). The pool water level may vary during the course of an accident and this must be considered in the model.

(2) For long-term cooling situations of practical interest for BWR accident calculations, the drywell-to-suppression chamber pressure differential is limited to the submergence of the drywell-to-pressure suppression pool vents.

(3) Capacity increases as the drywell-to-suppression pool pressure differential (vent line flow) increases over the small range between PCCS vent line clearance and clearance of the main horizontal vents.

(4) Capacity decreases with increasing partial pressure of noncondensible gases in the upper drywell because of the interference of the gas boundary layer within the PCCS tubes with the steam-to-wall heat transfer.

(5) Whenever the wetwell pressure approaches (or exceeds) the drywell pressure so that vent line flow is zero, the PCCS heat exchanger-condenser is subject to filling with noncondensible gases as the condensing steam is continuously replaced with a mixture of steam and noncondensible gas from the drywell. The PCCS is said to be “bound” when it contains only cool noncondensible gas so that all heat exchange and condensing operation is terminated.

(6) The average PCCS capacity over the long term is determined by the heat transfer from the outer surface of the PCCS heat exchanger tubes to the surrounding ICS/PCC pool. For the LOCA analysis presented in Section 6.2 of the SSAR, the General Electric Company has employed a constant heat transfer coefficient of 4500 W/(m^2-K) [792.5 Btu/(h-ft^2-F)] for the tube outer surface area.

(7) Capacity of the PCCS decreases as the pressure in the drywell falls below its optimum operational pressure. As the pressure drops in the drywell, the temperature of the steam and associated condensate drops, thereby lowering the heat transfer between the condenser wall and the steam. Heat transfer is determined by the heat transfer coefficient times the surface area times the difference between the steam temperature and the temperature of the condenser wall, which is very close to the surrounding pool temperature.

A general model interacting with MELCOR has been constructed from the available information and tested satisfactorily. Nevertheless, the most recent detailed information concerning experimental measurements or the results of sophisticated calculations of PCCS performance as a function of the ICS/PCC pool temperature, the drywell-to-wetwell atmosphere pressure differential, atmospheric pressure in the drywell, and the noncondensible gas fraction in the drywell atmosphere should be used to refine the input for this model (described in the CND Package Users' Guide) whenever production calculations are performed.

2.3 Operation of the PCCS Model

The PCCS model is contained within MELCOR Subroutine CNDRN1. In this section, the operation of the model is described as a 28-step process. Not all steps are executed each calculational timestep. One of the steps involves an iterative procedure, which is described in detail in Section 2.4. Those readers not interested in pursuing the level of understanding offered by a detailed discussion of model operation are encouraged to skip to Section 2.5, which provides an overview in the form of an example of calculated results.

It is important to recognize that the PCCS model operates on the assumption that the pressure within the PCCS remains equal to the drywell pressure and constant during a calculational timestep. Whenever material is removed, for example, when steam condenses and the condensate is transferred to the GDCS, a void is considered to be created within the PCCS. An uptake of mixture from the drywell atmosphere is required to fill this void at drywell pressure and the subsequent equilibrium conditions within the PCCS are calculated. This approach is taken to avoid the penalties (described in Section 2.1) of a mechanistic model for which mass transfers between the drywell and the relatively small PCCS would be based upon calculated pressure differentials.

The variable names mentioned in the following discussions and in Section 2.4 are the same as those used within Subroutine CNDRN1. The interested reader is encouraged to compare the stepwise operations described here with the actual FORTRAN in a listing of Subroutine CNDRN1; the COMMENT statements that are obtained with the program listing provide additional detailed information.

Before beginning the step-by-step discussion of model operation, it is necessary to define a few of the variable names that are encountered (the meaning of the others is obvious from the text).

NUMMAT Is the total number of materials considered present (or potentially present) within a control volume. These include the water pool, fog droplets, steam, and the noncondensible gases.

I I is the index of a particular material within a control volume.

Index	Material
1	water pool
2	fog
3	vapor
4 through NUMMAT	noncondensible gas

The control volume atmosphere is comprised of materials 2 through NUMMAT. The control volume total pressure is the sum of the partial pressures of materials 3 through NUMMAT.

CEFIC represents the running total kept within the model of the remaining PCCS heat exchanger capacity in Joules. The available capacity is established at the beginning of each timestep from tabular input supplied by the MELCOR user. This initial value depends upon the current ICS/PCC pool temperature, the current drywell-to-wetwell pressure differential, and the current mole fraction of noncondensible gas in the drywell atmosphere. It should be noted that the reduction in PCCS performance due to a buildup of noncondensible gas within the heat exchanger is not established from the tabular input, but rather is calculated by the PCCS model.

ENGIC(I) is the array containing the internal energies of the materials within the PCCS at the beginning of the timestep. During the timestep, the running values of these internal energies are contained in the array ETOTIC(I), which is copied to the ENGIC(I) array at the end of each timestep.

VLICMT is the volume of the materials (steam, fog, noncondensible gases) that constitute the atmosphere within the PCCS. Since the PCCS atmosphere is constrained to remain at a pressure equal to drywell pressure, this volume can be less than the actual PCCS structural volume if material is removed from the PCCS atmosphere during the calculation.

PCCS Model Steps

Steps 1 – 4: Establish Initial Conditions

These initial steps establish the equilibrium conditions within the PCCS with the volume filled at drywell pressure. Some of the available capacity is utilized to cool any noncondensible gas carried over from the previous timestep. Mixture is taken up from the drywell as required to maintain the PCCS at drywell pressure.

(1) Set the currently available heat removal capacity CEFIC based upon the drywell-to-wetwell pressure differential, the pressure in the drywell, and the noncondensible gas fraction in the drywell atmosphere. The dependence upon

the pressure differential and the source pressure are obtained from user-input tabular function IPCDPR and IPSRPR, respectively. The dependence upon the noncondensible gas mole fraction is obtained by interpolation between the user-input tabular functions IPLTMP (for 323.16 K) and IPCNCN (for 373.16 K), which correspond to ICS/PCC pool temperatures of 50 °C and 100 °C, respectively. (See Users' Guide for input record CND_PCCS02.)

(2) Cool any noncondensible gases remaining within the PCCS at the end of the previous timestep. The gas temperature is reduced to the ICS/PCC pool temperature TICPL by calling the routine NCGPRO to obtain the internal energy of the gases at the new temperature.

- Reduce the internal energies ENGIC(I) accordingly.
- Reduce the available capacity CEFIC.

(3) Take up enough mixture from the drywell atmosphere to make the calculated PCCS equilibrium pressure equal to the drywell pressure. (Section 2.4 provides a discussion of the iterative procedure used.)

- Reduce the drywell gas, vapor, and fog masses and energies accordingly.
- Output of the equilibration routine includes:

  ETOTIC(I) total internal energies and
  XMSICN(I) masses of the fog, vapor, and noncondensible gases.

- Set the PCCS material volume VLICMT equal to the internal volume of the PCCS structure.

(4) Determine if there is vent line flow this timestep.

- If No, continue with Steps 5 – 9.
- If Yes, continue with Steps 10 – 27.

Steps 5 – 9: No Vent Line Flow

The PCCS is now full at drywell pressure with its contents at an equilibrium temperature. If there was a void remaining at the end of the previous timestep, or if some cooling of the noncondensible gases occurred, then some steam (and fog) taken up with the mixture from the drywell atmosphere is included. CEFIC has already been reduced (Step 2) as necessary to account for the cooling of noncondensible gas.

(5) If no steam exists within the PCCS (No void at the end of the previous timestep and no noncondensible gas cooling or no steam in drywell atmosphere)

- Energy to ICS/PCC pool limited to that used to cool the noncondensible gases.

Go to Step 28.

(6) Condense the steam (and cool the fog) within the PCCS.

- May be limited because of insufficient capacity CEFIC remaining after the cooling of the noncondensible gas (Step 2).
- Add the masses and energies to the GDCS Pool.
- Reduce ETOTIC(I) and XMSICN(I) for steam and fog accordingly.
- Set RMVLIC equal to the accumulated void within the PCCS.
- Reduce the available capacity CEFIC accordingly.

(7) If CEFIC > 0.0 and RMVLIC > 0.0, take up enough mixture from the drywell atmosphere to use the available capacity and to partially fill the void (with noncondensible gas). On the other hand, it is possible that the noncondensible gas take-up completely fills the void without using all of the available capacity.

- The steam and fog taken up are never actually added to the PCCS volume within the model but rather are removed from the drywell atmosphere and added directly to the GDCS Pool as saturated liquid.
- Reduce the available capacity CEFIC by the amount of energy used in condensing the steam and cooling the fog.
- For the noncondensible gas take-up: Increase XMSICN(I) and ETOTIC(I) for these gases and remove the associated masses and energies from the drywell.
- Reduce the void RMVLIC according to the take-up of noncondensible gas (only)—note that RMVLIC remains greater than zero only if the take-up from the drywell atmosphere was limited by the available heat exchange and condensing capacity.

(8) Set VLICMT = VLICMT – RMVLIC. There is a void within the PCCS at the beginning of the next timestep if RMVLIC > 0.0 here.

(9) Add the energy used in cooling the noncondensible gases (Step 2) and in condensing the steam/cooling the fog (Steps 6 and 7) to the ICS/PCC pools.

Go to Step 28.

Steps 10 – 27: With Vent Line Flow

At this point, the PCCS is full at drywell pressure with its contents at an equilibrium temperature. If a void remained at the end of the previous timestep or if some cooling of the noncondensible gases occurred, then some steam (and fog) taken up with the mixture

from the drywell atmosphere is included. CEFIC has already been reduced (Step 2) to account for any cooling of the noncondensible gas.

(10) Calculate the PCCS vent line mass transfer XMS2FL. The transfer is based upon the pressure differential between the drywell and the vent line terminus, which is submerged in the pressure suppression pool.

(11) Move noncondensible gases from PCCS to wetwell and reduce the running total for XMS2FL accordingly.

- RMVLIC is the associated PCCS void.
- Reduce the values of
  XMSICN(I) masses and
  ETOTIC(I) internal energies
  for the noncondensible gases.

- At this point, either:
  XMS2FL=0.0; some noncondensible gas remains in PCCS
  or
  XMS2FL>0.0; all noncondensible gas has been removed so that only steam and fog remain within the PCCS.

(12) Condense the steam within the PCCS up to the limits of the available capacity CEFIC. Place the liquids in the GDCS pool.

- XMSREM is the mass of steam condensed.
- Reduce CEFIC accordingly.
- Reduce XMSICN(I) and ETOTIC(I) for the steam.

(13) If some steam remains in the PCCS and if some vent line mass transfer remains (XMS2FL > 0.0) then

- Move the steam (uncondensed) through the vent line to the pressure suppression pool.
- Reduce XMS2FL accordingly.
- Increase XMSREM so it now represents both the condensed steam drained to the GDCS and the uncondensed steam moved to the pressure suppression pool.
- Reduce XMSICN(I) and ETOTIC(I) for the steam.

(14) Increase RMVLIC to account for the void created by both the steam condensed and drained to the GDCS pool and the steam moved to the pressure suppression pool via the PCCS vent line.

Note: Steps 12 – 14 are actually performed (in sequence) for fog, steam, and any water pool that has formed within the PCCS volume. The handling of steam is demonstrated in this discussion; the fog and water pool (if it exists) are treated in a similar manner.

(15) Reduce the PCCS material volume VLICMT by subtracting the void RMVLIC.

Set VOLINT = 0.0
VINTNC = 0.0
XMNNST = 0.0

(16) If both the remaining heat exchanger capacity CEFIC and the remaining vent line mass transfer XMS2FL have been reduced to zero.

Go to Step 28.

Steps 17 – 18: Heat Removal Capacity/Vent Line Mass Transfer Imbalance
It is unlikely that the amount of mixture that must be taken up from the drywell in order to use the remaining heat removal capacity provides exactly the amount of noncondensible gas required to satisfy the remaining mass transfer requirement. These two steps determine the remaining model logic to be employed, based upon the sign of imbalance.

(17) Set VOLINT = Mixture volume required from drywell to use all remaining capacity CEFIC in condensing the associated steam and cooling the associated fog.

XMNNST = Mass of noncondensible gas associated with VOLINT.

VL2FL = 0.0

(18) Does XMNNST satisfy the remaining mass transfer requirement XMS2FL?

If No:

Go to Steps 19 – 21

If Yes:

Go to Steps 22 – 25

Steps 19 – 21: Mass Transfer Dominates
XMNNST (based upon use of all of the available heat exchanger-condenser capacity) is insufficient to satisfy the remaining mass transfer requirement XMS2FL.

(19) Set ADDRVL = mixed volume to be taken up from drywell solely to satisfy the mass transfer requirement.

(20) Add the steam (uncondensed) and fog associated with ADDRVL directly to the pressure suppression pool and remove them from the drywell atmosphere.

Transfer the noncondensible gases from the drywell to the wetwell atmosphere, while representing the heat transfer to the water that would occur during their bubbly passage through the pressure suppression pool.

(21) Set VLICMT = 0.0
CEFIC = 0.0

All material originally within the PCCS and all new material taken up from the drywell has been passed through the vent line. Also, all available heat exchanger capacity has been utilized.

Go to Step 26.

Note that VINTNC is 0.0 here while VOLINT is the mixture volume taken up from the drywell to satisfy the heat exchanger capacity.

Steps 22 – 25: Heat Removal Capacity Dominates

XMNNST (based upon satisfying the heat exchanger capacity requirement) exceeds the remaining mass transfer requirement XMS2FL. VOLINT (set in Step 17) is the mixture volume associated with XMNNST.

(22) Set VINTNC = noncondensible gas volume associated with VOLINT.

(23) Set VL2FL = noncondensible gas volume associated with XMS2FL. This is the volume that flows through the PCCS vent line this timestep based upon XMS2FL.

(24) If VINTNC > (RMVLIC + VL2FL)

Cannot take up all of the mass XMNNST (associated with volume VINTNC).

- Reduce the mixed volume to be taken up from the drywell.

$$VOLINT = VOLINT \;\; x \;\; \left(\frac{RMVLIC + VL2FL}{VINTNC}\right)$$

- Reduce the available heat capacity by the amount used

$$CEFIC = CEFIC - CEFIC \;\; x \;\; \left(\frac{RMVLIC + VL2FL}{VINTNC}\right)$$

- Reduce VINTNC to a value sufficient to fill the available PCCS void plus provide the remaining vent line mass transfer.

$$VINTNC = RMVLIC + VL2FL$$

Else

CEFOC = 0.0

All available energy is utilized if VINTNC is less than or equal to (RMVLIC + VL2FL)

(25) Adjust the material volume within the PCCS

VLICMT = VLICMT + VINTNC - VL2FL

Here VINTNC is the noncondensible gas volume to be taken up from the drywell and added to the PCCS volume.

Steps 26 – 27: Transfer of Steam, Fog, and Gas from the Drywell Atmosphere

(26) Remove the noncondensible gases associated with VOLINT from the drywell atmosphere and add them to the PCCS volume and the wetwell airspace.

If VINTNC is greater than zero here, then some of the noncondensible gases taken up from the drywell to satisfy the available heat removal capacity are not passed through to the pressure suppression pool, but rather remain within the PCCS.

Increase XMSICN(I) and ETOTIC(I) for the noncondensible gases accordingly.

For the portion of the noncondensible gases (maybe all) that are passed to the pressure suppression pool, add the masses to the wetwell atmosphere and represent the heat transfer from the bubbles to the pool, adding the residual energies to the wetwell atmosphere.

(27) Remove the steam and fog associated with VOLINT from the drywell atmosphere and add the condensate to the GDCS pool.

Step 28: Set PCCS Internal Energies for the Next Timestep

(28) Set ENGIC(I) = ETOTIC(I) for the steam, fog, and noncondensible gases within the PCCS.

This is the last step in each calculation of PCCS operation. Any material remaining within the PCCS is considered to remain at drywell pressure and may or may not fill the PCCS volume.

2.4 The Iterative Procedure

2.4.1 Purpose

The objective of this iterative procedure is to fill the PCCS volume with the mixture of gases, fog, and vapor from the drywell atmosphere to make the PCCS pressure equal to the drywell pressure. The iteration constitutes Step 3 of the PCCS operation as described in Section 2.3 and may be performed at the beginning of each timestep, depending upon the initial conditions within the PCCS volume.

2.4.2 Initial Conditions

The initial conditions within the PCCS are those established at the end of the previous timestep and fall into three categories.

a) The PCCS may be bound (filled) with noncondensible gases at the temperature of the ICS/PCC pool and the pressure of the drywell atmosphere.

b) The PCCS may be completely voided, or contain only steam and fog; in either event, there are no noncondensible gases within the PCCS.

c) The PCCS may contain a mixture of noncondensible gas and steam. If the temperature of the mixture exceeds the temperature of the ICS/PCC pool, then the noncondensible gases are cooled to the pool temperature (as explained in Section 2.3) before the iteration begins.

Initial filling of the PCCS volume from the drywell atmosphere is necessary only for cases (b) and (c), and is accomplished by means of the steps described below:

2.4.3 Iterative Steps

(1) Call the MELCOR equilibrium routine CVTWGE with input

CVMS(I) initial masses,
CVEM(I) internal energies, and
XNMCLS x VOLIC the total PCCS volume.

The calculated output includes the equilibrium values for

XMSICN(I) masses,
ETOTIC(I) internal energies,
PRIC pressure, and
TEMPIC temperature

For the equilibrium calculation, the index I represents fog (I=2), steam (I=3), and noncondensible gases (I=4, NUMMAT).

The first step is skipped in the first iteration if the PCCS is initially totally voided; in this case, the pressure PRIC is simply set to zero.

(2) Check to see if the pressure in the PCCS exceeds the pressure in the drywell after the initial equilibration calculation, which would indicate a current drywell pressure less than the pressure at the end of the previous timestep.

If the condition is met, then determine the expanded volume of the noncondensible gases at the new drywell pressure. If the expanded volume is greater than the volume of the condensers plus the source line volume, allow material to flow back from the PCCS to the drywell. The fraction of PCCS noncondensible gases to be removed from the condensers and transferred back to the drywell is:

$$F2FLBK = 1 - \frac{PCCS\ VOLUME}{VICDRY - PCCS\ VOLUME - PCCS\ SOURCE\ LINE\ VOLUME}$$

where VICDRY is the expanded volume of the noncondensible gases at the new drywell pressure:

$$VICDRY = \frac{PCCS\ \ PRESSURE\ x\ PCCS\ VOLUME}{DRYWELL\ PRESSURE}$$

FLMULT is then set to zero and the execution sequence is continued with Step 6.

(3) The mass transfer multiplier FLMULT is set depending upon the relative values of the PCCS pressure PRIC and the upper and lower boundaries of a pressure range centered on the drywell pressure PRES(IVPCSO) as follows,

..... PRES(IVPCSO) + 100
..... PRES(IVPCSO)
..... PRES(IVPCSO) – 100.

As indicated, the total width of the acceptable pressure range is 200 Pa (about 0.03 psi).

If PRIC is less than the lower boundary limit, then FLMULT is set to a positive value. Conversely, if PRIC is greater than the upper boundary limit, then FLMULT is set to a negative value. In either case, the absolute value of FLMULT is reduced by a factor of two each trip through the iterative loop.

When PRIC finally lies within the acceptable boundaries, FLMULT is simply set to zero.

(4) The volume to be transferred from the drywell to the PCCS during this iterative step is calculated from

VOL2FL = [PCCS VOLUME – VLICMT] x FLMULT

where VLICMT is the material volume at the end of the previous timestep, reduced by 10 percent. The value of VLICMT set in the initial iterative pass is used without change during all subsequent passages through the loop.

Returning to a consideration of the possible initial conditions, it should be recognized that VLICMT is zero at the end of the previous timestep if the PCCS is completely voided, in which case taking away ten percent would have no effect. The ten percent reduction is intended for cases in which noncondensible gases are present and are cooled before the iterative procedure is begun; some of the drywell atmospheric mixture must be brought into the PCCS to maintain a pressure equal to drywell pressure, and the iterative procedure accomplishes exactly this.

In fact, for the case with the PCCS completely voided at the end of the previous timestep, there is no need for iteration at all. The PCCS volume is very small in comparison with the drywell volume. Therefore, it is reasonable to assume that the PCCS is filled with a material mass and energy composition identical to that of the drywell. One pass through the iteration loop is made to confirm that the calculated PCCS pressure after filling is equal (within limits) to the drywell pressure.

What about the case in which the PCCS is bound (filled with cooled noncondensible gas) and at drywell pressure? Reducing VLICMT by ten percent here has no effect since FLMULT is zero and hence VOL2FL is zero regardless of the value of VLICMT.

The upshot of this rather complicated discussion is that VOL2FL is normally positive during the first pass through the iterative loop. An exception occurs if the PCCS pressure is already equal (within limits) to the drywell pressure. In that case, VOL2FL is zero and the iteration is not extended beyond a single pass through the loop.

(5) At this point, VOL2FL may be negative if the PCCS volume was overfilled during the previous pass through the iterative loop. Depending upon the sign of VOL2FL, the masses ADMS(I) and internal energies ADEM(I) of the steam, fog, and noncondensible gases within this volume of drywell atmosphere are added to (subtracted from) the PCCS volume. These masses and associated enthalpies are subtracted from (added to) the drywell control volume.

In these exchanges, portions of the drywell atmosphere are being transferred. Internal energy is added to or subtracted from the PCCS because a void is being either eliminated or created, as is the associated PV work term. For the drywell, gases entering or leaving do flow work upon (compression) or derive work from (expansion) the remaining gases. Hence enthalpy transfer is appropriate.

(6) CVEM(I) and CVMS(I) are adjusted depending upon the values of ADEM(I) and ADMS(I) for all materials within the PCCS atmosphere and the calculation returns to iterative step 1 unless FLMULT is zero. [FLMULT = 0 signifies that the PCCS pressure equals (within limits) the drywell pressure.]

(7) Once convergence is satisfied, VLICMT is set equal to the PCCS structural volume.

2.5 Example Results

This section provides, as an example, a discussion of the calculated PCCS operation for a MELCOR representation of the SBWR station blackout accident sequence. While reading this description, it is important to bear in mind that the available PCCS heat exchanger-condenser capacity (based upon current operation parameters) is assumed to be known each timestep; the purpose of the model is to determine the associated heat transfers and fluid flows, with due consideration of the current status of the PCCS with respect to binding. It is important to note that, for this example, no degradation in performance due to variations in the drywell pressure is assumed.

For an unmitigated station blackout accident sequence, reactor vessel depressurization would automatically occur when the vessel level reached a point about 3.6 m (12 ft) above the top of the core. The SBWR depressurization involves stepped opening of the safety relief valves, which discharge into the pressure suppression pool, followed by stepped opening of the six depressurization valves (DPVs), which discharge directly into the drywell atmosphere. The example results discussed here cover the period from just before the initial DPV actuation to five minutes thereafter.

Figure 2.1 shows the effect of the DPV openings, which begin at time 11161 seconds, upon the noncondensible gas fraction in the drywell. The actual DPV opening sequence is two valves at 11161 seconds, two valves at 11206 seconds, and two valves at 11251 seconds.

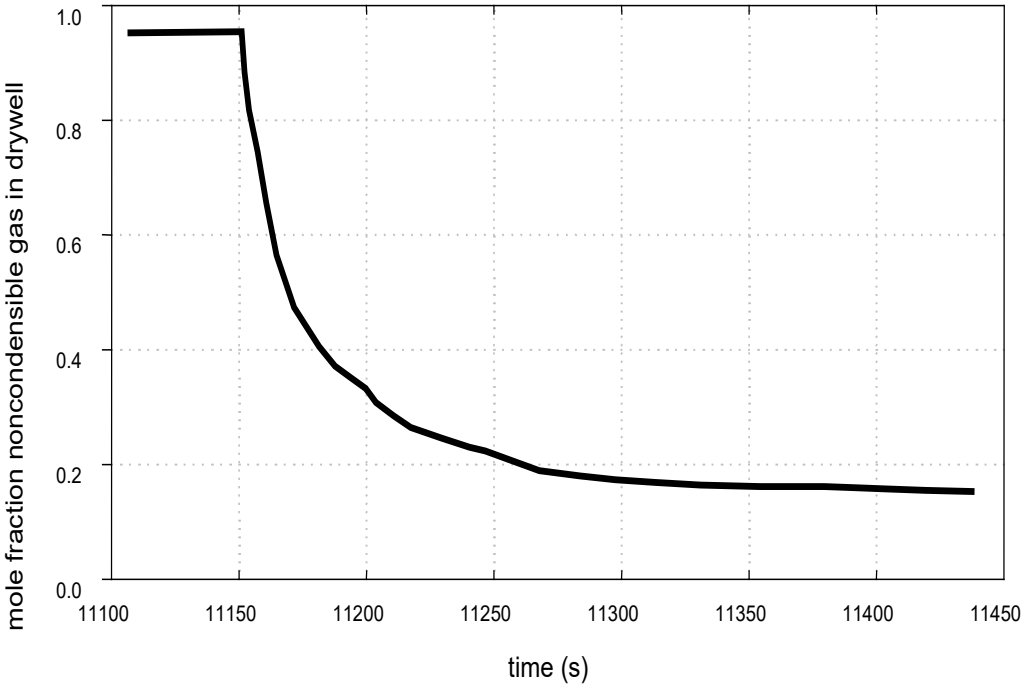


Figure 2.1 The noncondensible gas mole fraction decreases rapidly when steam is released directly into the drywell atmosphere during the final stage of an SBWR reactor vessel depressurization.

The reactor vessel depressurization also increases the drywell-to-wetwell differential pressure, as indicated by the response of variable **delpre**, shown in Figure 2.2. The variable **reqpre**, also plotted on this figure, represents the differential pressure required to induce flow through the PCCS vent line. It increases slightly during the period of the calculation as the height of the pressure suppression pool surface above the vent line terminus increases.

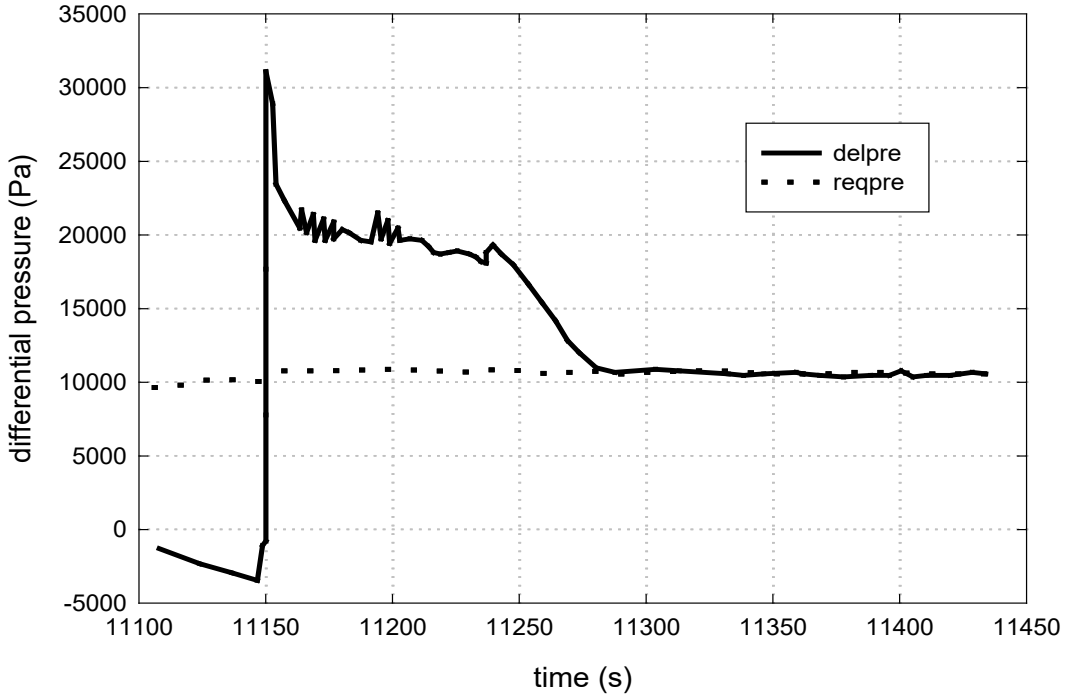


Figure 2.2 The drywell-to-wetwell pressure differential **delpre** and the differential pressure **reqpre** at which flow through the PCCS vent line is initiated.

At this point, it is necessary to consider the variation in PCCS performance in accordance with current conditions. The PCCS heat exchanger capacity is determined at the beginning of each timestep based (in order of increasing importance) upon (1) the current drywell-to-wetwell differential pressure, and (2) the current mole fraction of noncondensible gas in the drywell (considering the current ICS/PCC pool temperature and interpolating between values for two reference pool temperatures). The tabular input employed for this example calculation is listed in Table 2.1 through Table 2.3. The basic capacity per PCCS unit is 10 MW_t at an ICS/PCC pool temperature (saturation) of 374.15 K (213.8 F), a drywell-to-wetwell pressure differential of 7239.5 Pa (1.05 psi), and a drywell noncondensible gas fraction of 0.0 (pure saturated steam). As stated above, the performance of the condenser is assumed to be constant over all source volume pressures.

Table 2.1 Tabular input example for variation of PCCS performance with drywell-wetwell differential pressure

Differential Pressure (Pa)	psi	Variation Factor
0.0	0.00	1.000
7239.5	1.05	1.000
8618.5	1.25	1.072
10342.1	1.50	1.153
12065.8	1.75	1.227
13789.5	2.00	1.294
15423.6	2.24	1.353

Table 2.2 Tabular input example for variation of PCCS performance with the drywell noncondensible gas mole fraction at an ICS/PCC pool temperature of 323.16 K

Noncondensible Gas Mole Fraction	Variation Factor
1.00	0.00
0.10	0.60
0.05	0.82
0.02	0.90
0.01	0.96
0.00	1.00

Table 2.3 Tabular input example for variation of PCCS performance with the drywell noncondensible gas mole fraction at an ICS/PCC pool temperature of 373.16 K

Noncondensible Gas Mole Fraction	Variation Factor
1.00	0.00
0.10	0.60
0.05	0.82
0.02	0.90
0.01	0.96
0.00	1.00

The example calculation represents the operation of all three PCCS units. Changes in the ICS/PCC pool temperature are assumed to have no effect upon the PCCS system performance, chiefly because the pool is sufficiently large that its temperature increase is small during the period of the calculation. It may be noted by comparing the variation factors listed in Table 2.2 and Table 2.3 that no credit has been given for an enhancement of the PCCS heat exchanger capacity for ICS/PCCS pool temperatures below saturation. At the time that this example calculation was performed, no information concerning this enhancement was available. Subsequently, it has become apparent that such enhancement should be represented by providing different values in Table 2.2 and Table 2.3. Similarly, the variation in performance due to source volume pressure changes were added when the need for such a reduction became apparent. (See input card CND_PCCS02 for additional information.)

The drywell-to-wetwell differential pressure affects the heat exchanger performance because it determines the (forced-convection) velocity within the heat exchanger tubes. The velocity, in turn, affects the heat transfer coefficient (h) at the inner surface of the tubes. A conventional expression commonly used has the form

$$h = (const.)\ x\ Re^{0.8},$$

where Re (the Reynolds number) includes the velocity. As a result, the heat transfer coefficient for various differential pressures between the drywell and the wetwell can be represented (assuming all other variables are constant) by

$$h = (\text{const.})\ x\ (\text{differential pressure})^{0.4}.$$

Thus, as indicated in Table 2.1, the PCCS capacity is enhanced as the differential pressure increases.

By far the largest effect upon PCCS capacity derives from changes in the noncondensible gas fraction of the gas entering the PCCS from the drywell. This large influence can be observed in Figure 2.3., which compares the current (three-unit) PCCS capacity to the drywell noncondensible gas mole fraction (also shown in Figure 2.1). It is obvious that the increase in available capacity shown in Figure 2.3 is inversely proportional to the decrease in noncondensible gas mole fraction. This large effect of the noncondensible gas fraction in reducing the condensation effectiveness is well known. The tabular input reproduced in Table 2.2 and Table 2.3 is derived from information provided in the paper, Heat Removal of Isolation Condenser Applied as a Passive Containment Cooling System by H. Nagasaka et al., of the Nuclear Energy Group, Toshiba Corporation. [2]

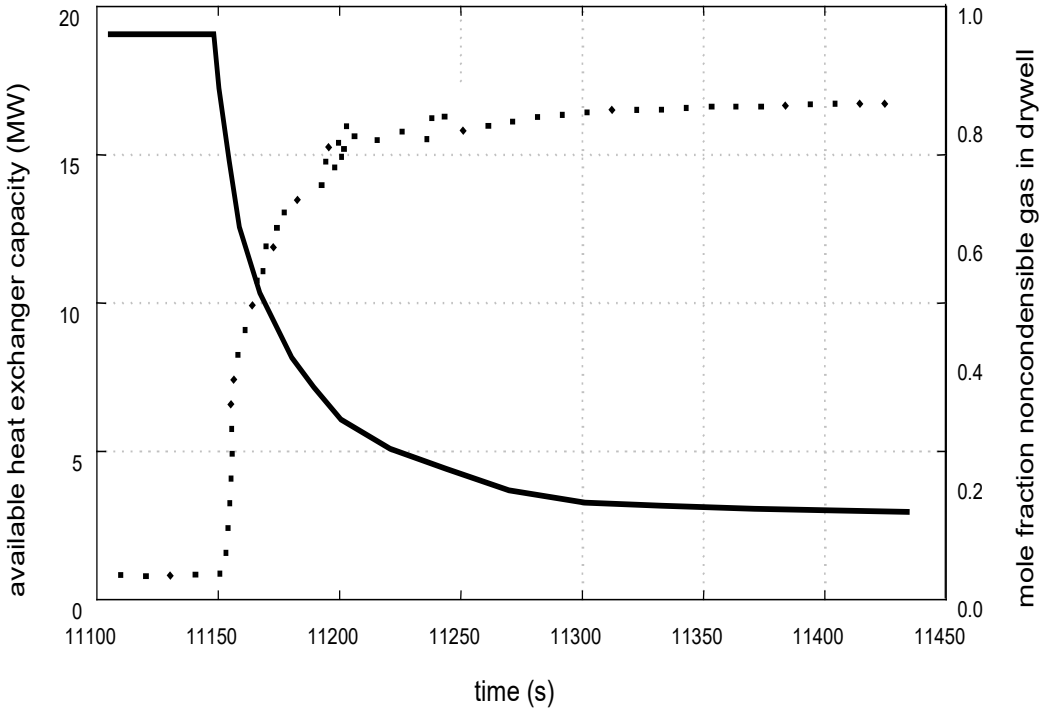


Figure 2.3 The available PCCS heat exchanger capacity (dots) is primarily determined by the drywell noncondensible gas mole fraction (solid line).

The available (three-unit) PCCS capacity is shown again, as variable **pitcef** on Figure 2.4. It should be recognized that three PCCS units operating under base conditions would have a combined capacity of 30 MW_t, whereas the maximum value of **pitcef** shown on Figure 2.4 is about 17 MW_t. Again, this reduction is primarily due to the presence of noncondensible gas in the drywell atmosphere, which is always the case.

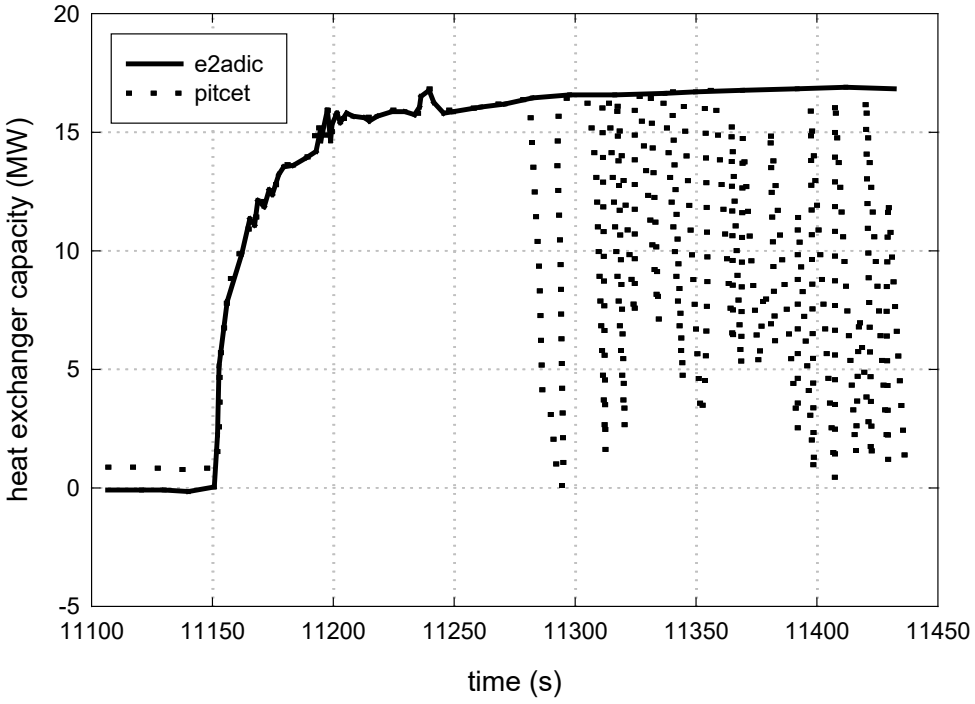


Figure 2.4 The available PCCS (three-unit) heat exchanger capacity **pitcet** and the power **e2adic** actually utilized.

Also shown in Figure 2.4 is the variable **e2adic**, which is the heat exchanger power being used. As indicated, none of the available capacity is utilized before the reactor vessel depressurization begins. This is because the PCCS heat exchanger tubes are “bound,” or filled with noncondensible gas. Once reactor vessel depressurization begins, however, (1) the available heat exchanger capacity greatly increases, and (2) all of this capacity is used.

The reason that all of the available capacity is used during the period immediately after DPV opening is that the vent line flow induced by the increasing drywell pressure now sweeps the noncondensible gases from the PCCS each timestep, permitting the mixture of gases and steam within the drywell to enter. The total vent line flow **pltifl** and the noncondensible gas vent line flow **pltnfl** are shown in Figure 2.5. It should be noted that the vent line flow initially consists entirely of noncondensible gas; all of the steam entering the PCCS during this initial period is condensed.

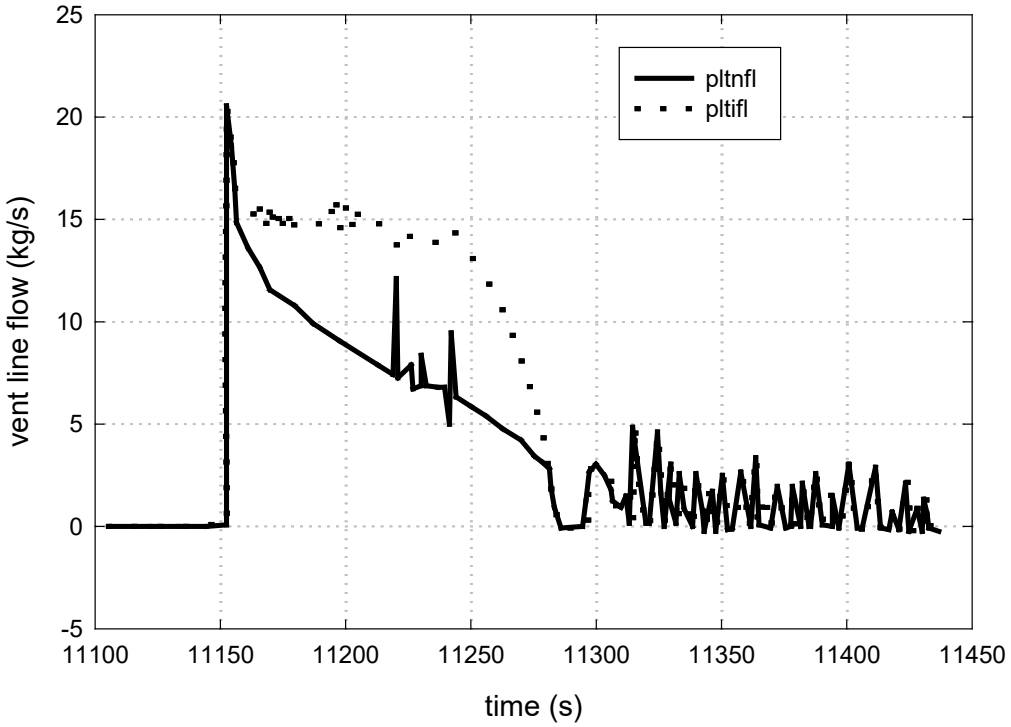


Figure 2.5 The total mass flow **pltifl** through the PCCS vent line and the associated flow **pltnfl** of noncondensible gases.

Steam flow through the vent (the difference between the two plotted variables) does not begin until about 20 seconds after vent line flow begins. Carryover of steam begins at this time because the concentration of steam in the drywell atmosphere has reached a level beyond the available heat exchanger capacity (even though the available capacity is also increasing; see Figure 2.3).

It is instructive to consider the events illustrated in these figures that occur just prior to time 11300 seconds. As shown on Figure 2.2, the drywell-to-wetwell differential pressure drops below the value needed to sustain PCCS vent line flow. This is substantiated by Figure 2.5, where the vent line flow is shown to be zero during this period. Figure 2.4 shows that the portion of available PCCS heat exchanger capacity actually used during this period decreases toward and ultimately reaches zero. This demonstrates that some time is required for the PCCS to fill with noncondensible gases and become bound after vent line flow ceases.

Almost exactly at time 11300 seconds, the drywell-to-wetwell differential pressure becomes sufficient to restore vent line flow (Figure 2.2), vent line flow (all noncondensible gas) is restored (Figure 2.5), and all available capacity is used (Figure 2.4) to condense the steam brought in with the mixed atmosphere from the drywell.

After time 11300 seconds, the drywell-to-wetwell differential pressure oscillates about the value required to induce vent line flow (Figure 2.2). During the periods when vent line flow occurs, this flow consists entirely of noncondensible gas (Figure 2.5). During the periods when vent line flow does not occur, the portion of the available capacity that is actually used decreases (Figure 2.4) as the PCCS tends to fill with cooled noncondensible gas. However, a fully bound condition is never attained.

That a fully bound condition is never attained during this final period of the example calculation is a testimony to the effectiveness of the PCCS system in controlling the drywell-to-wetwell differential pressure. Whenever the PCCS performance falters, this differential pressure increases, clearing the vent line and restoring the PCCS performance.

2.6 Effect of the Drywell Pressure on PCCS Operation

The nominal capacity of each PCCS heat exchanger-condenser is reported in the SSAR (Section 6.2.2.1) as 10 MW_t for conditions where the tubes are filled with pure saturated steam at 308 kPa (45 psia) and 407 K (273 °F), and the ICS/PCC pool temperature is 374 K (214 °F). The available capacity under accident conditions is, however, never more than about sixty percent of this because of the presence of noncondensible gases in the drywell atmosphere.

To estimate the variation in performance of the PCCS as a result of changes in the drywell pressure, the heat transfer (q) at the base condition is compared to the heat transfer rates at different pressures. The performance variation factor is thus calculated by dividing the heat transfer at the new condition by the heat transfer at the base condition. Ratios greater than one signify an improvement in performance.

The variation in performance = *q (new condition) / q (base condition)*

$$= \frac{h_{new} A\ (T_{steam}(new) - T_{wall})}{h_{base} A\ (T_{steam}(base) - T_{wall})}$$

where

- h = heat transfer coefficient ($W/m^2/K$),
- A = the surface area (m^2),
- T_{steam} = temperature of the steam (saturation temperature at the pressure of the drywell) (K), and
- T_{wall} = temperature of the tube wall (K) (assumed to be same as the temperature of the condenser pool, which is the saturation temperature at atmospheric conditions).

The heat transfer coefficient [3] for condensing steam in various geometries is examined next:

$$h = C\left[\frac{\rho_f(\rho_f - \rho_{steam}){k_f}^3 g \sin\theta \;\; h_{fg}}{\mu_f L\ (T_{steam} - T_{wall})}\right]^{1/4}$$

where

ρ_f = the density of the liquid film (kg/m^3),

ρ_{steam} = the density of the steam (kg/m^3),

T_{steam} = temperature of the saturated steam (K),

T_{wall} = temperature of the wall, the temperature of the condenser pool (K),

$\sin\theta$ = sine of the angle of the tubes with the horizontal; for vertical tubes, the value is one,

h_{fg} = latent heat of the steam being condensed (J/kg),

μ_f = viscosity of the film (Pa*s),

k_f = thermal conductivity of the film (W/m/K),

L = equivalent length (m),

g = gravitational constant (9.81 m/s^2), and

C = a constant value that must be calculated depending on the geometry, being either a vertical plate or a cylindrical tube.

It is important to note that this equation for the heat transfer coefficient is for condensers with relatively low vapor Reynolds numbers, less than 35,000. This equation underestimates the heat transfer coefficient for condensers with a higher value; however, since the primary purpose of the equation as used here is to determine the variation in performance (and not the absolute value of the heat transfer) at low pressures, the use of the equation is appropriate.

The variation in performance becomes:

$$= \frac{h_{new}(T_{steam}(new) - T_{wall})}{h_{base}(T_{steam}(base) - T_{wall})}$$

$$= \frac{\left[\frac{[\rho_f(\rho_f - \rho_{steam}){k_f}^3 h_{fg}]}{\mu_f(T_{steam} - T_{wall})}\right]_{(new)}^{1/4} (T_{steam}(new) - T_{wall})}{\left[\frac{\rho_f(\rho_f - \rho_{steam}){k_f}^3 h_{fg}}{\mu_f(T_{steam} - T_{wall})}\right]_{(base)}^{1/4} (T_{steam}(base) - T_{wall})}$$

The performance variation factors for pressures are shown in Table 2.4. The base operating condition for the PCCS is at 0.3 MPa as noted in Table 2.4 by a value of unity for the multiplication factor. Also, as the pressure increases the performance of the PCCS improves. Thus, it is obvious why the ICS (which operates at a pressure of 7.4 MPa versus 0.3 MPa for the PCCS) has an energy removal capacity that is three times larger than the PCCS but has a smaller heat transfer surface area.

Table 2.4 Variation in PCCS Performance with Pressure in Drywell.

Pressure (Pa)	**Multiplication Factors for PCCS Performance**
0.000E + 00	0.0000
6.113E + 02	0.0000
5.000E + 04	0.0000
1.000E + 05	0.0000
1.500E + 05	0.4250
2.000E + 05	0.6660
2.500E + 05	0.8495
3.000E + 05	1.0000
3.500E + 05	1.1289
4.000E + 05	1.2425
4.500E + 05	1.3450
5.000E + 05	1.4386
6.500E + 05	1.6807
7.000E + 05	1.7518

Table 2.4 provides multiplication factors for performance variation for drywell pressures up to 0.7 MPa; however, it is recognized that the SBWR containment is predicted to fail at pressures greater than 0.65 MPa.

3. ICS Model

3.1 Introduction and Concept

The ICS (Isolation Condenser System) is a safety-related passive operating system designed to remove the core decay heat directly from the reactor vessel following reactor shutdown and isolation. It is described in Section 5.4.6 of the SBWR Standard Safety Analysis Report (SSAR) [1]. Unlike the PCCS, the ICS is not continuously in operation. A motor-operated valve must be opened (or, if power is lost, a nitrogen-operated bypass valve must open) in order to initiate operation of the ICS.

Flow through the ICS is first induced by the action of condensate draining from the condenser tubes into the reactor vessel annulus. The drainage draws in steam from the upper portion of the reactor vessel; this steam is condensed and returned to the vessel annulus. In the event that the ICS becomes “bound” by noncondensible gases, a vent line is provided to permit release of the gases trapped within the ICS to the pressure suppression pool.

The flow through the vent line is started and stopped by an active control system that continuously monitors the reactor vessel pressure. Once the vessel pressure reaches the vent opening setpoint (implying the ICS is bound), the valves on the vent line open allowing the accumulated noncondensible gases to escape to the pressure suppression pool, thereby reinitiating operation of the ICS.

The vent line valves are signaled to close once the vessel pressure has decreased below the reset (closing) setpoint for the vent. A time delay circuit is integrated into the logic to protect the vent valves from excessive cycling.

The ICS modeling concept is the same as for the PCCS in that it is recognized that it is not a purpose of the MELCOR code to predict ICS performance based upon first principles. Rather, based upon the available experiment evidence, MELCOR should adequately represent the effects of the ICS heat exchanger-condenser system under the boundary conditions that would be imposed by accidents.

3.2 Operation of the ICS Model

The same basic algorithms, contained in Subroutine CNDRN1, are used to model both the ICS and the PCCS. There is, however, a block of coding specific to the ICS. This coding block mimics the operation of the ICS vent line control logic, which has no counterpart within the PCCS (the flow through the PCCS vent line is limited only by the submergence depth of the vent line in the pressure suppression pool). The following is a description of the significant differences between the operating characteristics of the ICS and the PCCS and the logic enhancements required to represent the ICS.

The ICS operates at pressures near normal reactor vessel pressure, approximately 7 MPa, as compared to the PCCS, which operates at post accident drywell pressures of less than 0.50 MPa.

Because of the difference in operating pressures, allowances had to be made in the calculation of the vent line capacity to limit the flow to sonic velocity (choked flow) at the exit conditions. This was done by the use of the Modified Darcy Formula taken from the Crane Technical Paper No. 410.[4] The Darcy Formula estimates a mass flow rate for compressible flow using a net expansion factor through the pipe and the differential pressure between the reactor vessel and the choke point at the pipe exit. (The pressure at the exit condition can be easily determined if the flow is choked.) The determination of the net expansion factor serves to limit the flow through the pipe to sonic velocity at the pipe exit conditions.

The mass flow rate is determined in a subroutine CNDICF, which is used for both the PCCS and the ICS vent line flow calculations. CNDICF first determines the resistance coefficient for the vent line. Using the resistance coefficient, the maximum net expansion factor and the maximum $\Delta P/P$ for sonic velocity are found by interpolating between the values found on page A-22 of the Crane Technical Paper for a k value of 1.4. If the pressure in the PCCS/ICS minus the wetwell pressure divided by the PCCS/ICS pressure is greater than the value found for $(\Delta P)/P$, then the flow is choked. If the flow is not choked, then a linear interpolation is performed between zero and the calculated differential pressure to determine the net expansion factor. If the flow is choked, then the maximum $(\Delta P)/P$ is used to determine the pressure at the exit condition, and the net expansion factor is simply equal to its maximum value. The mass flow rate can then be estimated.

Because of the higher pressures at which the ICS condensers operate, the condenser tube walls are significantly thicker than for the PCCS condensers. This greater tube wall thickness may require a different performance degradation curve to represent system response to increases in noncondensible gas mole fractions. Provision is made for this new curve, when available, to be represented in the ICS set of user-input tabular functions, which are applied in a manner identical to the PCCS tabular functions described in detail in Section 2.3.

The heat removal capacity of a single ICS unit is at least 30 MW_t at a reactor pressure of 7.420 MPa (1050 psig) when fed by pure saturated steam. The large (factor of 3) increase in capacity over the PCCS is a direct result of the increase in steam density at reactor vessel pressure (where 1 m^3 of steam contains approximately 8 times the mass of the same volume at drywell conditions). Therefore, the ICS has a greater amount of stored energy within the fluid contained in the condenser tubes.

As described in Section 3.1, the vent line for each ICS unit contains a motor-operated valve, which is actuated upon a high pressure within the reactor vessel such as would occur whenever the condenser tubes become bound with noncondensible gases.

Unlike the PCCS, the ICS condensers are not expected to operate after the equalization of reactor vessel and drywell pressures that would occur under accident conditions as a result of ADS actuation and DPV sequencing. This conclusion is not stated explicitly in the SSAR but follows from information contained in Section 5.4.6 and the control diagrams provided in Volume 15 of the SSAR. The control diagrams indicate that the controllers for the vent line valves receive their signals for automatic operation from reactor vessel pressure sensors exclusively.

After blowdown, these controllers would no longer receive a high-pressure signal since the vessel would be at the same pressure as the drywell. Thus, the ICS would quickly become bound by noncondensible gases with no provision for venting except by means of operator intervention. However, no guidance to the operator concerning this action can be found in the SBWR Emergency Procedure Guidelines (EPGs).

The drain line from the ICS returns condensate directly to the reactor vessel annulus. The elevation of the ICS condensers provides a sufficient gravity head so that the condensate drains to the vessel annulus even though the annulus water level may be several meters above the condensate return line. A loop seal is provided in the drain line to prevent steam from entering the condensers via this line should the water level fall below the connection point to the reactor vessel.

3.3 Example Results

Several test calculations have been performed using the ICS model with two units in operation for various accident sequences. The accident sequences considered are loss of offsite power (station blackout), a main steam line LOCA, and a break in the bottom head drain line. For the station blackout calculation, the ICS was predicted to operate continuously and to cause depressurization of the reactor vessel without SRV or ADS actuation, thus preventing loss of reactor coolant inventory and circumventing core degradation.

For the bottom head LOCA calculation, the ICS was predicted to operate until shortly after ADS actuation, when drywell atmosphere begins to be pulled into the reactor vessel (through the open DPVs) as the water drains from the bottom of the vessel. Subsequently, the presence of noncondensible gases within the reactor vessel causes rapid binding of the IC condenser tubes and without vent actuation, ICS operation terminates. The main steam line LOCA calculation shows a similar behavior with the ICS slowly becoming bound with the noncondensible gases that arise from hydrogen generation in the core and from the small amount of drywell atmosphere that mixes with the reactor vessel atmosphere after vessel depressurization.

To test the logic of the vent line control valve, additional calculations were performed in which a large amount of nitrogen was arbitrarily placed into the reactor vessel upper head for the station blackout and for the main steam line LOCA accident sequences. This

provides an overpressure of noncondensible gas such that the vessel water is initially subcooled. The large noncondensible gas mole fraction at the isolation condenser inlet limits the ICS capacity (while operating) to a value insufficient to remove the decay heat. These test calculations show a very short period of ICS operation prior to binding.

Because of the inability of the ICS to remove any energy while bound, the calculated pressure in the reactor vessel increases until the vent valve opening setpoint is reached. The vent valve then opens to remove noncondensible gases from the ICS tubes to the wetwell and thereby restore ICS operation. While the vent line is open, the pressure in the reactor vessel decreases slightly, which leads to closing of the vent valve.

This predicted cyclic behavior continues with increasing frequency until the water within the reactor vessel reaches the saturation temperature and the rate of vessel pressurization increases markedly. Subsequent ICS vent actuation does not provide sufficient gas release through the small vent line to prevent the increasing vessel pressure from reaching the SRV opening setpoint. The action of opening the SRVs forces most of the nitrogen out of the reactor vessel and reduces the noncondensible gas mole fraction from approximately fifty percent to less than one percent. This produces a steam-rich environment within the ICS so that operation can resume.

For the main steam line LOCA, the ICS also becomes quickly bound, but flow through the break removes most of the imposed nitrogen from the reactor vessel. However, the break flow also serves to prevent the reactor vessel pressure from ever increasing above the vent valve opening setpoint; therefore, the ICS remains bound after operating for only a short time after the accident is initiated. (Possible operator action to remote-manually open the vent valve was not considered in this calculation.)

Similar to the PCCS, the ICS efficiency degrades as the pressure in the reactor vessel decreases. To estimate this degradation, the same methodology described in Section 2.6 is utilized.

The multiplication factors for variation in performance for pressures are shown in Table 3.1. (Factors greater than unity signify an improvement in performance.) The base operating condition for the ICS is at 7.4 MPa as noted in Table 3.1 by a value of unity for the multiplication factor.

Table 3.1 Variation in ICS Performance with Pressure in the Reactor Vessel

Pressure (Pa)	Multiplication Factors for ICS Performance
0.000E + 00	0.0000
6.113E + 02	0.0000
5.000E + 04	0.0000
1.000E + 05	0.0000
1.500E + 05	0.1080

Pressure (Pa)	Multiplication Factors for ICS Performance
2.000E + 05	0.1692
2.500E + 05	0.2159
3.000E + 05	0.2541
3.500E + 05	0.2869
4.000E + 05	0.3157
4.500E + 05	0.3418
5.000E + 05	0.3655
6.500E + 05	0.4271
7.000E + 05	0.4451
7.500E + 05	0.4624
8.000E + 05	0.4787
8.500E + 05	0.4943
9.000E + 05	0.5092
9.500E + 05	0.5237
1.000E + 06	0.5376
1.100E + 06	0.5643
1.200E + 06	0.6507
1.300E + 06	0.6139
1.400E + 06	0.6368
1.500E + 06	0.6591
1.750E + 06	0.6661
2.000E + 06	0.6980
2.250E + 06	0.7259
2.500E + 06	0.7512
3.000E + 06	0.7956
3.500E + 06	0.8328
4.000E + 06	0.8645
5.000E + 06	0.9159
6.000E + 06	0.9556
7.000E + 06	0.9883
7.200E + 06	0.9942
7.400E + 06	1.0000
7.600E + 06	1.0054
7.800E + 06	1.0106
8.000E + 06	1.0156
8.200E + 06	1.0204
8.400E + 06	1.0252
8.600E + 06	1.0298
8.800E + 06	1.0340
9.000E + 06	1.0381
1.000E + 07	1.0568

4. Interface with MELCOR

The information for the condensers is stored in the ESF Package of the MELCOR database contiguous to the information for the FCL Package. A special routine to process PCCS/ICS model input has also been added for use in calculations for which these models are to be exercised. These modifications to the MELCOR database are bypassed (as are the PCCS/ICS model routine CNDRN1) unless the PCCS and/or ICS input cards are included in the MELGEN input deck.

If the user requests that the PCCS model be invoked for a calculation, then it is necessary that the control volume numbers representing the drywell, wetwell, ICS/PCC pool, and the Gravity-Driven Cooling System (GDCS) be provided on a dedicated MELGEN input card. If the user does not provide this card, the PCCS model is bypassed. An additional dedicated card is required to indicate the tabular functions that represent the PCCS performance adjustments (depending upon operating parameters).

If the ICS model is to be exercised in a calculation, the user must provide the control volume numbers for the reactor vessel upper head and annulus, ICS/PCC pool(s), and the wetwell. Similar to the case for the PCCS, if the input card carrying this information is not provided, the ICS model is bypassed.

A few simple descriptive input numbers for the PCCS and/or ICS are also required when these models are to be exercised. This special input consists of the volume of the condensers, the source line volume, the basic capacity of one unit of the condensers, and the dimensions of the vent line (minimum diameter and equivalent length) used in determining the mass flow. The user also inputs the number of units (maximum of three) that are to be operating.

For the ICS, the setpoints for the vent valve control logic are also required. The number of operating condensers may be changed during a calculation. The CND Package Users' Guide describes the input to both MELGEN and MELCOR required for operation of the PCCS and/or ICS models, and the plot variables and associated special external data files that may be created.

Because the condenser is part of the ESF Package, the condenser energy balance does not have a separate listing under the GLOBAL energy balance edit, but rather is combined with the FCL Package so that an overall ESF energy balance is given. Currently, however, the FCL Package does not have a separate energy balance so the energy balance for the ESF Package represents the condenser package exclusively. For a typical calculation, a relative energy error of approximately 1×10^{-7} percent is produced.

References

1. SBWR Standard Safety Analysis Report (SSAR), GE Nuclear Energy, 25A5113 Rev. A, (1992).

2. H. Nagasaka et al., *Heat Removal of Isolation Condenser Applied as a Passive Containment Cooling System*, of the Nuclear Energy Group, Toshiba Corporation (1990).

3 J.P. Holman, *Heat Transfer*, McGraw-Hill, Inc., New York, (1963).

4. “Flow of Fluids through Valves, Pipes, and Fittings,” Technical Paper 410, Crane Co., Chicago, IL (1969).

Core (COR) Package

The MELCOR Core (COR) package calculates the thermal response of the core and lower plenum internal structures, including the portion of the lower head directly below the core. The package also models the relocation of core and lower plenum structural materials during melting, slumping, and formation of molten pool and debris, including failure of the reactor vessel and ejection of debris into the reactor cavity. Energy transfer to and from the Control Volume Hydrodynamics (CVH) package and the Heat Structure (HS) package is calculated. This Reference Manual gives a description of the physical models in the COR package, including the nodalization scheme and calculational framework of the package, the heat transfer and oxidation models, the mass relocation models, and the default lower head model. Since the release of MELCOR 1.8.6 version, many new modeling enhancements have been added to the COR package to improve the capabilities of the code to better represent the late-phase behavior of severe accidents. As part of this development, the Bottom Head (BH) package was eliminated, and features formerly offered by the BH package that were missing from the COR package representation have been added to the COR package.

User input for running MELGEN and MELCOR with the COR package activated is described in the COR Package Users' Guide.

List of Figures

List of Tables

1. Introduction

The MELCOR COR package calculates the thermal response of the core and lower plenum structures, including the portion of the lower head directly beneath the core, and models the relocation of core materials during melting, slumping, and formation of molten pool and debris. Fuel pellets, cladding, grid spacers, canister walls (for boiling water reactors [BWRs]), core baffles and formers (for pressurized water reactors [PWRs]), other structures (e.g., control rods or guide tubes), molten pools, and particulate debris are modeled separately within individual cells, the basic nodalization unit in the COR package. Either BWR or PWR systems may be modeled, as specified on record COR_RT. (For the convenience of the user and the sake of clarity, numerous cross-references are made in this document to specific input records and quantities in the COR Package Users' Guide. The user should consult both documents for a more complete understanding of the models and their implementation.)

Since the release of MELCOR 1.8.6 version, many new modeling enhancements have been added to the COR package to improve the capabilities of the code to better represent the late-phase behavior of severe accidents. These new models include hemispherical lower head geometry, models for simulating the formation of molten pools both in the lower plenum and the upper core, crust formation, convection in molten pools, stratification of molten pools into metallic and oxide layers, and partitioning of radionuclides between stratified molten pools.

All-important heat transfer processes are modeled in each COR cell. Thermal radiation within a cell and between cells in both the axial and radial directions is calculated, as well as radiation to boundary structures (e.g., the core shroud or upper plenum, which are modeled by the Heat Structure package) from the outer and upper COR cells. Radiation to a liquid pool (or to the lower head, if a pool is absent) and to steam is also included. Heat transfer within fuel pellets and across the fuel cladding gap is evaluated. Axial conduction between segments of components in adjacent cells is modeled, as is radial conduction within core plates and within debris beds that are not interrupted by BWR canister walls. Intracell conduction is calculated between particulate debris and other components with which it is in intimate contact. An option is available to include radial conduction between the core and radial boundary heat structures. An analytical model for axial conduction is applied within structures that are partially covered with a liquid pool. Convection to the control volume fluids is modeled for a wide range of fluid conditions and structure surface temperatures, including nucleate and film boiling.

Oxidation of Zircaloy and steel is modeled as limited by both solid-state diffusion of oxygen through the oxide layer and gaseous diffusion of steam or oxygen through the mixture. The reaction of B_4C with steam is also modeled.

The core degradation model treats eutectic liquefaction and dissolution reactions, *candling* of molten core materials (i.e., downward flow and refreezing), local blockages

formed from refrozen materials, formation and heat transfer from convecting molten pools, and the formation and relocation of particulate debris. Geometric variables (e.g., cell surface areas and volumes) are updated for changing core geometry.

Many of the various physics models can be selectively disabled by setting the flags on MELCOR input record COR_TST. This action might be appropriate for testing purposes or to bypass phenomena that are not expected to arise during a particular calculation.

1.1 Nodalization Scheme

1.1.1 Core/Lower Plenum

The core and lower plenum regions of the reactor vessel are divided into concentric radial rings and axial levels, as shown in Figure 1.1; the numbers of rings and levels are input by the user on records COR_RP and COR_ZP, respectively. A particular radial ring and a particular axial level designate a COR cell, whose cell number is specified by a pair of two integers; the first integer represents the axial level number, and the second represents the radial ring number. For example, cell (2,1) denotes the second axial level and the first radial ring. Radial rings are numbered from the center out, and axial levels are numbered from the bottom head up. *This nodalization scheme applies only to structures treated by the COR package and is independent of the control volume nodalization specified for the CVH package.* The interface between the COR and CVH packages is discussed later in this section.

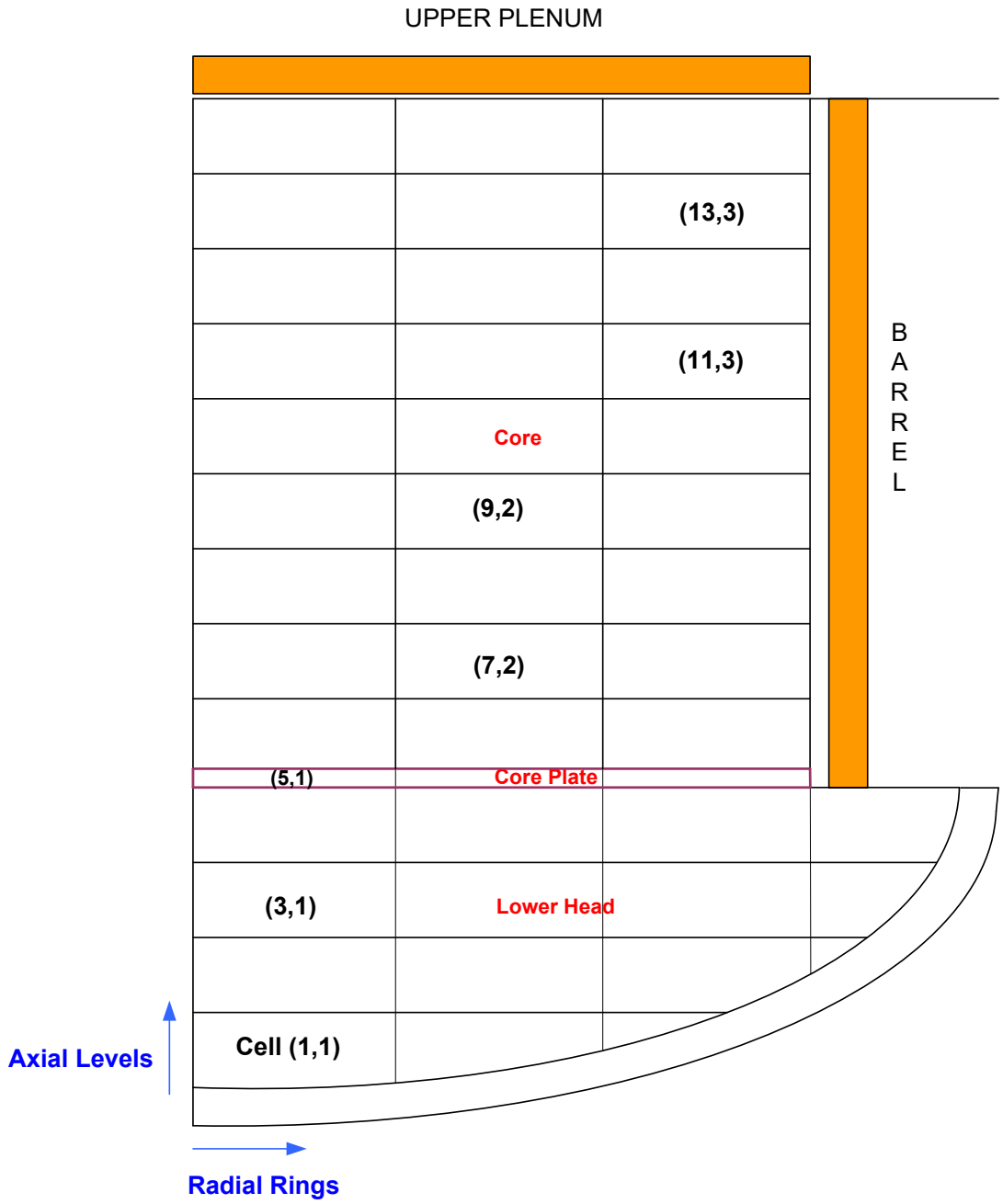


Figure 1.1 Core/lower Plenum Nodalization

Each cell may contain one or more components, as shown Figure 1.2. A number of distinct intact components are modeled: (1) fuel; (2) cladding; and (3) BWR canister walls, split into two parts: one part that is not adjacent to the control blade and another part that is; (4) supporting structure; (5) PWR core baffle (shroud); (6) PWR core formers between

the baffle and the core support barrel; (7) nonsupporting structure. The primary difference between the supporting and nonsupporting structure components is whether they have the ability to support other core components (core support structures) or not (control rods or blades). Note that the “Other Structure” component, OS, is not retained in the Fortran 95 based versions of MELCOR (e.g., MELCOR 2.x). This OS component is now treated as PWR core formers. The structure shown in Figure 1.2 may represent supporting and/or nonsupporting structures in the new representation.

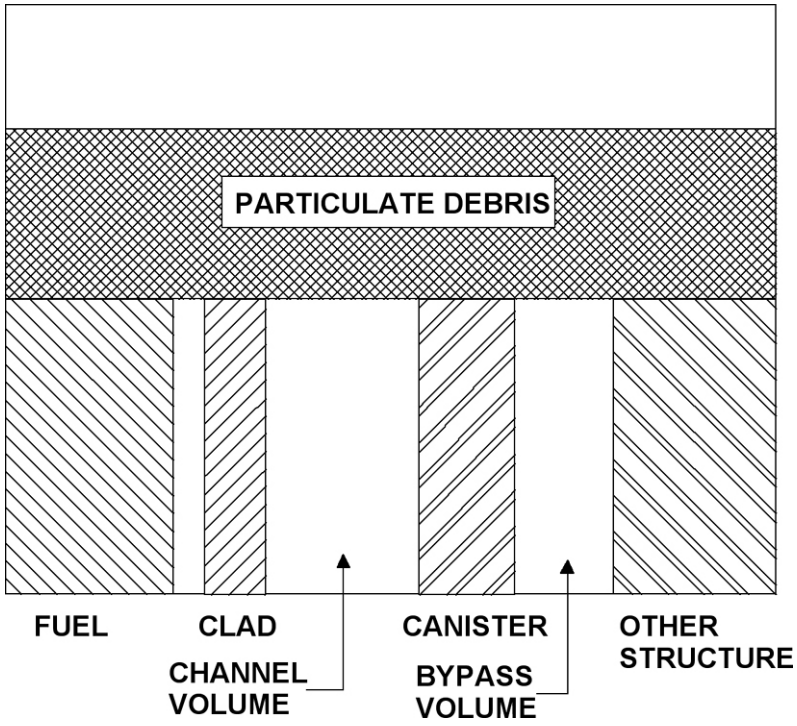


Figure 1.2 Core Cell Components

A core cell may also contain particulate debris (rubble) resulting from the collapse of fuel rods and other core components. It may also contain molten materials, local to the cell or part of a coherent molten pool that extends through a number of cells. In a BWR, such components, particulate debris or molten pool, may reside either inside or outside the channel box, in the channel or bypass region, respectively. As with MELCOR 1.8.6, MELCOR 2.x also allows definition of a bypass region in a PWR, between the core shroud (baffle) and the core support barrel in the outermost ring of the active core.

Particulate debris and molten pools in the channel are distinguished from those in the bypass, with separate components used for each. The distinction exists only for a BWR, and only for core cells that have distinct channel and bypass regions. For a PWR, a bypass region has been defined in the outer active core ring to represent the bypass volume between the shroud and the core support barrel (Section 1.1.3). Even then, most

of the distinction is lost when the channel box or core shroud fails and the fields in the two regions are assumed to have mixed and equilibrated. However, both sets of materials must continue to be tracked separately after canister failure because they typically occupy space in different CVH control volumes. When the canister fails, the transfer of debris between the channel and the bypass is not instantaneous. It is controlled by a time constant with a default value of one second, adjustable through sensitivity coefficient array C1021.

Conglomerate debris (i.e., core material that has melted and resolidified), is modeled as an integral part of the component onto which it has frozen, which may be any one of the intact components listed above except for intact fuel.

The following table identifies each component by its component number and component identifier, which are often used in the COR package documentation. The duplication of component numbers for CB, SH, and HR reflects the fact that, because only one of each pair can occur, the same portion of the database is used for both.

Table 1-1 Components modeled in COR package

1	FU	Intact fuel component
2	CL	Intact cladding component
3	CN	Intact canister component (portion not adjacent to the control blade)
4	CB	Intact canister component (portion adjacent to the control blade)
4	SH	Intact PWR core shroud (baffle)
4	HR	Heavy reflector (HR reactor type)
5	FM	Intact PWR core formers
6	PD	Particulate debris component (in the channel)
7	SS	Supporting-structure component
8	NS	Nonsupporting structure component
9	PB	Particulate debris component in the bypass (if any)
10	MP1	Oxidic molten pool (portion in the channel)
11	MB1	Oxidic molten pool in the bypass (if any)
12	MP2	Metallic molten pool (portion in the channel)
13	MB2	Metallic molten pool in the bypass (if any)
7/8	RK	Spent fuel pool rack (SFP-PWR=7, SFP-BWR=8)
-	FU-n	Multi-rod group, n, of intact fuel component
-	CL-n	Multi-rod group, n, of intact cladding component
-	NS-n	Multi-rod group, n, of intact nonsupporting structure component

Eight materials are currently modeled in the COR package: (1) UO_2, (2) zircaloy, (3) steel, (4) ZrO_2, (5) steel oxide, (6) control rod poison, which may be either boron carbide (B_4C) or silver-indium-cadmium alloy (Ag-In-Cd) as specified on record COR_RT, (7) Inconel, and (8) an electric heating element material, also specified on record COR_RT. Each component may be composed of one or more of these materials. For example, the cladding component may be composed of zircaloy, Inconel (to simulate grid spacers), and ZrO_2 (either initially present or calculated by the COR package oxidation models). The melting and candling of materials results in the possibility of any or all materials being found in a given component. The heating element material is intended for use in analysis of electrically heated experiments. Its use requires that the user modify subroutine ELHEAT to provide a calculation of the associated heating power in all cells containing the material.

Zircaloy is a single material in the COR package, with no distinction made between zirconium and the zircaloy alloying elements. Steel and steel oxide are also each modeled as single materials within the COR package, but the user must specify the fractions of iron, nickel, and chromium in the steel so that oxidation can be properly treated and the right amounts of each species can be transmitted to the Cavity (CAV) package during debris ejection. Inconel is treated as a single material, and currently it has the same properties as steel (and is ejected as steel), but it is not permitted to oxidize. Properties of the materials are obtained from MELCOR's Material Properties (MP) package. In MELCOR versions after 1.8.4, the user was given increased flexibility to use properties other than those of the default materials.

The user defines several geometric variables to further describe the cells and components. Representative dimensions for the intact components are specified on record COR_GP, and elevations and lengths (heights) for each cell are input on record COR_ZP. Equivalent diameters for each component in each cell for use in various heat transfer correlations also must be specified on record COR_EDR. Cell boundary areas for intercell radiation (both axially and radially) are defined by the user on record COR_BFA. Initial volumes of components and the empty CVH fluid volume are calculated based on user input for component masses and cell flow areas (records COR_CCM and COR_BFA) and are then tracked during core slumping and flow blockage calculations.

Several additional geometric variables are input on record COR_VP to describe the dimensions of the lower plenum / lower head. The radius of curvature of the lower head determines the surface areas for the lower head as well as the volumes of cells that intersect with the lower head. In addition, the inside radius of the pressure vessel is input to determine those surface areas along the cylindrical part of the vessel included in the lower head representation (Section 1.1.2).

For each intact component in each cell, a surface area is input by the user on record COR_SA for convection and oxidation calculations. (The single surface area value input for a canister is multiplied by elements in sensitivity coefficient array C1501 to obtain values for each side of each canister component to communicate separately with the

channel and bypass control volumes.) For particulate debris, a surface area is calculated from the total mass and a user-defined particle size input on record COR_EDR. (For oxidation of particulate debris, separate Zircaloy and steel surface areas are calculated.) The effects of conglomerate debris on component surface areas are factored into the heat transfer, oxidation, and candling calculations; this model is described in Section 3.1.6.

As discussed later in Sections 2.3 and 2.5, CVH package supplies fluid conditions for use by the COR package in calculating heat transfer and oxidation rates, which are then multiplied by the timestep and passed back to the CVH package as energy and mass sources or sinks. The nodalization for the reactor vessel used in the CVH package is typically much coarser than that used in the COR package, but finer CVH nodalizations can be used to simulate in-vessel natural circulation. The COR nodalization applies only to those components in the core and lower plenum treated by the COR package and is independent of the CVH nodalization, with some restrictions imposed.

Figure 1.3 gives a 2-D representation of the interface between the COR and CVH packages, but more accurate depiction of the relationship between the two nodalizations requires a 3-D illustration, shown in Figure 1.4. Each COR cell interfaces with a CVH control volume (input on record COR_RBV) representing the primary flow (channel volume), which provides boundary conditions for most core surfaces. Typically, many core or lower plenum cells interface with the same control volume. For BWRs, a separate CVH control volume (shown behind the channel volume in Figure 1.4) may also be specified for COR cells on record COR_RBV to represent the interstitial space between fuel assemblies (bypass volume). The outer canister surfaces and the supporting and nonsupporting structure surfaces, as well as the surface of any particulate debris in the bypass of a BWR, all communicate with this bypass control volume if it is distinguished from the channel control volume. In MELCOR 2.x, the total number of control volumes interfaced to the COR package is no longer a required input quantity. The only restrictions between CVH and COR nodalizations are that control volumes occupy a rectangular grid of core cells and have boundaries lying either on cell boundaries or entirely outside the core nodalization.

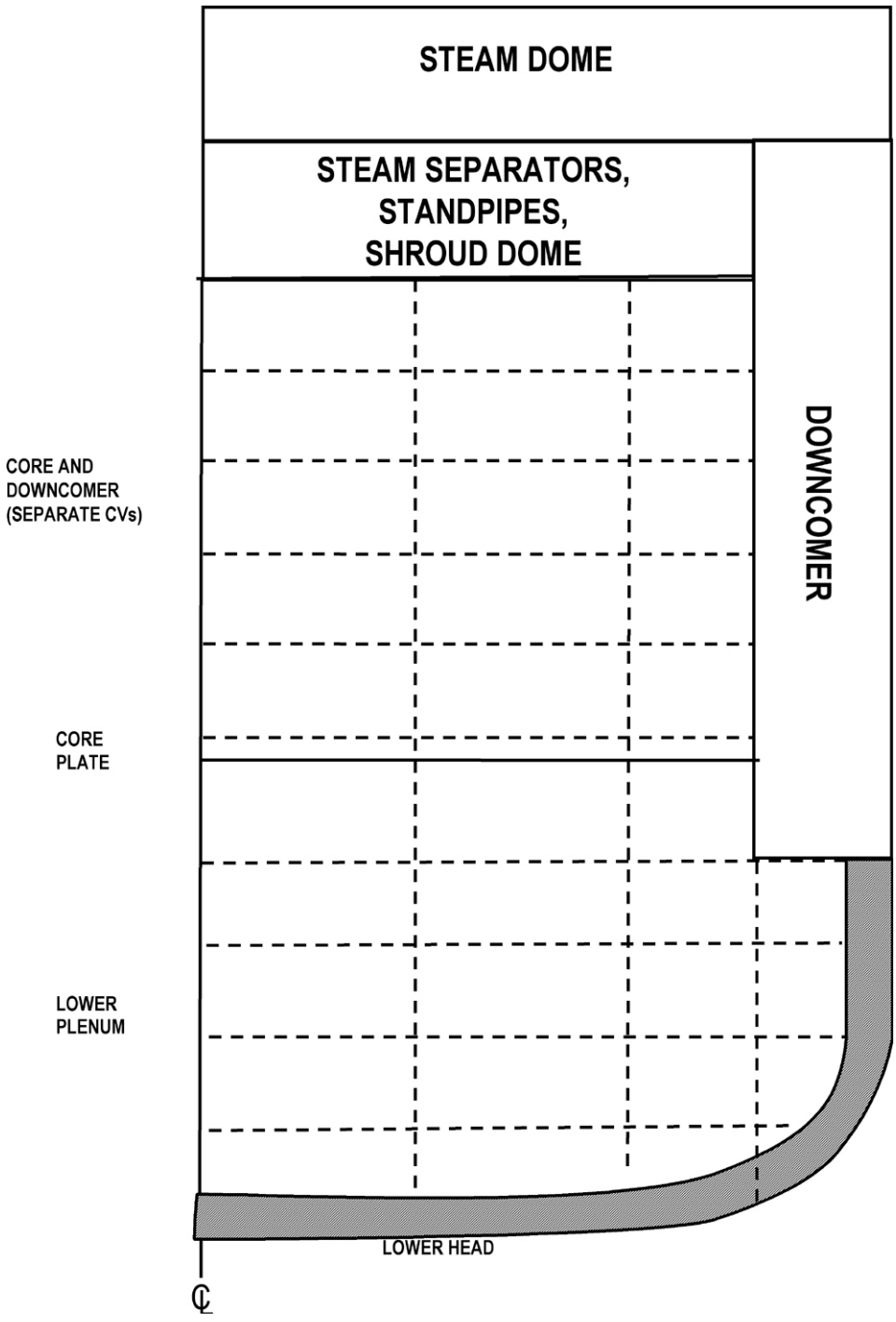


Figure 1.3 Typical COR-CVH Nodalization Interface (2D)

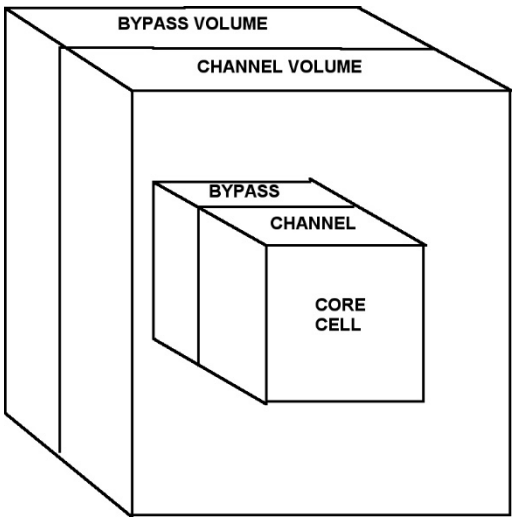


Figure 1.4 Typical COR-CVH Nodalization Interface (3D)

1.1.2 Lower Head

The basic elements of the COR package lower head heat transfer model are the lower head hemisphere; head penetrations, such as instrumentation tubes or guide tubes; the layer(s) of debris or molten materials resting on the lower head; and the CVH heat sink available in the reactor cavity. The lower head modeling in the MELCOR COR package has been substantially modified in MELCOR 1.8.6, and its application has been extended to include that portion (if any) of the cylindrical reactor vessel that is below the bottom of the baffle plate in a BWR or the lower plate in a PWR (HLST). This replaces the previous use of the HS package to provide a radial boundary condition for that portion of the core model that is below the bottom of the core barrel. As a result, the input for these lower levels must not include radial or axial boundary heat structures on the COR_RP or COR_ZP input records. Instead, a character string, NO, must be specified on the COR_ZP input record for these axial levels below HLST to indicate the absence of such a structure.

The new modeling allows a more general representation of the lower head for hemispherical, truncated hemisphere, cylindrical, or spheroidal vessel head geometry (Figure 1.5 and Figure 1.7). A truncated hemisphere may be defined by providing a hemispherical radius, RVLH, larger than the cylindrical vessel radius, RVESS (Figure 1.6). A generalized formulation based on an oblate spheroid geometry also exists and may also be truncated. This formulation is used by specifying a semi-major axis ELLIPA and a semi-minor axis ELLIPC and is truncated if ELLIPA is larger than RVESS. A detailed derivation of the necessary geometric features for the lower head nodalization is presented in the Spheroidal Lower Head Derivations section at the end of this guide (Appendix B) for the oblate spheroid. Since the geometry of the oblate sphere is equivalent to the hemisphere when the two axes equal each other, without loss of

generality, the remainder of this section describes the case assuming a single, hemispherical radius RVLH. As in the existing model, the outer surface communicates with the cavity region(s) while the inside surface communicates with internal coolant, penetrations, and debris. Both the region below the curved lower head and that outside the core barrel (essentially the downcomer region in a PWR or BWR) are formally included in the COR nodalization but are inaccessible to the COR package.

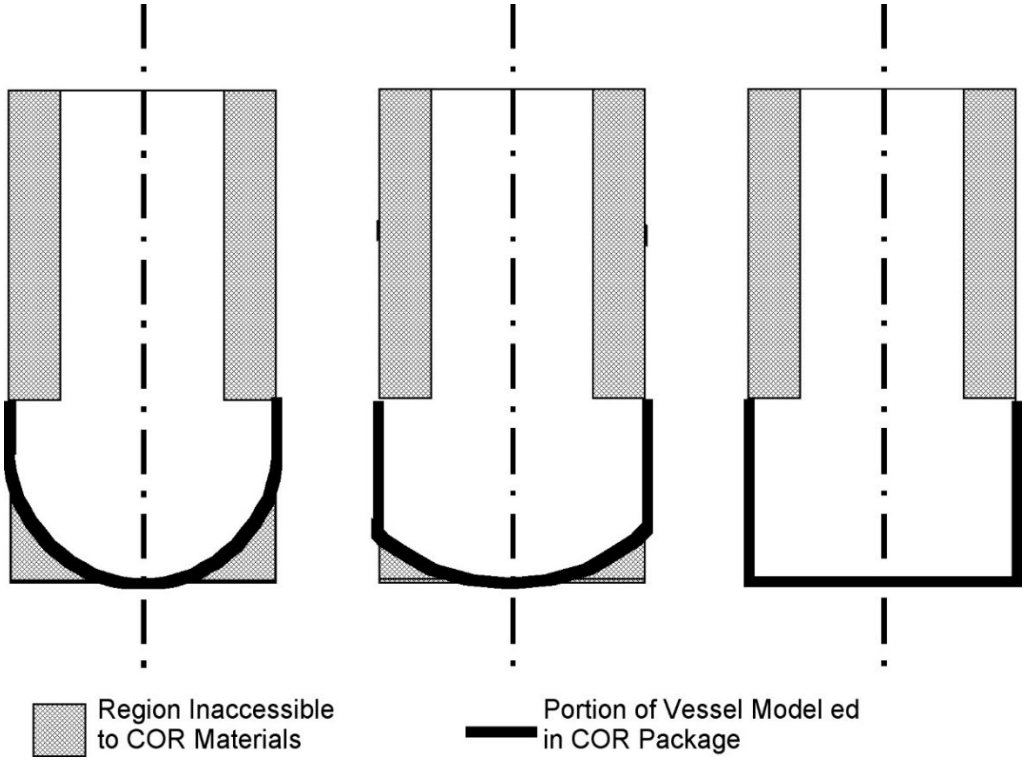


Figure 1.5 MELCOR 1.8.6 lower plenum geometric representations

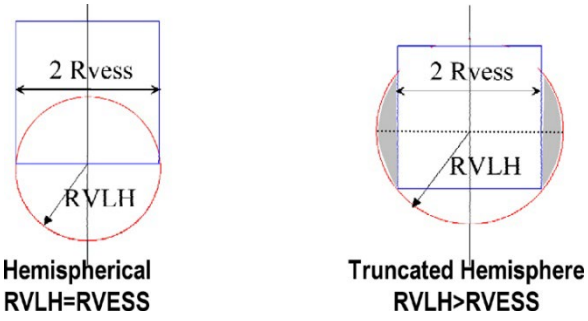


Figure 1.6 Specification of RVLH for a truncated hemisphere

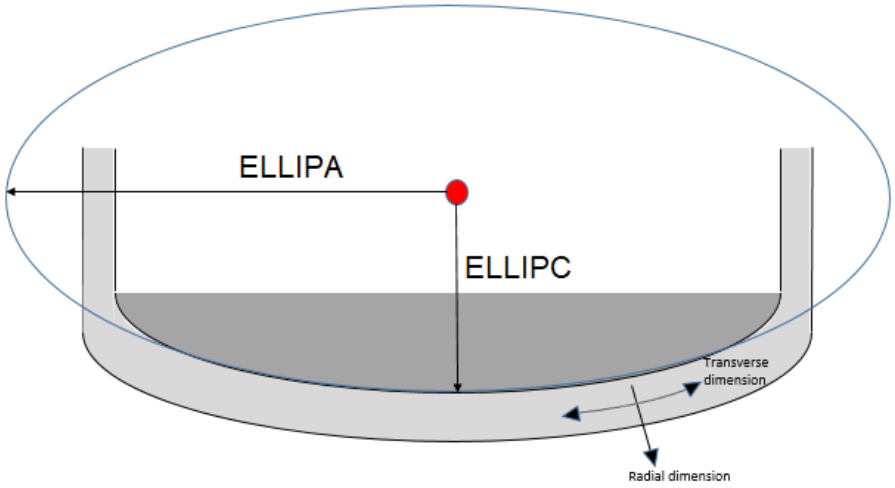


Figure 1.7 Basic geometry of the spheroidal lower head.

In the hemispherical (or truncated hemispherical) model (Figure 1.8), the active bottom cell need not be at the lowest elevation for all radial rings as was the case for the version 1.8.5 COR model. Instead, the elevation of the lowest active cell may be greater at positions radially closer to the outer rings. A consequence of this modification is that material in the lowest active cell in the outer ring settles downward into vacant inner cells. Similarly, MELCOR does not calculate radial spreading of the debris from the inner rings into the outermost ring until debris has accumulated in the bottom elevation of the outermost ring. This effectively reduces the contact area between debris in the lower plenum and in the lower head.

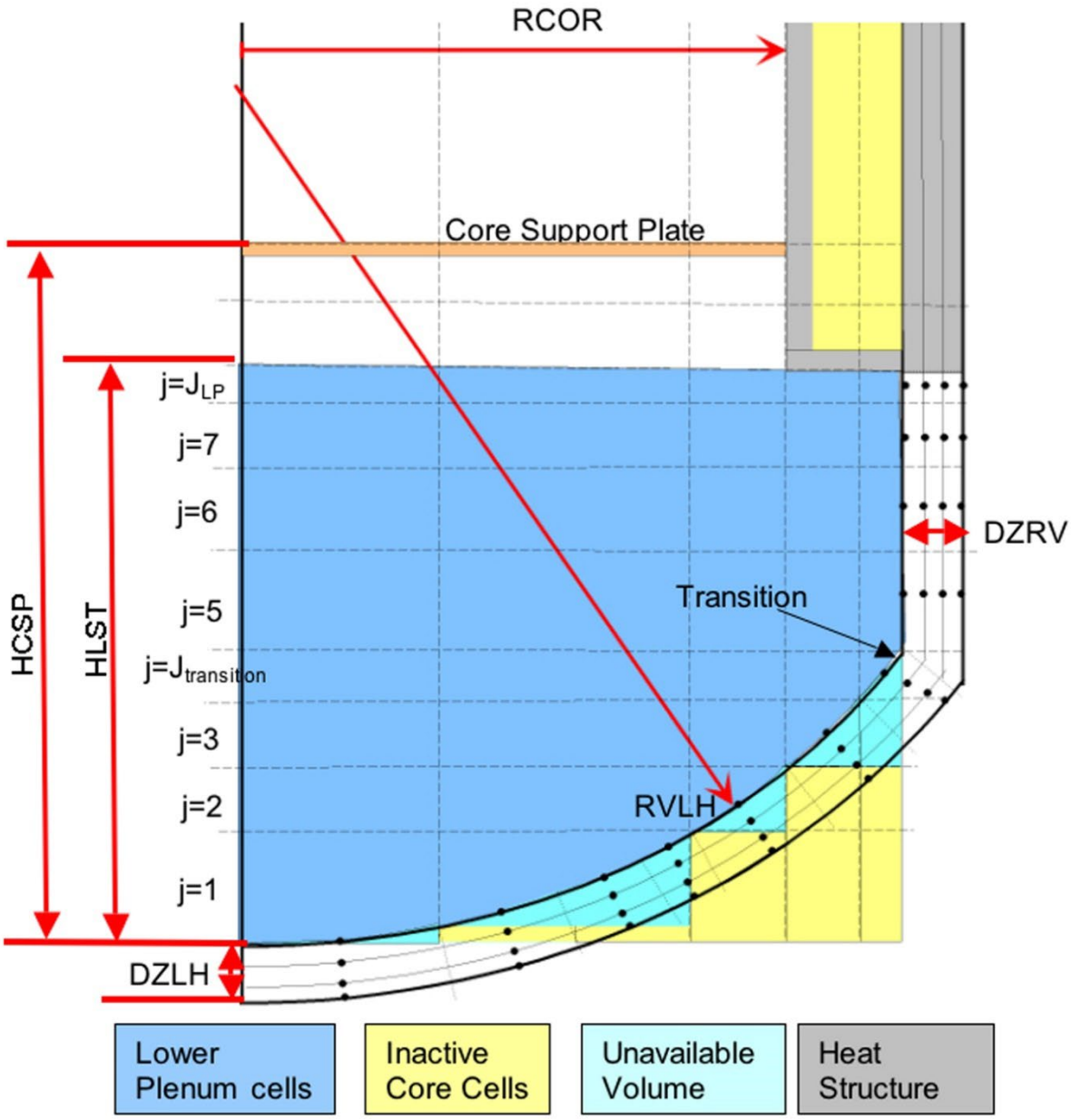


Figure 1.8 Hemispherical geometry of the lower plenum

The lower head is divided into segments specified by the user, and the local through-wall thickness is divided into a number (defined by entry NLH on record COR_LH) of finite-difference temperature nodes for treating conduction. Both the composition and mesh spacing in the lower head may be defined by the user (by default the lower head is divided into NLH-1 equal mesh layers of stainless steel, each of thickness DZLH/(NLH-1)). The NLH temperature nodes are located at the mesh layer boundaries. Heat transfer from hot debris to the inner surface of the lower head is modeled parametrically, with a user-

specified constant heat transfer coefficient. Heat transfer from the outer surface of the lower head to the reactor cavity is treated parametrically if the cavity is dry, using a constant, user-adjustable heat transfer coefficient with a default value of 10 W/m^2-K, or with a simple downward-facing boiling model if the cavity is flooded.

MELCOR 1.8.6 and 2.1 allow the specifications for multiple segments to interface with a COR cell to allow a more detailed calculation of the temperature profile in the lower head. In addition, it calculates the elevation of the upper surface of a molten pool in the lower plenum, and molten pool heat transfer occurs only for those segments below the upper surface (i.e., heat transfer areas are dependent on the calculated pool height). Also, temperatures in the lower head structure (which may include the lower vessel cylinder) are calculated from a semi-implicit two-dimensional heat transfer calculation. These changes were implemented to generate a more accurate calculation of the temperature profile in the vessel with possible peaked temperature profiles.

Because no lower head segment is permitted to interface with more than one core cell, there must be a segment boundary at each point at which the lower head crosses a boundary between radial rings or axial levels. In addition, one segment boundary must correspond to the location of the transition to cylindrical geometry. These requirements determine the minimum number of required segments in the curved (hemispherical geometry) or horizontal (cylindrical geometry) section of the lower head. As previously discussed, the user can specify additional segments in the curved or horizontal portion of the lower head for a total of NLHTA sections. In addition, if J_{LP} corresponds to the upper level in the lower plenum (from HLST), and $J_{transition}$ corresponds to the elevation of the transition from hemispherical to cylindrical geometry, then the total number of segments for the lower head is deduced from

$$\begin{aligned} NLHT &= NLHTA + J_{LP} - J_{transition} + 0 \quad then\ transition\ occurs\ at\ an\ axial\ elevation \\ &= NLHTA + J_{LP} - J_{transition} + 1 \quad otherwise \end{aligned} \tag{1-1}$$

MELGEN expects to read a COR_LHD input record for NLHT segments and issues a diagnostic message if the deduced number is incorrect. If the bottom of the head is curved, it is often convenient to begin with input for the NLHTA segments and allow MELGEN to determine how many more must be added. In order to simplify the task of defining consistent input, a new input record series has been added to define the radii corresponding to the ring boundaries. There must be a lower head segment corresponding to each of these radii.

The thickness of bottom curved lower head (DZLH) may differ from that of the cylindrical vessel (DZRV), and the transition in thickness may take place at either the radius of the core (typical of a BWR) or at the radius of the cylindrical vessel (typical of a PWR). The thicknesses DZLH and DZRV are specified by the user on the COR_VP record.

In MELCOR 1.8.5, a one-dimensional solution was obtained to determine the temperature profile through the vessel wall, and lateral conduction along the vessel was ignored. For

MELCOR 1.8.6 and 2.1, a lateral conduction calculation is performed and the heat transfer to or from each node is used as a heat source in the implicit through-wall heat transfer calculation. Even though both the lateral and the through-wall calculations are implicit, the two calculations are essentially independent, resulting in a “semi-implicit” conduction calculation. The two-dimensional conduction calculation within the lower head is discussed further in Section 6.1.

The calculated temperature profile through the lower head is used in a mechanical response model that determines stress and strain in the lower head to predict creep-rupture failure. Creep (plastic strain) is calculated from the Larson-Miller parameter and a life-fraction rule.

Figure 1.9 illustrates the lower head nodalization for a single segment. For each lower head segment, the user can define up to three representative types of penetrations (only one is shown in the figure), specifying the total mass and heat transfer areas associated with each penetration type and the initial effective diameter of the opening created when a penetration fails. Each penetration communicates thermally with the top lower head node, the debris, and the molten pool components. The total number of penetrations in all rings is a required input quantity on record COR_PEN, if such penetrations exist. There should be no duplication of mass or surface area between penetrations and structures modeled as ordinary core components in the first axial level of core cells; the user may divide such structures between penetrations and supporting or nonsupporting structure arbitrarily, but the thermal modeling interface is somewhat indirect. The user should also realize that penetration masses are not currently added to core/lower plenum debris masses and cannot be ejected from the reactor vessel.

Finally, MELCOR 1.8.6 and 2.1 allows for the possibility that different segments of the lower head interface with different volumes in the containment nodalization as the through-wall heat transfer is calculated for each lower head segment. A control volume is specified by the user for each segment on the COR_LHD record.

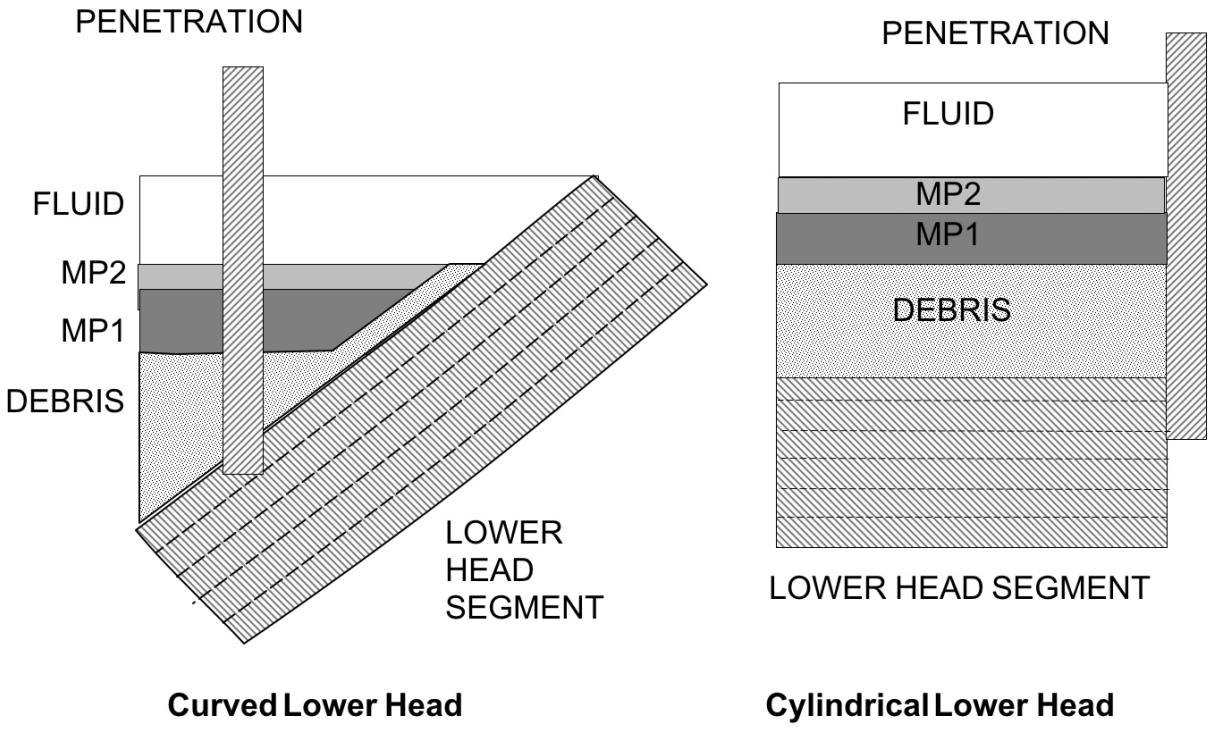


Figure 1.9 Lower head nodalization (one segment)

1.1.3 PWR Core Outer Periphery

For a PWR, the capability of MELCOR 1.8.6 for modeling the core shroud, formers, and bypass region has been greatly improved. Two new components for shroud SH and former FM (Figure 1.10), have been added to facilitate these models. These new components can be defined in the outer active ring (the outermost occupied ring above the bottom of the core support barrel) and are permitted for a PWR only. The shroud component has the property that it can fail, allowing debris to relocate into the peripheral bypass volume between the shroud and the core support barrel, as was observed in the TMI-2 accident. Debris in the bypass volume can then be relocated downward, supported by formers until the formers fail. A distinct CVH volume may be specified as the bypass volume for such cells and may provide fluid boundary conditions for FM, the outside of SH, and the inside of the radial boundary heat structure.

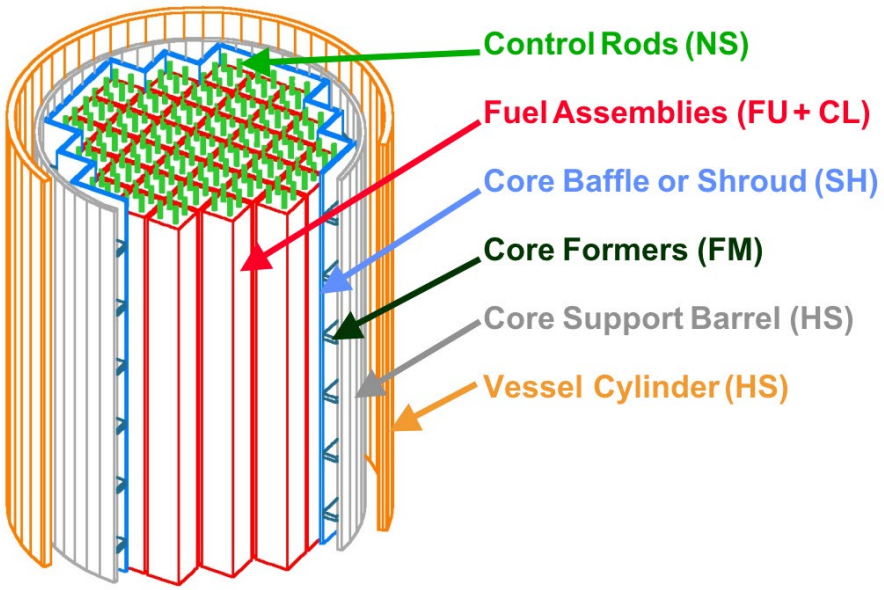


Figure 1.10 PWR bypass region

1.1 Calculation Framework

All thermal calculations in the COR package (both in the core/lower plenum components and in the lower head) are done using internal energies of the materials (i.e., temperature is a derived variable calculated from the material internal energies; initial temperatures are defined on record COR_CIT). The mass and internal energy of each material in each component are tracked separately to conserve total mass and energy to within machine round-off accuracy.

The COR package uses an explicit numerical scheme for advancing the thermal state of the core, lower plenum, and lower head through time. To mitigate numerical instabilities, a subcycling capability has been developed to allow the COR package to take multiple timesteps across a single Executive (EXE) package timestep. All energy generation, heat transfer, and oxidation rates are evaluated at the beginning of a COR package subcycle based on current temperatures, geometric conditions, and an estimate of the local fluid conditions (calculated by the COR package dT/dz model to reflect the temperature variation within a control volume containing many individual COR cells). The net energy gain (or loss) across the subcycle is determined for each component by multiplying these rates by the COR package timestep.

The temperature change of most components is limited to a user-input maximum; if the calculated temperature change for a component is greater than this limit, the COR package subcycle timestep is reduced accordingly, but not lower than the minimum timestep input by the user for the COR package. Components with a total mass below a

critical minimum are not subjected to this limit. If the energy input to any fluid volume changes from previous values in such a way as to possibly result in numeric instability between the COR and control volume packages, the system timestep may be cut immediately or a reduction may be requested for the next EXE timestep. The various timestep control parameters may be specified by the user on record COR_DTC and using sensitivity coefficient arrays C1401 and C1502 (see COR Package Users' Guide).

At the end of a COR package timestep, after the thermal state of the core has been updated by the heat transfer and oxidation models described in Section 2, relocation of core materials and debris formation are calculated by the core degradation models described in Section 3. Molten portions of intact structures are transferred to the conglomerate debris associated with the structure. Liquefaction of intact structures caused by eutectic reactions between materials within the structure and dissolution of intact structures by existing molten material within the core cell are calculated, if the materials interactions model has been activated. Molten materials are relocated downward by the candling model, and molten pool components are formed when local blockages are detected. In the absence of a local blockage, the molten pool relocates into the interstitial volume of particulate debris and be transformed into conglomerate, thereby equilibrating with the particulate debris. Intact components are converted to debris if various debris formation criteria are met.

Downward relocation of particulate debris from one cell to a lower one by gravitational settling is generally modeled as a logical process, and relocation is completed over a single timestep with consideration given only to constraints imposed by the porosity of the debris, the availability of free (open) volume to hold it, and support by structures such as the core plate. (These constraints are not imposed on molten debris, which always relocates to lower regions unless the path is locally blocked.) However, numerical limits are imposed to ensure that the mass relocated goes to zero in the limit of small timesteps, and a rate limitation is imposed for the falling debris quench heat transfer model. In MELCOR 1.8.5, debris in the bypass of a BWR is distinguished from that in the channel. In core cells containing a canister, the downward relocation of particulate or molten debris can be blocked separately in the channel and in the bypass. After the canister has failed, debris in the channel and the bypass are mixed and equilibrated. As long as the canister is intact, the majority of the particulate debris in the bypass of a BWR are the remnants of control blades. Most of the space available to it is in the bladed bypass region, adjacent to canister component CB. Therefore, the existence of CB is taken as the criterion for the separation of the particulate debris in the bypass from that in the channel.

Reactor components such as control rods and blades may be supported from above or below, with parametric models for failure based on the temperature and the remaining thickness of the structural metal. Either load-based structural models or simpler parametric models may be used for the failure of components, such as the core plate and the Control Rod Guide Tubes (CRGTs) in a BWR.

Gravitational leveling of molten pools and debris beds across the core rings is calculated with a user-adjustable time constant. In a BWR, this leveling is blocked by the presence of intact canisters, so that no leveling is possible until any distinction between the debris in the channel and that in the bypass has disappeared. Debris beds are completely leveled; the angle of repose is not considered. Whenever mass is relocated or debris formed, material energies in the new or changed components are re-evaluated and the temperature updated to maintain thermal equilibrium, and any relevant geometric variables are recalculated to reflect the change in geometry.

2. Heat Transfer and Oxidation Models

This section describes the models implemented in the COR package to treat various modes of heat transfer and oxidation within the core and lower plenum; lower head heat transfer models are discussed separately in Section 6. Radiation, conduction, and convection are covered in Sections 2.1, 2.2, and 2.3, respectively, and oxidation is covered in Section 2.5. Section 2.5 describes the dT/dz model used by the COR package to provide approximate local (core cell) fluid temperatures and gas compositions within the possibly larger CVH control volume. Fission power generation in ATWS accident sequences (and in some experiments) is covered in Section 2.7.

Most of the constants (including exponents) used in the correlations described in this section have been implemented as sensitivity coefficients, thus allowing the user to change them from the default values described in this document, if desired. Sensitivity coefficients are grouped into numbered arrays, Cnnnn(k), where 'nnnn' is an identifying number that refers to a set of related coefficients, such as the several constants appearing in a single correlation (see the MELGEN/MELCOR Users' Guide). Appendix A gives a table of sensitivity coefficients used in the COR package and their default values. Unless otherwise noted, all variables and dimensional constants are in SI units, in conformance to MELCOR coding conventions.

2.1 Radiation

Thermal radiation among components within COR cells, across cell boundaries, and from components to steam is modeled as exchange of radiation between pairs of gray surfaces with an intervening gray medium; the model is constructed following the description provided in Kreith [1]. The radiosity, J_I, is defined as the total energy flux leaving the i-th surface (i = 1 or 2 in this model), both reflected and emitted:

$$J_i = (1 - \varepsilon_i)\, G_i + \varepsilon_i E_{bi} \qquad (2\text{-}1)$$

where

ε_i = emissivity of surface i,

G_i = radiation flux incident on surface i, and

E_{bi} = blackbody emissive power of surface i, σT_i^4 .

The net heat transfer rate from the i-th surface is the difference between the radiosity and the incident radiation, multiplied by the area of surface i, A_i:

$$q_i = A_i(J_i - G_i) \tag{2-2}$$

Combining Equations (2-1) and (2-2) gives q_i in terms of the radiosity and blackbody emissive power:

$$q_i = A_i \frac{\varepsilon_i}{1-\varepsilon_i} \ (E_{bi} - J_i) \tag{2-3}$$

The net heat transfer rate from surface i to surface j is given in terms of the surface radiosities by the expression

$$q_{ij} = A_i F_{ij} \ \tau_{ij} \ (J_i - J_j) \tag{2-4}$$

where

F_{ij} = geometric view factor from surface i to surface j and

τ_{ij} = geometric mean transmittance between surfaces i and j.

Radiation heat transfer also occurs between each of the surfaces and the steam medium, according to the expression

$$q_{i,m} = A_i \ \varepsilon_m \ (J_i - E_{b,m}) \tag{2-5}$$

where

ε_m = steam emissivity/absorptivity = $(1-\tau_{ij})$ and

$E_{b,m}$ = blackbody emissive power of medium, σT_m^4 .

With the additional requirement

$$q_i = q_{im} + q_{ij} \quad (2\text{-}6)$$

Equations (2-3), (2-4), (2-5), and (2-6) are solved in the COR package to obtain q_i and q_{im} (i = 1, 2) for various pairs of surfaces. The subsections below discuss the calculation of surface and steam emissivities ε_i and ε_m, the geometric view factors F_{ij}, and the implementation logic (i.e., how pairs of surfaces are chosen for multiple cell components that may relocate during a calculation).

2.1.1 Emissivities

The emissivities of core materials were hard-coded in MELCOR 1.8.5. In MELCOR 1.8.6 and 2.1, the correlations have been recoded using sensitivity coefficients to allow the user some flexibility to modify them. The default correlations are essentially unchanged, with one exception: extrapolation of the original correlation for oxidized Zircaloy to a very large oxide thickness could return a negative value; in MELCOR 1.8.6 and 2.1, the correlation is cut off (by default) at an oxide thickness of 1 mm.

The surface and steam emissivities are evaluated by models adapted from MARCON 2.1B [2], an extended version of MARCH 2 [3]. For cladding and canister components, the surface emissivity of Zircaloy is used, which is calculated in these models as a function of temperature and oxide thickness from the equations used in MATPRO [4].

For Zircaloy surfaces whose maximum temperature has never reached 1500 K, the surface emissivity in MELCOR 1.8.5 was given as a piecewise linear function of the oxide thickness as

$$\varepsilon_i = 0.325 + 0.1246\ x\ 10^6 \Delta r_{ox}\ [\Delta r_{ox} < \ 3.88\ x\ 10^{-6}] \quad (2\text{-}7)$$

$$\varepsilon_i = 0.808642 - 50.0\ \Delta r_{ox}\ [\Delta r_{ox} \geq \ 3.88\ x\ 10^{-6}] \quad (2\text{-}8)$$

where Δr_{ox} is the oxide thickness.

In MELCOR 1.8.6 and 2.1, this has been replaced by linear interpolation in the table

Δr_{ox} (m)	ε (-)
0.0	0.325
3.8799999E-06	0.808448
0.001	0.758642

that is equivalent to Equations (2-7) and (2-8) for oxide thicknesses of less than 1 mm.

For surfaces that have reached temperatures greater than 1500 K at some time, the calculated emissivity is then multiplied by the factor

$$f = \exp\left[\frac{1500.0 - T_{i,max}}{300.0}\right] \tag{2-9}$$

where $T_{i,max}$ is the maximum temperature the surface has reached. This factor is limited to a lower bound of 0.325.

Finally, the emissivity is bounded to lie in the range

$$0.0001 \leq \varepsilon_{Zircaloy} \leq 0.9999 \tag{2-10}$$

All constants in the table and in Equations (2-9) and (2-10) are coded as sensitivity coefficients in array C1104.

The surface emissivity of SS and NS components in these models is calculated as

$$\varepsilon_{steel} = min\{max[0.25617 + 0.0003474(T - 616.4833),\ 0.0001],\ 0.9999\} \tag{2-11}$$

This matches the correlation used in MELCOR 1.8.5, adapted from the relationship used in MARCON 2.1B for stainless steel, taken from Reference [5], where it was originally written for temperature in °F. All constants in Equation (2-11) are coded as sensitivity coefficients in array C1102.

In MELCOR 1.8.5, the emissivity of particulate debris was taken as a constant. In MELCOR 1.8.6 and 2.1, the form has been generalized to

$$\varepsilon_{steel} = min\{max[0.9999 + 0.0(T - 1000.0),\ 0.0001]\ \ 0.9999\} \tag{2-12}$$

with all constants in Equation (2-12) coded as sensitivity coefficients in array C1103.

The steam emissivities, ε_m, are evaluated in these models from a table taken from Reference [6], which specifies the steam emissivity versus steam temperature and optical depth (steam partial pressure times mean beam length BL) at the high-pressure limit, as seen in Table 2.1. Mean beam lengths are calculated for each component type based only on representative distances for an intact core geometric configuration and are used in calculating the steam emissivity, ε_m.

Table 2-1 Steam emissivity vs temperature and optical depth [6].

Optical Depth (cm-atm)	Temperature (K)						
	370	600	1000	1500	2000	2500	3000
1.0	0.12	0.09	0.041	0.02	0.01	0.0063	0.004
3.2	0.25	0.195	0.11	0.06	0.03	0.019	0.011
10.0	0.37	0.315	0.23	0.145	0.085	0.053	0.033
32.0	0.47	0.425	0.37	0.29	0.20	0.135	0.086
100.0	0.56	0.533	0.55	0.47	0.365	0.277	0.193
320.0	0.65	0.625	0.70	0.66	0.555	0.47	0.35
1000.0	0.73	0.71	0.82	0.80	0.74	0.65	0.52
3200.0	0.79	0.78	0.92	0.90	0.88	0.78	0.65
10000.0	0.85	0.85	1.00	0.92	0.92	0.85	0.73

Enabled through COR_BL, the mean beam length, BL, for each surface is calculated according to the equations in Table 2-2, which are taken from Reference [7]. The geometric lengths, BLRC, BLCCB and BLCC are described below and may be defined on record COR_BL. The calculation may also be overridden for any given COR surface by a valid specification of sensitivity coefficient C1506.

Table 2-2 Equations used in calculating the mean beam length.

COR Surface	Abbreviation	Equation
Fuel	FU	$BL = 3.5\left(P - 2R_{CL}\right)$
Clad	CL	$BL = 3.5\left(P - 2R_{CL}\right)$
Channel facing canister away from the blade.	CN	$BL = 1.8BLRC$
Channel facing canister adjacent to the blade.	CB	$BL = 1.8BLRC$
Supporting Structure	SS	$BL = 1.8BLCCB$
Core Support Plate	CSP	$BL = 1.8BLCCB$
Non-supporting Structure	NS	$BL = 1.8BLCCB$
Bypass facing canister – no blade	CNB	$BL = 1.8BLCC$
Bypass facing canister adjacent to the blade	CBB	$BL = 1.8BLCCB$
All other surfaces		$BL = 0$

Where:

P = Rod pitch (m).

R_{CL} = Clad outer radius (m).

BLRC = The distance between the outer fuel rods and the canister (Default = 4 mm).

BLCCB = The distance between the canister and the blade (Default = 5.05 mm).

BLCC = The distance between adjacent canister walls (Default = 14 mm).

The beam-length is then used in the calculation of optical depth, scaling appropriately for the appropriate units of (atm.cm):

$$P_{OD} = \mathrm{BL}\frac{P}{1013.25} \tag{2-13}$$

The absorptivity of steam, α_v, is then calculated using:

$$\alpha_v = \varepsilon_m(T_v(IA,IR),OD)\left(\frac{T_v(IA,IR)}{T_s}\right)^{0.45} \tag{2-14}$$

where T_v(IA,IR) is the temperature of the vapor at COR node axial level IA and ring IR, T_s is the temperature of COR surfaces and OD is the optical depth.

2.1.2 View Factors

The view factors F_{ij} used in Equation (2-4) model the effects of surface orientation and are implemented as user-specified parameters. The surface areas A_i used with F_{ij} are the actual component areas for radiation between components within a cell and are cell boundary areas for intercell radiation. Values for the view factors are input by the user as “exchange factors” on record COR_RF. These values should be based on standard expressions for simple geometries, where possible, or on experimental data or detailed radiation calculations for complicated geometries involving intervening surfaces, such as for radiation between “representative” structures in cells containing a number of similar structures (e.g., fuel rod bundles). In the absence of any information to aid in selection of view factors, they should be used as arbitrarily varied parameters to examine the effects of radiation on the course of a calculation. View factors are not dynamic, that is, they do not change as the core degrades; however, they may be changed across a MELCOR restart. Because of reciprocity (i.e., $F_{12}A_1 = F_{21}A_2$), the user-input component surface areas, unmodified by the effects of conglomerate debris, of intact components

are always used with these constant view factors. Only the areas of particulate debris are treated as time dependent.

Figure 2.1 depicts the conceptual framework for radiative heat transfer in MELCOR. The framework is geared toward intact BWR cores, but it is general enough to treat PWR cores, as well as degraded cores and lower plenum radiation. The precise situation represented, with *part* of the control blade and *part* of the fuel rods failed, cannot exist within a single core cell in MELCOR. The figure is for illustration only, as an aid to visualizing which surfaces can radiate to other surfaces under various conditions.

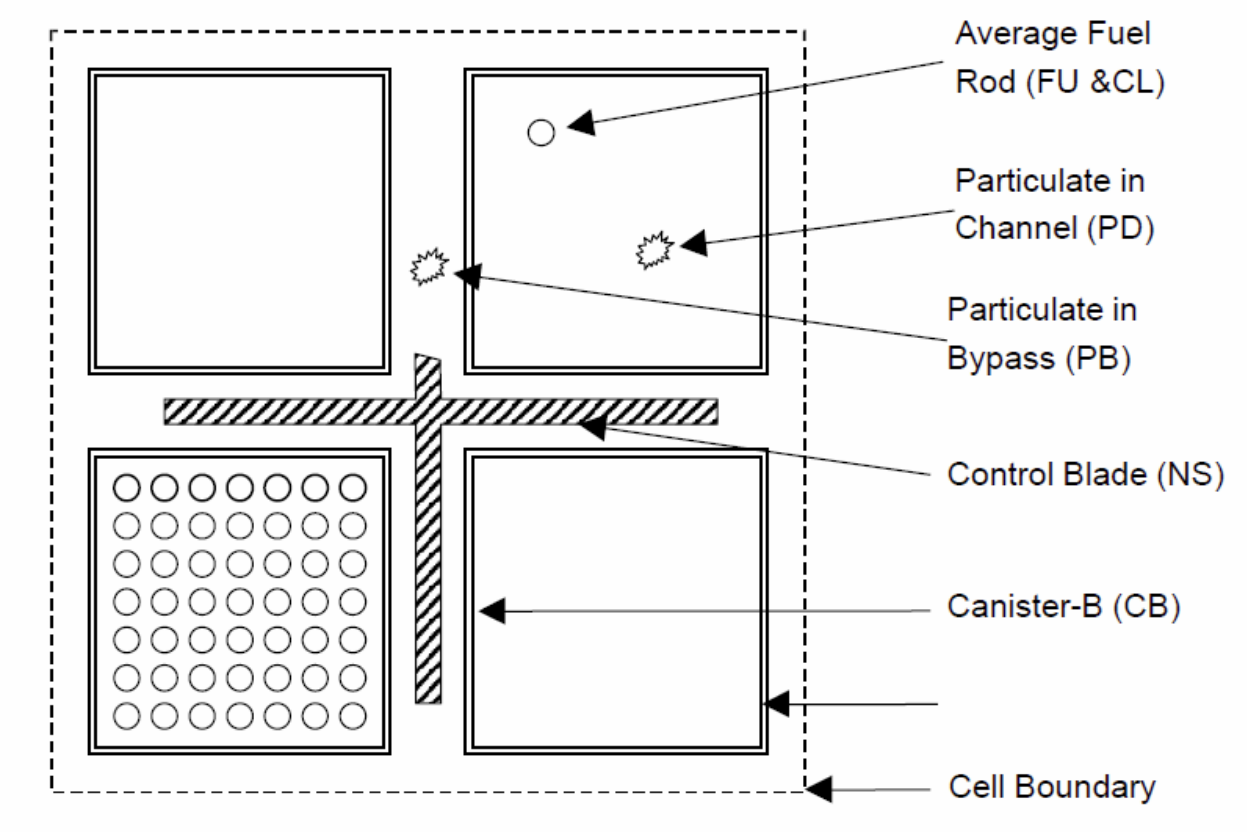


Figure 2.1 Radiative heat transfer framework—BWR cell cross-section

Other structures (SS and/or NS) representing core support structures and control elements are always treated as the innermost component in a cell; these components can radiate to adjacent cells only if no other component exists in the cell. The canister component not adjacent to the control blade (CN) is always treated as the outermost component in a cell; no other cell components can radiate to adjacent radial cells if the canister component CN is present. Particulate debris and molten pool components can exist in the channel (PD, MP1, and MP2) and/or in the bypass region outside the canisters (PB, MB1, and MB2).

For intracell radiation, the user must input two view factors that control radiation between the “average“ fuel rod (cladding component, or perhaps “bare“ fuel) and canister walls (used for both canister components) and between the canister wall (component CB only) and other structures (SS and/or NS):

$F_{cn,cl}$ — view factor for radiation between canister (both components CN and CB) and fuel rods or particulate debris, used with the canister component inside surface areas

$F_{ss,cn}$ — view factor for radiation between any other structure (SS and/or NS) and canister (component CB only), used with the structure surface area

In radiation to or from a fuel bundle or a debris bed, the view of interior surfaces is partially obstructed by outer rods or particles. Whenever radiation is an important mechanism for heat transfer, a temperature gradient is established within the fuel bundle or debris bed. Therefore, the effective temperature difference for radiative exchange with another surface is less than would be predicted from the average temperature of the bundle or bed. This effect can be important in reducing the radiation to a surrounding canister and may be captured by assigning the view factor $F_{cn,cl}$ a value significantly less than unity. The value input for $F_{ss,cn}$, on the other hand, should ordinarily be some value close to unity since the entire control blade surface is directly adjacent to the surface to which it radiates.

For radiation between any other structure (SS and/or NS) and another component within the same cell, SS and/or NS surface area and the view factor $F_{ss,cn}$ are used in Equation (2-4). For radiation between either of the two canister components and the cladding, the canister surface areas and the view factor $F_{cn,cl}$ are used.

As discussed in Section 1.1, particulate debris in the bypass of a BWR (PB) can exist separate from that in the channel (PD) only in the presence of intact canister component CB. Otherwise, it is assumed that the two are mixed and equilibrated. In the following discussion, PD is therefore used to mean <u>all</u> particulate debris (including any in the bypass region of a BWR) in a cell unless intact canister component CB is in that cell.

If PD is present in a cell containing fuel rods, an implicit view factor $F_{cl,pd}$ of 1.0 is used with the cladding (or bare fuel) surface area to model radiation from the rods to the debris. Otherwise, if debris is present in a cell with either canister or other structure components (SS and/or NS), implicit view factors $F_{cn,pd}$ and $F_{ss,pd}$ of 1.0 are used with the canister or other structure surface areas to model radiation between these components and the debris.

If a cell contains both components of a BWR canister (CN and CB) but no fuel rods, the view factor from the inner surface of CN to the inner surface of CB, $F_{cn,cb,}$ is taken as $2^{-1/2}$ (from standard tables, assuming a square canister), used with the area of the inner surface of CN.

For intercell radiation, the user must input two view factors that control radiation in the radial and axial directions:

$F_{cell,r}$ - view factor for radiation radially from one cell to the next outer one, used with cell outer radial boundary area and

$F_{cell,a}$ - view factor for radiation axially from one cell to the next higher one, used with cell axial boundary area.

Intracell radiation is calculated for the outermost ("most visible") components. Again, because of temperature gradients, the effective temperature difference for radiative exchange is less than would be predicted from cell-average temperatures. This effect, which is dependent on the coarseness of the nodalization, should be considered in choosing the values input for these view factors. For radiation from any component to another cell, the appropriate cell boundary area and $F_{cell,r}$ or $F_{cell,a}$ are used in Equation (2-4), although the actual component temperatures are used. For radiation between the liquid pool or lower head and the first cell containing a component, the lower head surface area and $F_{lp,up}$ (defined below) are used in Equation (2-4).

If no components exist in the next outer or higher cell, the radial ring or axial level beyond that is used, until a boundary heat structure is reached. Thus, components in one cell can communicate to nonadjacent cells all the way across the core if there are no components in intervening cells. The boundary heat structures, both radially and axially, specified on records COR_RP and COR_ZP, respectively, receive energy from the outermost cells that contain a component. An additional view factor controls radiation to the liquid pool, if one exists, or to the lower head:

$F_{lp,up}$ - view factor for radiation axially from the lowermost uncovered COR cell to the lower head or liquid pool, used with the lower head surface area.

2.1.3 Intercell Radiation Model (FCELR)

Little guidance is available to aid the user in choosing appropriate values for the intercell radiation exchange factor. This is, at least in part, because the inter-cell values need to be problem-dependent—and even cell-dependent—within the current formulation of the model. Expressed in terms of rod-to-rod radiation between COR cells, the basic difficulty is that portions of rod surfaces more than a few rod diameters from the cell boundary are "seen" with greatly reduced (or zero) differential view factors. This has two closely related consequences:

(1). The appropriate radiation area is the cell boundary area for very large cells and the rod surface area (axially) or perhaps half of it (radially) for very small cells;

(2). The appropriate difference in T^4 for radiation across the boundary is much less than $\left(T_1^4 - T_2^4\right)$ for large cells.

This is the motivation for providing a simple model for view factors that takes into account these two effects. Rather than considering the actual geometry of rod arrays, we consider only a simple model with some qualitative relationship to the "real" world. We assume that the combination of distance between differential surfaces (the factor of r^{-2} in the solid angle subtended) and the obscuring of line of sight by intervening surfaces may together be approximated by a simple exponential. That is, we assume that the fraction of unobscured solid angle remaining visible to a differential surface at depth x is $e^{-\alpha x}$. In consequence, the rate at which solid angle *becomes* obscured—i.e. is intercepted by other differential surface—is $\alpha\, e^{-\alpha x}\, dx$.

In terms of this simple representation, the view factor between a cell of length (perpendicular to the cell boundary) of L_1 and one of length L_2 may be calculated as

$$A_1 F_{12} = \int_{-L_1}^{0} dx_1 A_{cell} \left(\frac{A}{V}\right)_1 e^{\alpha_1 x_1} \int_0^{L_2} dx_2\, \alpha_2 e^{-\alpha_e x_2} \tag{2-15}$$

Here $(A/V)_1$ is the surface area per unit volume in cell 1, and we have assumed that the contents of the two cells may not be identical. In terms of dimensionless variables, Equation (2-15) becomes

$$\begin{aligned} A_1 F_{12} &= A_{cell} \left(\frac{A}{\alpha V}\right)_1 \int_{-\alpha_1 L_1}^{0} dy_1\, e^{y_1} \int_{-\alpha_2 L_2}^{0} dy_2\, e^{y_2} \\ &= A_{cell} \left(\frac{A}{\alpha V}\right)_1 (1 - e^{-\alpha_1 L_1})(1 - e^{-\alpha_2 L_2}) \end{aligned} \tag{2-16}$$

Similarly,

$$A_2 F_{21} = A_{cell} \left(\frac{A}{\alpha V}\right)_2 (1 - e^{-\alpha_1 L_1})(1 - e^{-\alpha_2 L_2}) \tag{2-17}$$

and, by reciprocity,

$$A_2 F_{21} = A_2 F_{21} = AF = A_{cell} F_0 = A_{cell} K (1 - e^{-\alpha_1 L_1})(1 - e^{-\alpha_2 L_2}) \tag{2-18}$$

where

$$K = \left(\frac{A}{\alpha V}\right)_1 = \left(\frac{A}{\alpha V}\right)_2 \tag{2-19}$$

This establishes a relationship between α and cell geometry. Because the volume of each cell is $V_i = A_{cell}L_i$, we have

$$\alpha_i = \frac{A_i}{KL_iA_{cell}} \quad ; \quad \alpha_i L_i = \frac{A_i}{KA_{cell}} \tag{2-20}$$

In various limits, Equation (2-18) becomes

$$AF \to A_{cell}K \qquad \text{for both cells large} \tag{2-21}$$

$$AF \to A_1 \qquad \text{for cell 1 small and cell 2 large} \tag{2-22}$$

$$AF \to \frac{A_1A_2}{KA_{cell}} \qquad \text{for both cells small} \tag{2-23}$$

In consequence of the large-cell limit, it seems that K should have a value of 1.0. However, for now it is left general.

Similarly, the "effective" view factor that accounts for the restricted temperature difference seen is something like

$$(AF)_{eff} = -A_{cell}K \int_{-\alpha_1L_1}^{0} dy_1\, e^{y_1} \int_{-\alpha_2L_2}^{0} dy_2\, e^{y_2} \frac{2(y_1+y_2)}{\alpha_1L_1+\alpha_2L_2} \tag{2-24}$$

where the fraction in the integrand is the fraction of the average difference in T^4 between point 1 and point 2. (We have assumed that T^4 is linear in αx largely because of the relatively simple form that results.) Thus,

$$(AF)_{eff} = -2\frac{A_{cell}K}{\alpha_1L_1+\alpha_2L_2} \int_{-\alpha_1L_1}^{0} dy_1\, e^{y_1} \int_{-\alpha_2L_2}^{0} dy_2\, e^{y_2}(y_1+y_2) \tag{2-25}$$

By virtue of Equation (2-20), this is

$$(AF)_{eff} = -2\frac{(A_{cell}K)^2}{A_1+A_2} \int_{-\alpha_1L_1}^{0} dy_1 e^{y_1} \int_{-\alpha_2L_2}^{0} dy_2 e^{y_2}(y_1+y_2) \tag{2-26}$$

which is

$$(AF)_{eff} = 2\frac{(A_{cell}K)^2}{A_1 + A_2}\{2(1 - e^{-\alpha_1 L_1})(1 - e^{-\alpha_2 L_2}) - \alpha_1 L_1 e^{-\alpha_1 L_1}(1 - e^{-\alpha_2 L_{21}}) - \alpha_2 L_2 e^{-\alpha_2 L_2}(1 - e^{-\alpha_1 L_1})\} \quad (2\text{-}27)$$

which has the more convenient form

$$(AF)_{eff} = 2\frac{(A_{cell}K)^2}{A_1 + A_2}\{[1 - (1 + \alpha_1 L_1)e^{-\alpha_1 L_1}](1 - e^{-\alpha_2 L_2}) + (1 - e^{-\alpha_1 L_1})[1 - (1 + \alpha_2 L_2)e^{-\alpha_2 L_2}]\} \quad (2\text{-}28)$$

Limits are

$(AF)_{eff} \to 4\frac{(A_{cell}K)^2}{A_1 + A_2}$ for both cells large (2-29)

$(AF)_{eff} \to K\frac{A_1 A_{cell}}{A_1 + A_2}$ for cell 1 small and cell 2 large (2-30)

$(AF)_{eff} \to \frac{1}{2}\frac{A_1^2 + A_2^2}{A_1 + A_2}$ for both cells small (2-31)

2.1.4 Implementation Logic

As already noted, the radiation model employs a superposition of pairwise surface-to-surface radiation calculations. The determination of which surfaces “see” which other surfaces is not exhaustive but is intended to ensure that (1) the most important radiation exchange paths are included and (2) no surface is isolated, with each being allowed to radiate to at least one other surface. Assumptions about which terms dominate in a BWR are based largely on Figure 2.1, as qualitatively described above.

When a dominant radiation path for some surface involves an adjacent radial or axial cell, only a single selected surface in that cell is considered. In considering other structure components such as SS or NS, NS takes precedence over SS; For PWR core formers (FM; this component in MELCOR 2.x substitutes what was available as “Other Structures (OS)” in MELCOR 1.8.6), this can only occur in a calculation that does not employ SS or NS components. In the radial case, surfaces in the next cell are considered in the following order: outside of CN, CL, and FU and then inside of CB, NS, SS, FM, and PD. If none of these exists, the next radial cell is considered. In the axial case, the order is CL, FU, inside of CN, inside of CB, NS, SS, FM, and PD. If none of these exists, the next axial cell is considered. Note that particulate debris in the bypass (PB) does not appear in either of these lists. This is because, if it exists independent of particulate in the channel

(PD), CB must also be present, which defines a more important radiation path (in the axial direction) or shield it from external view (in the radial direction).

View factors are used only in combination with areas, as the product $A_1F_{12} = A_2F_{21} = AF$, where the equality is required by reciprocity. In some cases, limits are imposed because direct use of the view factors and areas would result in an implied reciprocal view factor greater than unity.

(1). For radiation exchange between surfaces 1 and 2 that crosses a cell boundary, the product actually used is $F_{\text{cell,x}}$ MIN($A_{cell,x}$, A_1, A_2), where *x* may be r or *a*.

(2). For radiation exchange involving particulate debris PD, the product actually used is F MIN(A_1, A_{PD}), where F is the view factor cited in Section 2.1.2.

The following describes the model implementation in MELCOR 1.8.6 and 2.1.

The logic begins by considering the outer surfaces of an intact canister in a BWR.

1a. That portion of the outer surface of intact canister CB in a core cell that does not see other outer CB surface in the same cell must radiate to NS representing the control blade and/or to PB in the same core cell. Similarly, some portion of the NS surface may radiate to PB. The fraction of the surface of NS and of the outer surface of CB that sees PB is proportional to the fraction *f* of the available space in the bypass that is occupied by PB. AF = MIN (f A_{surf} , $A_{pb}/2$), where surf is ns or cbb.

1b. The remaining portions of these surfaces, A'_{surf} = MAX(A_{surf} - $AF_{surf,pb}$, 0), see each other with AF = MAX(A'_{ns} $F_{ss,cn}$, A'_{cbb}). This formulation, rather than simple use of a factor (1-*f*), accounts for that fact that porosity may result in large holes through the debris bed.

2. That portion of the outer surface of intact canister CN in a core cell that does not see other outer CN surface in the same cell radiates to a component in the next radial cell: AF = MIN($A_{cell,r}$, A_{cnb}, $A_{s,out}$) $F_{cell,r}$.

If fuel rods are present in a core cell in a BWR or PWR, their view factors are considered next. If intact CL is present, only its outer surface is included, with FU-to-CL radiation treated as part of the gap model. The surface of bare FU, however, can radiate to other components.

3a. Fuel rods radiate to the inner surface of canister CB in the same cell, if present ($AF = A_{cb}\, F_{cn,cl}$); otherwise they radiate to other structures (SS or NS) present in the same core cell ($AF = A_{xs}\, F_{ss,cn}$), with the same precedence as in item 1.

3b. Fuel rods radiate to PD in the same core cell (AF = MIN(A_{rod}, A_{pd}) 1), if any is present.

3c. If intact canister CN is present in the same core cell, fuel rods radiate to its inner surface ($AF = A_{cn}\, F_{cn,cl}$); otherwise, they radiate to a selected component in the next radial cell (AF = MIN($A_{cell,r}$, A_{rod}, $A_{s,out}$) $F_{cell,r}$).

3d. Fuel rods also radiate to a selected component in the next axial cell, with AF = MIN($A_{cell,a}$, A_{rod}, $A_{s,up}$) $F_{cell,a}$.

If there is canister is intact but there are no fuel rods in a core cell, the view factors for the inner canister surfaces are considered next.

4. If there are no fuel rods in a cell, the inner surface of canister CN radiates to the next axial level (AF = MIN($A_{cell,a}$, A_{cn}, $A_{s,up}$) $F_{cell,a}$) *unless* there is PD in the same cell. This case is covered later.

5. If there are no fuel rods in a cell, the inner surface of canister CB radiates to the inner surface of CN in the same cell, if present ($AF = A_{cb}\ 2^{-1/2}$), or to a selected component in the next radial cell (AF = MIN($A_{cell,r}$, A_{cb}, $A_{s,out}$) $F_{cell,r}$).

The major view factors for NS or SS in a cell are the outer surface of canister CB or fuel rods, if either or both exist. Canister CB blocks the view of fuel rods. These are covered by items 1 and 3a. Otherwise, the dominant radiative heat transfer for NS or SS involves some other surface.

6a. In the absence of fuel rods *and* canister CB in a cell, NS or SS radiate to the inner surface of canister CN ($AF = A_{xs}\, F_{ss,cn}$) *unless* there is PD in the same cell. This case is covered later.

6b. In the absence of fuel rods *and* both canister components (CN and CB) in a cell, NS or SS partition radiation between any PD in the same cell and selected surfaces in the next axial and radial cells. The fraction going to other cells is taken to be MAX(0, 1-A_{pd}/A_{xS}), where xS represents NS or SS, with NS taking precedence over SS, as previously discussed. AF = MIN($A_{cell,y}$, A_{xs}, $A_{s,out}$) $F_{cell,y}$, , where y is *a* or *r*. Radiation to PD is covered later.

It is assumed that fuel rods in the same cell dominate radiative heat transfer for PD. This is covered by item 3b. If there is no PD in the core cell, other surfaces must be considered.

7a. In the absence of fuel rods, PD radiates to the inner surface of canister CB with AF = MIN(A_{cb}, A_{pd}) $F_{cn,cl}$,or if there is no CB, to some other structures (NS or SS) in the same cell with AF = MIN(A_{xs}, A_{pd}) $F_{ss,cn}$. (As with intercell

radiation, NS takes precedence over SS.) The latter case completes items 6a and 6b.

7b. In the absence of fuel rods, PD also radiates to the inner surface of canister CN (AF = MIN(A_{cn}, A_{pd}) $F_{cn,cl}$) or, if there is no CN, to a selected component in the next radial cell (AF = MIN($A_{cell,r}$, A_{pd}, $A_{s,out}$) $F_{cell,r}$). The former case completes item 4.

7c. In the absence of fuel rods, PD also radiates to a selected component in the next axial cell (AF = MIN($A_{cell,a}$, A_{pd}, $A_{s,up}$) $F_{cell,a}$).

If a water pool is present, radiation is considered between its surface in each radial ring and a selected component in the first nonempty core cell in the same ring above the pool. If there is no water pool, radiation is considered between the lower head and a selected component in the first axial level in each ring. MELCOR 2.x (as well as MELCOR 1.8.5 and 1.8.6 versions) allows additional control of the emissivity and view factor to be used when this component is a supporting structure, through input on COR_PR record. This can aid in modeling radiation to the core support plate.

2.2 Conduction

MELCOR models conduction between components in adjacent core cells, both axially, and radially, and between components in the same cell. Within the portion of a component within a single cell, axial conduction is generally insignificant, except in cases in which there is a steep gradient associated with a quench front. Although modeling of this region was included in previous versions, it has been improved significantly in MELCOR 1.8.6 and 2.1 by the inclusion of a specific reflood quenching model.

The model for axial temperature profiles within components is described in Section 2.2.1, and the model for axial conduction in a component within a core cell is described in Section 2.2.2. The model for the velocity of the quench front is described in Section 2.2.3. The treatment of heat transfer between components in axially adjacent cells is described in Section 2.2.4.

Conduction between particulate debris (PD and/or PB) and other components within the same cell is treated, as described in Section 2.2.7. Cell-to-cell radial conduction is treated for SS representing a continuous plate and for PD and/or PB following failure of any intact canister component in the two cells. In addition, a component in the outermost ring may optionally be designated to conduct heat directly to the boundary heat structures. (This is useful in simulating some experiment geometries.)

Fuel pellets (FU) and cladding (CL) within a core cell are strongly coupled, and the cladding has a relatively small heat capacity. Conduction (and radiation) through the gap between them is therefore treated as a special case, as described in Section 2.2.8.

In MELCOR 1.8.6 and 2.1, convection in molten debris pools is treated by the new molten pool model. If this model is disabled in a calculation, the core package attempts to capture some of the effects of convection within the conduction model, as was done in previous code versions. This is done by increasing the rate of conduction whenever the larger of the two component temperatures used to calculate intercomponent conduction exceeded an assumed "melt" temperature, *TKMIN*. The enhancement factor for axial, radial, and intracell conduction is given by

$$FAC = \max\left\{1.0, \left[TKFAC\,(T_{\max} - TKMIN)\right]^{3}\right\} \qquad (2\text{-}32)$$

where T_{max} is the larger of the component temperatures and *TKFAC* and *TKMIN* are given by sensitivity coefficients C1250 with the default values of .01 K^{-1} and 3200 K, respectively. (The default values give an enhancement factor of 10 when T_{max} exceeds the melting point of UO_2 by about 300 K and are primarily intended to eliminate excessive hot spots when rapid convection/radiation, etc., would clearly preclude their existence.) The enhancement factor for conduction in the lower head uses a hard-wired value of *TKFAC* = 0.01 and the melting temperature of the material between adjacent temperature nodes in the lower head for *TKMAX*.

Particulate debris, penetrations, and molten pool components resting on the lower head conduct heat to the lower head structure. The transient heat conduction from the molten pool to the lower head is calculated by an integral solution to the Stefan model as discussed in Section 2.4.3. Conduction from other components in the lower plenum is discussed in Section 6.1.

2.2.1 Axial Temperature Profiles

Starting with Version 1.8.3, the COR package in MELCOR has contained a model to account for partial coverage of the surface of a core component within a core cell by dividing that component into two sections, each having different temperatures and different coupling to the pool and atmosphere in the core cell. These two sections can be thought of as "cold" and "hot," or "quenched" and "unquenched." Convective heat transfer is calculated separately for them, with the resulting convection terms transferred to the appropriate portion of the fluid (pool or atmosphere).

A detailed energy equation is solved for the average temperature of each core component in each cell. This equation considers the total convective heat transfer and many other effects such as internal heat generation, radiation, conduction to other core components in the cell, cell-to-cell conduction, and the effects of candling, all cast in a form that is manifestly conservative.

The model also solves separate approximate energy equations for the two sections. These include only the internal heat generation, the individual convective heat transfer terms, and a term for conduction between the uncovered and covered portions. The

effect of the terms omitted from the sectional energy equations is corrected on the next timestep, after the full energy equation has been solved. This allows a dynamic evaluation of the distinct temperatures of the covered and uncovered portions of the surface for use in calculating convection and conduction.

Figure 2.2 illustrates the implementation of the reflood and quenching model within the axial extent of a MELCOR core cell. Previously, the interface between the quenched and unquenched sections of a surface was assumed to coincide with the pool surface. That assumption has been eliminated in MELCOR 1.8.6 and 2.1, in which the hot section is permitted to extend below the surface. If the quench front is defined as that location below which the water fully wets the fuel rod and the temperature gradients are small, then this figure shows that three possible regions may be defined: the region below the quench front, the region between the quench front and the surface of the pool, and the region above the pool surface.

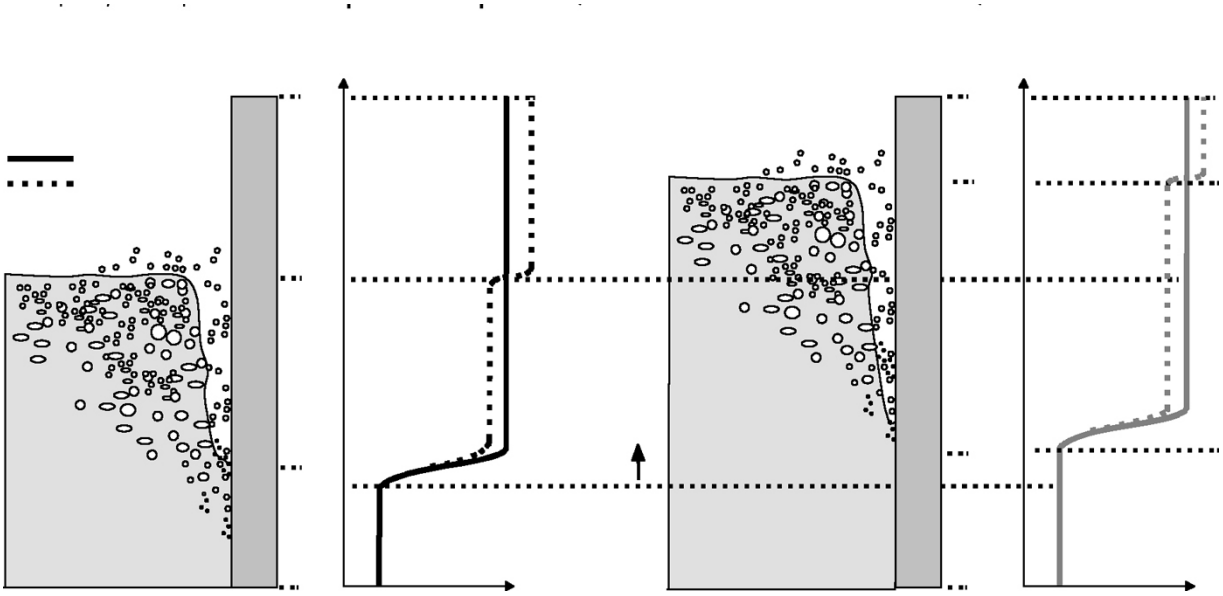


Figure 2.2 Quench front and water level treatment used in the reflood model

It would be possible to implement the model in a way that would evaluate independent surface temperatures for all three regions. However, the full three-region model would have been markedly more difficult to implement, and preliminary investigations suggested that the major distinction was between the quenched (wetted) portion of the surface and the unquenched (unwetted) portion. We therefore chose to implement a slightly simplified approach that allows for three regions of heat transfer but that combines the two portions of the core component above the quench front into a single unquenched region. Conceptual surface temperature profiles for the two models are shown in Figure 2.2. The temperatures shown are axial averages over the length of each region.

The two parts of Figure 2.2 show the independent movement of the water level and the quench front over a timestep advancement, ∆t. The water level is determined from the pool mass, the void fraction, and the pool's presumed axial distribution, as currently treated by the CVH package. As is described in Section 2.2.3, the quench front velocity is independently determined within the COR package from a correlation. When reflood is initiated, the quench front may well advance at a rate slower than the two-phase surface. Later, after the water level has stabilized, it may advance more rapidly, but it cannot advance above the surface. Further, if there is sufficient heat generation in (or heat transfer to) the unquenched portion of the surface, the quench velocity may be negative and the quench front may actually recede.

2.2.2 Axial Conduction in a Component within a Core Cell

The model considers two regions, cold and hot, denoted by "c" and "h" in cases where a quench front exists involving a component within a core cell. However, it is useful in the derivation of the model to temporarily define a third region, denoted by "*", that contains the quench region itself. The phenomena in this region, which is assumed to be of constant but negligible width, are extremely complex. Ultimately, a correlation is used to eliminate the need to represent them in detail.

The rate of change of the average temperature (characteristic of the heat content) of the section of a component between elevations $z_{1,i}$ and $z_{2,i}$ may be written as

$$T_i = \int_{z_{1,i}}^{z_{2,i}} T(z)\ dz \big/ \left(z_{2,i} - z_{1,i}\right) \tag{2-33}$$

(A radial average is assumed in writing this equation.) If the fraction of the component in the section is defined as

$$x_i \equiv \left(z_{2,i} - z_{1,i}\right)/L \tag{2-34}$$

where L is the total length (height) of the core cell, then because $z_{1,i}$ and/or $z_{2,i}$ may be functions of time it can be written as

$$\frac{d}{dt}(x_i T_i) = \frac{1}{L}\left[\int_{z_{1,i}}^{z_{2,i}} \frac{dT}{dt}\ dz + T\left(z_{2,i}\right)\frac{dz_{2,i}}{dt} - T\left(z_{1,i}\right)\frac{dz_{1,i}}{dt}\right] \tag{2-35}$$

The (radially averaged) conduction equation

$$\frac{C}{L}\frac{\partial T}{\partial t} = k\frac{V}{L}\frac{\partial^2 T}{\partial z^2} + \frac{\dot{Q}}{L} - h\frac{A}{L}\left(T - T_f\right) \tag{2-36}$$

may be substituted into Equation (2-35), yielding

$$\frac{d}{dt}(x_i C T_i) = k\frac{V}{L}\left(\left.\frac{\partial T}{\partial z}\right|_{z_{i,2}} - \left.\frac{\partial T}{\partial z}\right|_{z_{i,1}}\right) + x_i\left[\dot{Q} - h_i A\left(T_i - T_{f,i}\right)\right] + \frac{C}{L}\left[T\left(z_{2,i}\right)\frac{dz_{2,i}}{dt} - T\left(z_{1,i}\right)\frac{dz_{1,i}}{dt}\right] \quad (2\text{-}37)$$

In these equations,

- C is the total heat capacity of the component ($\sum m_i c_{pi}$),
- V is its total volume,
- A is its lateral surface area,
- $\dot{Q}$ is the total internal heat generation rate in the component,
- h_i is the heat transfer coefficient,
- T_f is the local fluid temperature, and other symbols have their usual interpretations.

Equation (2-35) is applied to the three regions, “c,”, “h,”, and “*”, and (as in previous versions of the model) the results are simplified based on the assumption that temperature gradients are negligible at 0, L, and also within the entire quenched region.

$$\left.\frac{\partial T}{\partial z}\right|_0 \approx \left.\frac{\partial T}{\partial z}\right|_{q-} \approx \left.\frac{\partial T}{\partial z}\right|_L \approx 0 \quad (2\text{-}38)$$

The simplification process further assumes that temperatures in the region of enhanced heat transfer are not significantly different from that in the cold (quenched) region itself. (This assumption differs from that in early versions of the model and corrects an error in the original derivation [8].)

$$T_{q-} \approx T_{q+} \approx T_c \quad (2\text{-}39)$$

In addition, if x^* is small but essentially constant, the velocities of the moving boundaries may be expressed in terms of the velocity of the quench front, v_q, as

$$\frac{d}{dt}\left(x_{2c}\right) = \frac{d}{dt}\left(x_{1h}\right) = v_q \quad (2\text{-}40)$$

The resulting equations for the three sections are

$$\frac{d}{dt}(x_c C T_c) = x_c\left[\dot{Q} - h_c A\left(T_c - T_{f,c}\right)\right] + \frac{C}{L} v_q T_c \quad (2\text{-}41)$$

$$0 = k\frac{V}{L}\frac{\partial T}{\partial z}\bigg|_{q+} - x^* h^* A\left(T^* - T_{f,q}\right) \tag{2-42}$$

$$\frac{d}{dt}(x_h C T_h) = -k\frac{V}{L}\frac{\partial T}{\partial z}\bigg|_{q+} + x_h\left[\dot{Q} - h_h A\left(T_h - T_{f,h}\right)\right] - \frac{C}{L}v_q T_c \tag{2-43}$$

In general, as shown in Figure 2.2, there are two regions of heat transfer involving the "hot" section of the component. Therefore, the full form of the convective term in Equation (2-43) is

$$x_h h_h A\left(T_h - T_{f.h}\right) = x_u h_u A\left(T_h - T_p\right) + x_a h_a A(T_h - T_a) \tag{2-44}$$

where subscript "*u*" refers to the submerged, but unquenched, portion of the surface; subscripts "*p*" and "*a*" refer to the pool and atmosphere, respectively; and

$$x_h \equiv x_u + x_a \tag{2-45}$$

The heat transfer coefficient in the submerged, but unquenched, region is currently treated as a constant, and is implemented as part of sensitivity coefficient array C1260, with a default value of 125 W/m^2-K. The other heat transfer coefficients are evaluated as described in Section 2.3.

Equation (2-42) simply defines the enhanced heat transfer associated with quenching and with conduction from the hot region to the quench front, and thence to the pool,

$$\dot{Q}_q \equiv x^* h^* A\left(T^* - T_{f,q}\right) = k\frac{V}{L}\frac{\partial T}{\partial z}\bigg|_{q+} \tag{2-46}$$

The sum of Equations (2-41) and (2-43) is a simplified total energy equation.

$$C\frac{d}{dt}(x_c T_c + x_h T_h) = \dot{Q} - \left[x_c h_c\left(T_c - T_{f,c}\right) + x_h\left(T_h - T_{f,h}\right)\right]A - \dot{Q}_q \tag{2-47}$$

$\dot{Q}_q$ is evaluated based on the temperature gradient, and so the heat transfer coefficient h^* does not appear in the conservation equations. (It does appear in the quench front velocity model, described in Section 2.2.3.) This temperature gradient is evaluated from a closed-form solution of Equation (2-36) in the case of a constant quench front velocity,

$$T(z,t) = T_h + (T_c - T_h)\exp\left[\gamma\left(v_q t - z\right)\right] \tag{2-48}$$

Under most conditions, the characteristic length γ^{-1} of the region with a strong temperature gradient is a few centimeters. The conduction area at the plane $z = v_q t$ is V/L, and the total conductive heat flow to that plane can be shown to be

$$\dot{Q}_q \equiv \dot{Q}_c\big|_{z=vt} = -k\frac{V}{L}\frac{\partial T}{\partial z}\Big|_{z=vt} = K(T_h - T_c) \tag{2-49}$$

where

$$K \equiv k\frac{V}{L}\gamma = \left[K_0^2 + \left(\frac{Cv_q}{2L}\right)^2\right]^{1/2} + \left(\frac{Cv_q}{2L}\right) \tag{2-50}$$

and K0 is defined by Equation (2-51)

$$K_0 = k\frac{V}{L}\left(\frac{hA}{kV}\right)^{1/2} \tag{2-51}$$

Using Equation (2-49), Equations (2-41) and (2-47) may be written in implicit finite difference form as

$$\begin{aligned}&\frac{C}{\Delta t}(x_c^n T_c^n + x_h^n T_h^n) = \frac{C}{\Delta t}(x_c^o T_c^o + x_h^o T_h^o) + \dot{Q}\\ &-\left[x_c^n h_c\left(T_c^n - T_{f,c}\right) + x_h^n h_h\left(T_h^n - T_{f,h}\right)\right]A - K(T_h^n - T_c^n).\end{aligned} \tag{2-52}$$

$$\frac{C}{\Delta t}T_c^n = \frac{C}{\Delta t}T_c^o + \dot{Q} - h_c A\left(T_c^n - T_{f,c}\right) \tag{2-53}$$

and may be solved for the new sectional temperatures T_c^n and T_h^n.

As noted earlier, several terms in the full energy equation are missing from Equation (2-52), so that the results can only be viewed as estimates. However, the old temperatures appear in the total energy equation only as the old *average* temperature,

$$\bar{T}^o \equiv x_c^o T_c^o + x_h^o T_h^o \tag{2-54}$$

for which a value calculated using all terms is available. Use of this value in Equation (2-52) helps to couple the two-temperature model to the average temperature. As currently coded, the old sectional temperatures are individually adjusted to account for processes outside of this model. The values actually used are

$$T_i^o = T_i^{n-1} + \left[\bar{T}^o - \left(x_c^o T_c^{n-1} + x_h^o T_h^{n-1}\right)\right] \quad (2\text{-}55)$$

where the superscript "n–1" denotes the value calculated on the previous timestep.

There is a complication for cladding in that the heat source includes heat transfer from tightly coupled fuel pellets. By analogy with the analysis above, the simplified net and cold-section energy equations for the fuel pellets are simply

$$\begin{aligned}&\frac{C_{FU}}{\Delta t}\left(x_c^n T_{c,FU}^n + x_h^n T_{h,FU}^n\right) = \frac{C_{FU}}{\Delta t}\bar{T}_{FU}^o + \dot{Q}_{FU} \\ &-\left[x_c^n\left(T_{c,FU}^n - T_{c,CL}^n\right) + x_h^n\left(T_{h,FU}^n - T_{h,CL}^n\right)\right]h_{gap}A_{FU} \\ &-K_{FU}\left(T_{h,FU}^n - T_{c,FU}^n\right)\end{aligned} \quad (2\text{-}56)$$

$$\frac{C_{FU}}{\Delta t}T_{c,FU}^n = \frac{C_{FU}}{\Delta t}T_{c,FU}^o + \dot{Q}_{FU} - h_{gap}A_{FU}\left(T_{c,FU}^n - T_{c,CL}^n\right) \quad (2\text{-}57)$$

where h_{gap} is the gap heat transfer coefficient and the same cold and hot fractions are assumed to apply to fuel and cladding. The simplified net and cold-section energy equations for the cladding are then

$$\begin{aligned}&\frac{C_{CL}}{\Delta t}\left(x_c^n T_{c,CL}^n + x_h^n T_{h,CL}^n\right) = \frac{C_{CL}}{\Delta t}\bar{T}_{CL}^o + \dot{Q}_{CL} \\ &-\left[x_c^n h_c\left(T_{c,CL}^n - T_{f,c,CL}\right) + x_h^n h_h\left(T_{h,CL}^n - T_{f,h,CL}\right)\right]A_{CL} \\ &+\left[x_c^n\left(T_{c,FU} - T_{c,CL}^n\right) + x_h^n\left(T_{h,FU}^n - T_{h,CL}^n\right)\right]h_{gap}A_{FU} \\ &-K\left(T_{h,CL}^n - T_{c,CL}^n\right)\end{aligned} \quad (2\text{-}58)$$

$$\frac{C_{CL}}{\Delta t}T_{c,CL}^n = \frac{C_{CL}}{\Delta t}T_{c,CL}^o + \dot{Q}_{CL} - h_c A\left(T_{c,CL}^n - T_{f,c,CL}\right) + h_{gap}A_{FU}\left(T_{c,FU}^n - T_{c,CL}^n\right) \quad (2\text{-}59)$$

For all components, a heat flow given by Equation (2-49), representing processes within the quench front, must be added to the heat transfer to the liquid. For fuel rods, there is an analogous term associated with the fuel pellets. As implemented, this heat is deposited directly in the liquid interface. As a result, the generated vapor releases directly to the atmosphere (perhaps first flowing through a vapor boundary layer) rather than forming bubbles that must escape from the pool. The distinction is that vapor created by the quenching process does not contribute to level swell in the liquid pool.

For a completely unquenched component ($x_c = 0$) in a core cell that contains water, immediate quenching is assumed to occur if the excess temperature (T_c—T_{sat}) is less than $dt_{Q,min}$. The value, corresponding roughly to CHF, is programmed as an element of sensitivity coefficient array C1260, with a default value of 40 K. Otherwise, the general equations are solved with T_c^o taken as the temperature of the supporting component in the core cell below (this is usually the same component). This allows a quench front to propagate across cell boundaries.

MELCOR hydrodynamics accounts for bubble separation during boiling separately in each control volume. This can lead to the existence of both pool and atmosphere in each of two or more vertically stacked volumes. Therefore, if several hydrodynamic volumes interface with the COR package, there may be more than one liquid level within the core. To reduce the effects of this artifact of the solution scheme, coverage of components is assumed to be continuous across control volume boundaries if pools occupy more than 0.10 of the lower volume and more than 0.001 of the upper volume. These thresholds for "pool bridging" are implemented in sensitivity coefficient array C1270.

2.2.3 Quench Front Velocity Model

The quench velocity correlation of Dua and Tien [9] was implemented, as recommended by Carbajo and Siegel [10]. The model has been extended, as described in Reference 8, to allow for the unquenching of surfaces with large internal heat sources and the resulting in regression of the quench front. The basic correlation takes the form of

$$Pe = \left[\overline{B}\left(1 + 0.4\overline{B}\right)\right]^{1/2} \tag{2-60}$$

where Pe is the dimensionless quench velocity or Peclet number

$$Pe = u^{*} = \frac{u\delta}{\alpha} \tag{2-61}$$

$\overline{B}$ is related to the wetside Biot number

$$Bi = \frac{h^{*}\delta}{k} \tag{2-62}$$

by

$$\overline{B} = Bi\left(1 - \Theta\right)^2 / \Theta \tag{2-63}$$

where

$$\Theta = \frac{T_h - T_{sat}}{Tsat_{Q,max}} \tag{2-64}$$

is the dimensionless temperature. Equation (2-60) may be thought of as an interpolation between a result based on one-dimensional conduction in thin surfaces (small Bi) and one based on two-dimensional conduction in thick surfaces (large Bi).

In these equations,

U	=	the quench front velocity (m/s),
δ	=	the surface thickness (m),
α	=	the thermal diffusivity (m^2/s),
k	=	the thermal conductivity (W/m-K),
h^*	=	a heat transfer coefficient associated with the quench front itself (W/m^2-K),
T_h	=	the temperature of the unquenched surface (K),
T_{sat}	=	the saturation temperature (K), and
$T_{Q,max}$	=	the maximum temperature against which a quench front can progress (K).

As implemented, δ is evaluated as the volume of the component divided by its surface area. For a thin sheet or cylindrical cell, this is essentially the thickness; for a solid cylinder, it is half the radius; and for spherical debris, it is a third of the radius.

$T_{Q,max}$ is the temperature at which the heat conducted through the steep temperature gradient to the quench front (at or near T_{sat}) is the maximum that can be removed by the enhanced heat transfer processes at that front. The value is dependent on (at least) the pressure, as currently represented by the relation

$$T_{Q,max} = T_{sat} + \Delta T_{Q,max} \tag{2-65}$$

Here, $\Delta T_{Q,max}$ is assumed to be a constant and is coded as a sensitivity coefficient in array C1260. The initially chosen default value was 300 K, based on preliminary calculations of REWET II calculations, as reported in Reference 8. These experiments involved quenching of non-prototypic heater rods from relatively low temperatures. We have found that the higher value of 600 K gives better results in simulating the QUENCH experiments; this value was used in the calculation of ISP45. Consequently, the default value for MELCOR 1.8.6 and 2.1 has been changed to 600 K.

The heat transfer coefficient h^* is also programmed as a sensitivity coefficient in array C1260. Calculations of QUENCH experiments showed that results are relatively insensitive to the value used, and the initially chosen default value of 1.5×10^5 W/m^2-K works as well for QUENCH as it did for the REWET II experiments. The default was therefore used in the calculation of ISP45.

Because of the explicit coupling of the quench model to the CVH flow equations, the numerical solution to the overall thermal hydraulics equations has the potential to exhibit significant numerical “chatter”. To alleviate this issue the current numerical

implementation introduces two adjustments. First, if the pool level is within a small distance of the quench location (specified by another sensitivity coefficient in array C1260), the quench velocity specified by Equation (2-60) is modified and smoothly driven to zero. This eliminates the potential for small undershoot-overshoot fluctuations by the quench model when the quench location is essentially (but not quite) the same as the liquid water level. Second, the application of Equation (2-60) is also modified to introduce a temporal relaxation of its rate-of-change that is based on a time-scale specified by a model coefficient set equal to 1.0 second. Note that this does not change the steady value associated with the model correlation, but rather how quickly the value can change over time. This suppresses fluctuations that are smaller (order of magnitude) than the one second time-scale. This relaxation is only applied to positive quench velocities.

2.2.4 Axial Conduction between Components in Different Core Cells

Axial conduction is computed between like components in adjacent axial cells (e.g., cladding-to-cladding). An exception occurs in the case of molten pool components that constitute a contiguous, convecting molten pool in which the mixing of the convecting pool dominates (Section 2.4.1). However, molten pool material found outside of the contiguous molten pool does not mix with the convecting molten pools and is treated as are all other components. Heat transfer is also calculated between any supporting structure modeling a plate and all components supported by it. In addition, if a given component exists in only one of the two adjacent cells (because of the specification of intact geometry or the failure of the component in one of the cells), conduction is evaluated between the component and particulate debris in the adjacent cell if it exists and if physical contact between debris and component is predicted. Such contact is assumed if the debris resides in the overlying cell where it is presumed to rest on components in the underlying cell, or if the debris completely fills the available volume in the underlying cell so that it reaches the overlying cell. The heat transfer rate axially from one cell component i to component j is given by

$$q_{ij} = K_{eff}\left(T_i - T_j\right) \tag{2-66}$$

where K_{eff} is an effective conductance between the two cells, defined in terms of the individual component conductances by

$$K_{eff} = \frac{1}{{}^{1}/_{K_i} + {}^{1}/_{K_j}} \tag{2-67}$$

$$K_i = \frac{k_i A_i}{\Delta x_i} \tag{2-68}$$

and where

k_i = thermal conductivity of component in cell i (W/m-K),

A_i = axial conduction area of component in cell i (m^2),

Δx_i = axial conduction distance in cell i (m), and

T_i = temperature of component in cell i (K).

For axial conduction, the axial conduction area is considered to be the average horizontal cross section of the component, including the conglomerate,

$$A_i = \frac{V_{tot,comp,i}}{\Delta z_i} \tag{2-69}$$

and the conduction distance considered to be

$$\Delta x_i = \frac{1}{2}\Delta z_i \tag{2-70}$$

where Δz_i is the height of the core cell. If a quench front exists in either cell, the conduction distance is unchanged, but the component temperature is considered to be the temperature of the quenched or unquenched portion of the component, as appropriate.

2.2.5 Radial Conduction

Conduction is calculated between elements of supporting structure (SS) modeling contiguous segments of a plate in radially adjacent core cells. Conduction is also calculated between particulate debris in radially adjacent core cells unless intact canisters block the paths. Conduction is based on Equations (2-66) through (2-68); the conduction area and conduction distance used in Equation (2-68) are

$$A_i = \frac{V_{tot,comp,i}}{V_{tot,cell,i}} A_{rad} \tag{2-71}$$

$$\Delta x_i = \frac{V_{tot,i}}{2\,A_{rad}} \tag{2-72}$$

where $V_{tot,cell,i}$ is the total volume of cell *i* and A_{rad} is the area of the common radial boundary between cell *i* and cell *j*. Equation (2-71) accounts for that fraction of the height of the cell that is occupied by the component. It also introduces a factor of (1 – porosity) into the calculation of the conductance for particulate debris.

2.2.6 Radial Conduction in the HTGR

For the HTGR, radial conduction in the core and between fuel elements is computed. The PBR (pebble bed reactor) uses an effective bed conductance for the pebble bed core. Generally, MX component is allowed to axially and radially conduct between core cells in a PBR reactor type, but FU component can conduct neither axially nor radially because FU – the fueled portion of a pebble fuel element – is "wrapped up" in MX – the unfueled portion of a pebble fuel element. The PMR (prismatic modular reactor) uses an effective radial conductivity for conduction through the fuel compacts, graphite blocks, and gaps between graphite blocks. This effective conductivity governs radial MX component intercell conduction in a PMR reactor type. Axial FU-to-FU conduction is allowed in a PMR reactor type because fuel compacts are in thermal contact with each other (like fuel pins in an LWR context).

2.2.6.1 Pebble Bed Effective Thermal Conductivity

Bed conductance formulations were extensively investigated for use in MELCOR to characterize pebble bed heat transfer. For MELCOR, a general formulation including the important parameters of the pebble bed is preferable.

There are three modes of heat transfer in a pebble bed: conduction through pebbles and fluid, direct conduction through pebbles (assumes that the pebble-pebble contacts are not points), and radiation through the fluid. All three modes are encompassed under the Zehner-Schlunder-Bauer unit cell approach [11,12]. As shown in [13], the main component of the bed heat transfer at high temperature is radiation.

The model adopted for MELCOR is a general Zehner-Schlunder-Bauer formulation which depends on COR cell coolant and fuel conductivities, fuel (graphite) emissivity, and porosity. The formulation from Tsotsas and Martin [14] was used (simplified to remove the terms for secondary effects not needed for HTGR pebble beds). The conduction and radiation terms were retained. The radiation term was modified as per Breitbach and Barthels [15].

The Zehner-Schlunder-Bauer formulation without Knudsen regime or contact conduction effects with the radiation term modified as per Breitbach and Barthels is:

$$k_{eff} = \left(1 - \sqrt{1-\varepsilon}\right)\varepsilon 4\sigma T^3 D_p + \left(1 - \sqrt{1-\varepsilon}\right)k_f + \sqrt{1-\varepsilon}k_c \qquad (2\text{-}73)$$

where

k_c = Effective bed conduction [W/m-K]

k_f = Fluid (gas) conductivity [W/m-K]

ε = bed porosity

T = Solid temperature [K]

D_p = Particle diameter [m]

σ = Stefan-Boltzmann constant [W/m^2-K^4]

The conduction term k_c is given by

$$k_c = \frac{2}{N}\left\{ \frac{B\left(1 + \frac{1}{\Lambda_s\left(\frac{2}{\varepsilon_r} - 1\right)} - \lambda\right)}{N^2 \frac{1}{k_f}} ln\left(\frac{\lambda_P + \lambda_r}{B}\right) + \frac{B+1}{2B}\left(\frac{4\sigma T^3 D_p}{\frac{2}{\varepsilon_r} - 1} - Bk_f\right) - \frac{B-1}{N}k_f \right\} \quad (2\text{-}74)$$

and

$$N = 1 + \frac{1}{\left(\frac{2}{\varepsilon_r} - 1\right)\Lambda_s} - \lambda B$$

where

ε_r = solid emissivity

λ_P = k_s/k_f is the ratio of solid to fluid conductivity

λ_r = k_r/k_f is the ratio of radiative to fluid conductivity

λ = Fluid/solid conductivity ratio = k_f/k_s

B = Shape factor

Λ_s = Solid/radiative conductivity ratio,

$$\Lambda_s = \frac{k_s}{4\sigma T^3 D_p}$$

The shape factor B is determined from the geometry of the Zehner-Schlunder unit cell, and can be approximated by the fit [12]

$$B = C\left(\frac{1-\varepsilon}{\varepsilon}\right)^m$$

where

C = 1.25

m = 10/9

The first term in k_{eff} accounts for radiation through the open part of the unit cell. This differs from the Zehner-Schlunder expression in that the cell walls are assumed to be black, not gray. This is the Breitbach-Bartels modification.

2.2.6.2 Radial Effective Conductivity of Graphite Blocks

The Tanaka and Chisaka expression for a continuous solid system is used for the effective radial conductivity of graphite blocks in the PMR, including the effects of the coolant channels and fuel compacts. The Tanaka-Chisaka expression is

$$k_{eff} = k_s \left[A + (1 - A) \frac{ln\left(1 + 2B\left(k_{por}/k_s - 1\right)\right)}{2B\left(1 - k_s/k_{por}\right)} \right] \quad (2\text{-}75)$$

where

k_{eff} = effective conductivity [W/m-K]

A = $2(1\text{-}\varepsilon)/(2+\varepsilon)$

B = $(1\text{-}\varepsilon)/3$

k_s = thermal conductivity of solid (continuous) material [W/m-K]

k_{por} = thermal conductivity of pores (discontinuous) material [W/m-K]

ε = porosity

For the case of helium gas as the pore material, the pore conductivity should be modified by adding an effective radiative conductivity in parallel with the helium gas conductivity. The radiative conductivity can be written as

$$k_{rad} = 4\varepsilon_r \sigma T^3 D$$

where

k_{rad} = radiative conductivity [W/m-K]

ε_r = emissivity in pores (channels walls)

σ = Stefan-Boltzmann constant [$W/m^2\text{-}K^4$]

D = effective diameter of pores [m]

The effect of the discontinuous material on the continuous material appears in the Tanaka-Chisaka equation as an effective porosity, and only the volume ratios of discontinuous to continuous material are necessary to define an effective porosity. Hence, all that is required are the volume ratios of features in the block.

If the fuel compacts are assumed to have the same thermal conductivity as the graphite, then the ratio of the coolant channel volume to the volume of block + fuel channel defines the effective porosity in the block. The coolant conductivity, the conductivity of graphite, and the porosity define the effective block conductivity. The thermal resistance of the gaps between blocks is then added to come up with an effective radial conductivity. The effective radial block conductivity then can be expressed as

$$k_{er} = \frac{1}{\frac{1}{h_{gap} D_{blk}} + \frac{1}{k_{blk}}} \tag{2-76}$$

where

k_{er} = Effective radial block conductivity (W/m-K)

h_{gap} = Gap heat transfer coefficient (W/m^2-K)

D_{blk} = Effective radial diameter of a block (m)

k_{blk} = Effective radial block conductivity - Tanaka-Chisaka model (W/m-K)

2.2.7 Other Intracell Conduction

As debris accumulates in a core cell and the free volume in the cell vanishes, there is undoubtedly intimate contact between the debris and any remaining intact core components. Therefore, conduction between the debris and core components in the same cell is calculated from Equations (2-66) through (2-68), using

$$A_i = A_j = \frac{V_{tot,PD}}{V_{tot,PD} + V_{free}} \; A_{intact} \tag{2-77}$$

$$\Delta x_{intact} = \frac{V_{tot,intact}}{2A_{intact}} \tag{2-78}$$

$$\Delta x_{PD} = \frac{V_{bed}}{2 \; A_{bed}} \tag{2-79}$$

where

A_{intact} = initial component surface area for the intact component (m^2),

V_{free} = additional volume available to PD (m^3),

V_{bed} = total volume of debris bed (including porosity) (m^3), and

A_{bed} = surface area of debris bed (boundary with other components, as opposed to surface area of debris particles) (m^2),

and a factor of $V_{tot,PD}/V_{bed}$ is included in the conductivity of the particulate debris.

An intact canister (specifically, component CB), separates particulate debris in the bypass from that in the channel. Under these circumstances, intracell conduction from PD is calculated only to fuel rods and both canister components (CN and CB). Conduction from PB is calculated to the outer surface of CB and to the other structures: SS or NS.

2.2.8 Fuel Cladding Gap Heat Transfer

Conduction radially across the fuel pellet and the fuel cladding gap is calculated assuming a parabolic temperature profile across the fuel, negligible cladding thermal resistance, and a constant user-specified gap thickness (input on record COR_GP). Two options are now available on the COR_MS record: one of which assumes that there is negligible cladding thermal resistance and the other applies a thermal resistance term to a zirconium clad and a thin oxide layer.

If it is assumed that the thermal resistance through the clad is negligible, the effective conductance is calculated by combining in conventional fashion the various serial and parallel resistances:

$$\frac{1}{h_{gap}} = \frac{1}{h_f} + \frac{1}{\frac{1}{\frac{1}{h_g} + \frac{1}{h_{CF}}} + h_{rad}} \tag{2-80}$$

where

$$h_f = \frac{4k_f}{R_{FU}} \tag{2-81}$$

$$h_g = \frac{k_g}{\Delta r_g} \tag{2-82}$$

$$h_{rad} = \frac{4\sigma(\frac{1}{2}(T_{CL} + T_{FU}))^3}{\frac{1}{\varepsilon_f} + \frac{1}{\varepsilon_c} - 1} \quad (2\text{-}83)$$

and where

σ = Stefan-Boltzmann constant [W/m^2-K^4]

R_{FU} = radius of the fuel pellet [m]

Δr_g = thickness of the fuel cladding gap [m]

k_g = gap gas thermal conductivity [W/m-K]

h_{CF} = effective heat transfer coefficient from control function [W/m^{2}K]

T_{FU} = fuel bulk temperature [K]

T_{CL} = clad bulk temperature [K]

ε_f = fuel surface emissivity (default value, 0.8) and

ε_c = cladding inner surface emissivity (default value 0.325).

The term representing the thermal resistance of the fuel pellet, $1/h_f$, is combined in series with an effective resistance of the gap. This gap resistance includes radiation across the gap in parallel with the conductive resistance of the gap gas. On record COR_TP, the user may specify an additional resistance, $1/h_{CF}$, calculated via a control function and added serially to the conductive resistance of the gap gas. The fuel and cladding emissivities used to calculate radiation across the gap are stored in sensitivity coefficient array C1101.

The heat transfer rate from the fuel to the cladding is then calculated from the total effective gap conductance using the equation

$$q_{gap} = h_{gap} A_f (T_f^{\tilde{n}} - T_c^{\tilde{n}}) \quad (2\text{-}84)$$

where A_f is the surface area of the fuel pellet and $\tilde{n}$ denotes the projected new-time temperature values.

An option may be set on the COR_MS record card that allows MELCOR to calculate the thermal resistance of the clad and oxide layer. With this option selected the following equation is computed:

$$(hA)_{rod} = \frac{1}{\left(\frac{R_{FU}}{4k_{FU}} + \frac{1}{h_{g,tot}}\right)\frac{1}{A_{FU}} + \left(\frac{R_{CL-\delta r}}{k_{CL-\delta r}}\left[ln\left(\frac{R_{CL-\delta r}}{R_{FU}}\right)\right] + \frac{R_{\delta r}}{k_{ox}}\right)\frac{1}{A_{CL}}} \quad (2\text{-}85)$$

where δr is thickness of the oxide layer and is assumed to be thin. A_{CL} and A_{FU} is the total interfacial surface area of the respective COR surface of the clad and fuel components, respectively. For the purposes of the MELCOR conduction calculation, the mean temperatures are assumed to act as surface temperatures.

The thermal conductivity of the clad, k_{CL}, is calculated as a volume averaged thermal conductivity based on each COR material associated with the clad component, including any zirconium dioxide present. Thus, k_{CL} needs to be corrected to account for the fact that k_{ox} is now taken explicitly from a material property table in the MELCOR database:

$$k_{CL} = \frac{\overline{k_{CL}}(T)\left(V_{\text{int}}(IA,IR) + V_{con}(IA,IR)\right) - \dfrac{k_{ox}(T)m_{ox}(IA,IR)}{\rho_{ox}}}{V_{\text{int}}(IA,IR) + V_{con}(IA,IR) - \dfrac{m_{ox}(IA,IR)}{\rho_{ox}}} \tag{2-86}$$

where V_{int} and V_{con} are the volumes of the intact and conglomerate debris volumes associated with a given COR cell of coordinates (IA,IR), respectively. Bar k_{CL} is the volume average thermal conductivity that has been used in the thin-clad model.

Both equations assume that a steady state temperature profile is valid and that the heat source in the fuel is uniform throughout the COR node. It is further assumed that the conductivity of the material is homogeneous and that the temperature associated with the fuel is an average bulk quantity. The equations are derived assuming average bulk temperatures and it is assumed that the gas gap and zirconium dioxide layers are thin with respect to the thicknesses of the clad and the fuel.

Because of the tight coupling between the fuel and the cladding an implicit treatment is necessary to prevent numerical oscillations for reasonable timesteps. The projected temperatures are found as solutions of the following equations:

$$C_f\left(T_f^{\tilde{n}} - T_f^o\right) = \left(\Delta E_{cond} + \Delta E_{conv} + \Delta E_{rad} + \Delta E_{oxid}\right)_f - q_{gap}\Delta t \tag{2-87}$$

$$C_c\left(T_c^{\tilde{n}} - T_c^o\right) = \left(\Delta E_{cond} + \Delta E_{conv} + \Delta E_{rad} + \Delta E_{oxid}\right)_c + q_{gap}\Delta t \tag{2-88}$$

where C_f and C_c are the total heat capacities of the fuel and cladding, respectively, and the ΔE terms on the right-hand sides are other terms in their respective energy equations. These terms, which account for conduction, convection, radiation, and oxidation, are calculated as described in the corresponding sections of this report. The projected temperatures are used only in evaluating the gap heat transfer.

2.2.9 Treatment of Fuel-Matrix Heat Transfer for HTGRs

For the PBR reactor type, the fuel component is the fueled part of the graphite pebble, and the matrix component is the thin unfueled graphite shell on the surface of the pebble. The fuel-matrix heat transfer coefficient is as described above in Equation (2-80), except that the fuel term $h_f = 4k_f/R_{FU}$ becomes $h_f = 5k_f/R_{FU}$ to account for the spherical geometry.

In the PMR reactor type, the fuel component represents the fuel compact, and matrix represents part of the graphite block webbing associated with a fuel compact and coolant channel. The matrix component is not "thin" (e.g. as clad is treated in an LWR) but is instead treated as a thick cylinder (radial temperature profile is also assumed in the matrix component). A steady-state profile for a cylinder without volumetric power source is assumed. This can be expressed as [16]

$$T(r) = T_0 + (T_1 - T_0)\frac{ln(r/R_0)}{ln(R_1/R_0)} \tag{2-89}$$

where

$T(r)$ = Temperature in cylinder at radius r [K],

T_0 = Temperature at inner radius R_0 [K]

T_1 = Temperature at outer radius R_1 [K]

R_0 = Inner radius [m]

R_1 = Outer radius [m]

Heat transfer in MELCOR COR components is framed in terms of average component temperature. The average temperature for a cylinder can be derived as

$$\bar{T} = T_0 + (T_1 - T_0)\left[\frac{1}{2}\frac{2R_1^2\, ln(R_1/R_0) - (R_1^2 - R_0^2)}{(R_1^2 - R_0^2)\, ln(R_1/R_0)}\right] \tag{2-90}$$

The factor in brackets in Equation (2-90) can be interpreted as the weighting factor between the inner and outer surface temperatures of the cylinder, or alternatively as the fractional location in the cylinder wall of the average temperature. If we define f as

$$f = \frac{1}{2}\frac{2R_1^2 \ln(R_1/R_0) - (R_1^2 - R_0^2)}{(R_1^2 - R_0^2)\ln(R_1/R_0)}$$

then f can be used as a weighting factor for the conductive resistance in the cylinder wall. It can be shown that this varies between 0.5 for a thin cylinder (wall thickness small compared to the average radius) to 1.0 as the outer radius becomes very large compared

to the inner radius. The value of f for a typical pitch and fuel radius for the PMR is about 2/3. In terms of f, the average temperature is

$$\overline{T} = T_0 + (T_1 - T_0) f \quad .$$

This equation can be used to express either T_0 or T_1 in terms of the other surface temperature and the average temperature, allowing the equation for $T(r)$ to be written in terms of the average temperature and one of the two surface temperatures. When the heat flux at a surface is then equated to the heat flux from the fuel (inner surface) or the heat flux to the coolant (outer surface), an expression for the heat flux in terms of the average matrix temperature may be derived. In the effective fuel-matrix heat transfer conductance Equation (2-80), "thick" matrix necessitates an added term

$$\frac{1}{z_0}, \qquad z_0 \equiv \frac{k_c a}{f R_0}$$

where

$$a \equiv \frac{1}{ln \frac{R_1}{R_0}}$$

Thus, $1/z_0$ is the effective matrix thermal resistance for the inner surface of a cylindrical shell. A similar term can be derived for convective heat transfer from the outer surface of the matrix to the coolant and is included in the effective heat transfer coefficient from the clad to the coolant:

$$z_1 \equiv \frac{k_c a}{(1-f) R_1}.$$

2.2.10 Consideration of Heat Capacity of Components

The heat transferred between components by conduction is evaluated from a numerically implicit form of Equation (2-66):

$$\begin{aligned} q_{12}\Delta t &= K_{eff}\left[\left(T_1^o - \frac{q_{12}\Delta t}{C_1}\right) - \left(T_2^o + \frac{q_{12}\Delta t}{C_2}\right)\right]\Delta t \\ &= K_{eff}\frac{C_1 C_2}{C_1 C_2 + (C_1 + C_2) K_{eff}\Delta t}(T_1^o - T_2^o)\Delta t \end{aligned} \qquad (2\text{-}91)$$

Here, C_i is again the total heat capacity of component i.

2.2.11 Effective Heat Capacity of Cladding

The formulation of gap heat transfer in Section 2.2.8 implicitly considers the finite heat capacities of the fuel and the cladding. Equations (2-87) and (2-88) are solved for T_c in the form of

$$T_c = \frac{(\Delta E_{cond}+\Delta E_{conv}+\Delta E_{rad}+\Delta E_{oxid})_c}{C_{CL}+\frac{C_{FU}h_{gap}A_f\Delta t}{C_{FU}+h_{gap}A_f\Delta t}} + \text{other terms} \tag{2-92}$$

that may be interpreted as defining an effective heat capacity

$$C_{CL,eff} \equiv C_{CL} + \frac{C_{FU}h_{gap}A_f\Delta t}{C_{FU}+h_{gap}A_f\Delta t} \tag{2-93}$$

for the cladding. This effective heat capacity implicitly accounts for energy transferred to the fuel pellets through the relatively tight coupling of CL to FU. It is used in estimating the temperature change of cladding in Equation (2-91) and in several other heat transfer models.

2.2.12 Conduction to Boundary Heat Structures

Optionally, conduction from a designated component in the outermost radial ring to the radial boundary heat structures specified on input records COR_ZP may be calculated. The heat flux is given by

$$q_{C-HS} = \frac{T_C - T_{HS}}{R} \tag{2-94}$$

where T_C is the temperature [K] of the core component, T_{HS} is the temperature [K] of the first node of the heat structure (typically an insulator), and R is the total contact resistance, defined as

$$R = R_{gap} + R_{dif} \tag{2-95}$$

where

$$R_{gap} = \Delta r_{gap} / k_{gap} \tag{2-96}$$

$$R_{dif} = \sqrt{\frac{\pi \Delta t}{(k \rho c_p)_{HS}}} \tag{2-97}$$

In the above equations, Δr_{gap} is the thickness of a gap between the core component and the heat structure, k_{gap} is the thermal conductivity of the gap material (calculated from the Material Properties package), Δt is the COR package timestep, and k, ρ, and c_p are the thermal conductivity, density, and specific heat, respectively, of the heat structure material. The thermal diffusive resistance R_{dif} is used to mitigate temperature oscillations that may arise from the numerically explicit coupling between the COR and HS packages. The user may specify on input record COR_BCP which core component is used in this model, what the gap material and thickness are, and what the value of the thermal diffusion constant $(\pi / \mathrm{k}\, \rho\, \mathrm{c_p})^{1/2}$ is for the heat structure (since these properties are not currently accessed from the MP package).

2.3 Convection

Convective heat transfer is treated for a wide range of fluid conditions. Emphasis has been placed on calculating heat transfer to single-phase gases, since this mode is the most important for degraded core accident sequences. A simple set of standard correlations has been used for laminar and turbulent gas flow in both forced and free convection; these correlations give the Nusselt (Nu) number as a function of Reynolds (Re) and Rayleigh (Ra) numbers. Because the numerical method is only partially implicit, the dependence of heat transfer coefficients on surface and fluid temperatures can induce numerical oscillations in calculated temperatures. The calculated heat transfer coefficients for both vapor and liquid heat transfer are therefore relaxed by averaging each with its previously calculated value to mitigate the oscillations.

Since the COR cell nodalization is typically much finer than the CVH nodalization, approximate temperature and mass fraction distributions in the control volumes interfacing with the core and lower plenum must be calculated in the COR package to properly determine the convective heat transfer rates for each COR cell. This temperature distribution is calculated in the COR package in what is termed the dT/dz model, which is described separately in Section 2.5.

In earlier versions of MELCOR, limitations in several models made it difficult—if not impossible—to perform calculations using a fine CVH nodalization with one control volume for each core cell or each small number of core cells. MELCOR 1.8.4 and later versions of the code include improvements in the dT/dz model and incorporate a core flow blockage model (in the FL package). These make such calculations more practical, although some penalty in terms of increased CPU time requirements should still be expected. It is recommended that the new default dT/dz modeling be used (COR_TIN record is *not allowed* in MELCOR 2.x) and that the flow blockage model be invoked and momentum flux terms calculated in the core flow paths (see the FL Package Users' Guide). In the discussion that follows, all fluid temperatures refer to local temperatures, whether calculated by the dT/dz model or taken directly from a fine-scale CVH nodalization.

Heat transfer rates are calculated for each component by the equation

$$q = h_{rlx} A_s (T_s - T_f) \quad (2\text{-}98)$$

where

h_{rlx} = relaxed heat transfer coefficient [W/m^{2}K],

A_s = component surface area [m^2] for heat transfer, accounting for the effects of conglomerate debris (see Section 3.1.6),

T_s = component surface temperature [K], and

T_f = local fluid temperature [K].

MELCOR 1.8.4 and earlier versions used estimated new-time component temperatures in an effort to prevent numerical oscillations in the component heat transfer rates. This approach has been replaced by a semi-implicit calculation of the gap term, described in Section 2.2.8, which has been found to be more effective and reliable.

The unrelaxed heat transfer coefficient, h_{corr}, is calculated from various correlations for the Nusselt number (which is discussed in the following subsections):

$$Nu = h_{corr} D_h / k \quad (2\text{-}99)$$

where

D_h = hydraulic diameter [m] for each component surface, defined by the user on input record COR_EDR and

k = fluid thermal conductivity [W/m-K].

Relaxed heat transfer coefficients for COR subcycle n are given by

$$h^n_{rlx,f} = f_{old,f} h^{n-1}_{rlx,f} + \left(1 - f_{old,f}\right) h_{corr,f} \quad (2\text{-}100)$$

where $f_{old,f}$ is the fraction of the old value to be used for fluid f (vapor or liquid), adjustable through sensitivity coefficient array C1200, with default values of 0.5 and 0.9 for vapor and liquid heat transfer, respectively.

2.3.1 Laminar Forced Convection

For laminar forced flow in intact geometry, the Nusselt number is given by a constant, representing the fully developed Nusselt number for constant heat flux, multiplied by a developing flow factor:

$$Nu = C(n)\ g_{dev} \tag{2-101}$$

where the constant C(n) is currently defined for both rod bundle arrays (n = 1) and circular tubes (n = 2) to be 4.36 and is implemented as sensitivity coefficient array C1212. The developing flow factor is currently that used in MARCH 2 in connection with gaseous diffusion-limited oxidation [17], with the Prandtl number used instead of the Schmidt number:

$$g_{dev} = 1 + \frac{0.00826}{F(z) + 0.0011} \tag{2-102}$$

The constants have been implemented in sensitivity coefficient array C1213, and *F(z)* is a nondimensional entrance length:

$$F(z) = \frac{(z - z_0)}{D_h\ Re\ Pr} \tag{2-103}$$

where (z—z_0) is the distance from the flow entrance, D_h is the hydraulic diameter, Re is the Reynolds number, and Pr is the Prandtl number. In the present version of the code, (z—z_0) is set to 1000 m, effectively eliminating any developing flow effects.

2.3.2 Turbulent Forced Convection

For turbulent flow in channels, the Dittus-Boelter correlation [18] is used:

$$Nu = 0.023\ Re^{0.8}\ Pr^{0.4} \tag{2-104}$$

The coefficients and exponents in Equation (2-104) are implemented in sensitivity coefficient array C1214.

Rather than defining a critical Reynolds number that controls whether laminar or turbulent correlations are used, both correlations are evaluated, and the maximum of the turbulent and laminar Nusselt numbers is used to calculate the forced convection heat transfer coefficient.

2.3.3 Laminar and Turbulent Free Convection

For laminar free convection in narrow channels, the following correlation for an enclosed air space between vertical walls is used [19]:

$$Nu = 0.18\ Ra_f^{1/4}\ (L / D_h)^{-1/9} \tag{2-105}$$

where *L* is the channel length. For turbulent free convection a similar correlation is used, differing only in the default values for the multiplicative constant and the exponent for the Rayleigh number [10]:

$$Nu = 0.065\, Ra_f^{1/3} (L/D_h)^{-1/9} \quad (2\text{-}106)$$

The coefficients and exponents in Equations (2-105) and (2-106) have been implemented as sensitivity coefficient arrays C1221 and C1222, respectively.

As for forced convection, the maximum of the laminar and turbulent Nusselt numbers is used to evaluate the free convection heat transfer coefficient. The maximum of the forced and free convection heat transfer coefficients is then used in Equation (2-98) to calculate the heat transfer rate for a given component. This treatment alleviates some numerical difficulties that may occur if ranges are defined for the various flow regimes, with discontinuities in the Nusselt number at the transition points between regimes.

2.3.4 Convection from Particulate Debris

For particulate debris, correlations for isolated spherical particles are currently used in the COR package for convection to gases. (Surface areas for particulate debris are normally so high that practically any correlation almost completely equilibrates the gas temperature with the debris temperature.) For forced convection, the following correlation is used [20]:

$$Nu = 2.0 + 0.6\ Re_f^{1/2}\ Pr_f^{1/3} \quad (2\text{-}107)$$

For free convection, the Reynolds number is replaced by the square root of the Grashof number [11]:

$$Nu = 2.0 + 0.6\ Gr_f^{1/4}\ Pr_f^{1/3} \quad (2\text{-}108)$$

The coefficients and exponents in Equations (2-107) and (2-108) have been implemented as sensitivity coefficient arrays C1231 and C1232, respectively. In both equations, the properties are evaluated at the film temperature (i.e., the average of the debris and dT/dz model fluid temperatures). The maximum of the free and forced convection Nusselt numbers is once again used to calculate the heat transfer coefficient.

2.3.5 Convection from Pebble Bed

The convection correlations for the particulate debris are also used for the PBR core. Presently, the porosity and hydraulic diameter for these correlations are also the same as for the debris (see COR_ZP in the COR users' guide)

2.3.6 Boiling

By default, for liquid-covered components, the COR package uses the correlations from the HS package to treat boiling (see the HS Reference Manual). However, if the default value of sensitivity coefficient C1241(5) is changed, the simplified boiling curves from the MARCH 2.1 code [3] can be used to calculate the heat transfer coefficient:

$$h = 34.5P^{1/4}\Delta T^{1.523} \quad \Delta T < 23.4K \tag{2-109}$$

$$h = 1.41 \times 10^{7} P^{1/4}\Delta T^{-2.575} \quad \Delta T \geq 23.4K \tag{2-110}$$

where

P = pressure [Pa] and

ΔT = surface superheat, (T_s - T_{sat}),

and the constants have been implemented as sensitivity coefficient arrays C1241 and C1242.

For the film-boiling regime ($\Delta T \geq 23.4$ K), a radiation component is added to the convective heat transfer coefficient:

$$h_{rad} = \sigma \ \varepsilon \ \frac{T_s^4 - T_l^4}{T_s - T_l} \tag{2-111}$$

where ε is a hardwired constant emissivity of 0.4.

2.3.7 Heat Transfer from Horizontal Surfaces of Plates

For most core components—fuel rods, BWR canisters, control elements, and BWR Control Rod Guide Tubes—convective heat transfer takes place from a lateral (vertical) surface. If there is a water pool in the associated core cell, the component surface is progressively and smoothly covered or uncovered as the pool surface rises or falls.

Plates, however, have horizontal bottom and top surfaces that can be covered or uncovered with a relatively small change in the pool level. Moreover, different CVH control volumes are ordinarily used to model the regions above and below the core plate, which can be associated with (at most) one of these volumes. Thus, (at least) its other horizontal surface sees fluid in a different control volume than that from which other boundary conditions for the core cell are derived.

When the SS component is used to represent a plate, an optional model exists to calculate heat transfer from its horizontal surfaces to water pools above and/or below.

The model may be controlled independently for the two surfaces and is off by default. If the model is on, the heat transfer coefficient for the top surface is ordinarily evaluated from the built-in pool boiling correlation (Section 2.3.6) and that for the bottom surface from the built-in correlation for downward-facing boiling (Section 6.1). Either or both may be overridden by constant values or by values calculated as control functions. In any case, the temperature difference is based on the average temperature of the plate in the core cell and that of the pool.

For either plate surface, the total area is considered to be the total cross-sectional area of the core cell. However, the surface of a water pool is not an idealized plane. One would therefore expect some contact with the bottom of the plate while the average pool surface is some finite distance below it and less-than-complete coverage of the top until the average surface is some finite distance above it. In order to account for this, the fraction of the lower horizontal surface involved in heat transfer to a pool is linearly ramped on as the surface of the pool in the core cell below rises to the bottom surface of the plate. Similarly, the fraction covered above is ramped off as the surface of the pool in the core cell above falls to the top surface of the plate. User input is required for both the clearance below the plate that is needed for no contact and the pool depth over the plate that is needed for complete coverage.

This model is activated by specifying necessary inputs on COR_PC record, as described in the COR Package Users' Guide.

2.3.8 Debris Quenching and Dryout

Heat transfer from debris to liquid water pools may occur in two distinct modes. In the falling-debris quench mode, failure of the core support plate triggers the relocation of a large mass of hot debris from the core region to the lower plenum. In this mode, it is assumed that transient heat transfer rates may be sufficient to rapidly quench the hot debris and/or generate large steam pressure excursions. Following the quench mode, it is assumed that continued decay heat generation in the stationary debris bed in the lower plenum either boils off any remaining water in the lower plenum or quickly lead to debris-bed dryout with an overlying water pool. The heat transfer from the debris bed to the overlying pool of water following debris-bed dryout is relatively modest and is calculated with an appropriate dryout heat flux correlation, whose description follows.

The falling-debris quench model is active by default. If *de*activated through user input, the debris is assumed to relocate instantaneously from the core region to an unquenched debris bed in the lower plenum. The model may be *de*activated if a value of 0.0 for the quench heat transfer coefficient is specified on input record COR_LP. No other parameters on this record are then necessary. The heat transfer calculated by the model may or may not be sufficient to fully quench the debris before it reaches the bottom of the lower plenum, depending on the values chosen for the model parameters that are described in this section.

Beginning from the time of core support plate failure in each radial ring, the elevation of the leading edge of the falling debris is determined, assuming a constant user-specified descent velocity (with a default of 5 m/s). The axial elevation of the leading edge of the falling debris is given by

$$z_d = z_{csp} - v_d\ (t - t_{fail}) \tag{2-112}$$

where z_{csp} is the initial elevation of the core support plate, v_d is the velocity of the falling debris, t is the current time, and t_{fail} is the failure time of the support plate in the ring. Debris from core cells above elevation z_d is relocated downwards, subject to the availability of free volume and the absence of additional supporting structures.

When the leading edge of the falling debris enters the pool of water in the lower plenum, quench heat transfer begins. The heat transfer surface area is the value calculated based on the assumption that the debris particles have an equivalent spherical diameter equal to the user-specified hydraulic diameter for particulate debris (which is input on record COR_EDR). The user-specified quench heat transfer coefficient (which is input on record COR_LP) is assumed to remain constant until the leading edge of the falling debris reaches the bottom of the lower plenum (i.e., the elevation of the lower head). After that time, a decay factor initially equal to unity is applied to the user-specified heat transfer coefficient.

The decay factor is intended to simulate the reduction in heat transfer that occurs during the transition from the quench period to the debris-bed configuration. During this period of transition, additional hot debris from the core region may relocate to the lower plenum as a result of radial spreading between the rings in the core region. Therefore, the decay factor has a time constant equal to the time constant for radial spreading of solid debris (see Section 3.2.5). The decay factor also includes a term to arrest the decay as long as significant amounts of debris continue to migrate into the failed ring from other core regions. Soon after the bulk of the debris has relocated, the decay factor quickly decreases. When the value of the decay factor falls below 0.01, it is assumed that the transition to a stable debris bed geometry is complete, and all subsequent debris-to-pool heat transfer in that radial ring is limited by the dryout heat flux correlation discussed below. The time-dependent heat transfer decay factor, f(t), is given by

$$f(t+\Delta t) = min\ [\ 1, f(t)\ exp\ (-\Delta t\ /\ \tau_{spr}\ ) + V_{cor}\ /\ V_{LP}\ ] \tag{2-113}$$

where τ_{spr} is the time constant for radial spreading of solid debris described in Section 3.2.5, V_{cor} is the volume of debris that relocates into the ring from radial spreading in the core region during the core timestep Δt, and V_{LP} is the volume of debris in the ring beneath the level of the core support plate.

During the short period between the failure of the core support plate and the time at which the leading edge of the falling debris reaches the lower head, the models for candling, dissolution, and radial spreading of debris in the affected ring are deactivated. This action is taken because those models implicitly assume a stationary debris configuration. In addition to the quench heat transfer coefficient, the user may specify a reactor vessel failure pressure (with a default value of 2.0e7 Pa). When the differential pressure between the lower plenum CVH volume and the reactor cavity CVH volume reaches the failure pressure, it is assumed that the lower head in all the core rings contained in the lower plenum CVH volume fails totally. When this happens, all of the debris in the core cells above the failed lower head is ejected immediately, and further quench heat transfer in those rings is suppressed. Currently, users are advised to specify a failure pressure below the critical pressure of water (22.0 MPa) because the CVH package may encounter problems above that pressure.

Because of the relatively low default value for the failure pressure (compared to actual failure pressures that may be much higher) the quench model may have a rather limited range of usefulness for some PWR calculations. If the PWR relief valves cycle around 16–17 MPa, then there is very little margin (3–5 MPa) for steam generation between the relief pressure and the critical pressure; hence, even modest fuel-coolant interactions following support plate failure tend to cause “vessel failure.”

For stationary particulate debris beds in liquid water pools, the heat transfer rate is limited by hydrodynamic phenomena that limit the amount of liquid that can reach the debris particles. The conceptual view taken in the COR package is that liquid water moves downward from above to cool the debris and that vapor is produced and moves upward to restrict the flow of liquid. At some total bed-heat flux, this vapor prevents further liquid from reaching the debris. This is the point of incipient dryout.

The COR package uses the Lipinski zero-dimensional correlation [21] to calculate the dryout heat flux, q_d, which is then applied as a limiting maximum heat transfer rate from a particulate debris bed (using the cell cross-sectional area rather than the total particulate surface area), which may occupy one or more axial levels:

$$q_d = 0.756\ h_{lv} \left[\frac{\rho_v\ (\rho_l - \rho_v)\ g\ d\ \varepsilon^3\ (1 + \lambda_c / L)}{(1-\varepsilon)\left[1 + (\rho_v / \rho_l)^{1/4}\right]^4} \right]^{1/2} \qquad (2\text{-}114)$$

In this equation, h_{lv}, ρ_l, and ρ_v are the latent heat, liquid, and vapor densities of water, respectively; g is the gravitational acceleration; d is the debris particle diameter; ε is the bed porosity; L is the total bed depth; and λ_c is the liquid capillary head in the debris bed,

$$\lambda_c = \frac{6\ \sigma\ \cos\theta\ (1-\varepsilon)}{\varepsilon\ d\ (\rho_l - \rho_v)\ g} \qquad (2\text{-}115)$$

where σ is the water surface tension and θ is the wetting angle. The leading constant, the nominal capillary head for 0.5 mm particles in approximately 0.089 m of water, and the minimum bed porosity allowed in the correlation are accessible to the user as sensitivity coefficient array C1244. A default minimum porosity of 0.15 was selected to ensure that some heat transfer occurs from molten debris pools. The actual capillary head is adjusted for particle diameter size within the model.

If one or more axial levels give heat transfer rates totaling the dryout maximum, no heat transfer is calculated for particulate debris or other intact structures below this axial level. Furthermore, in cells in which debris is undergoing quenching at the rate given by the dryout heat flux, no convective heat transfer to the pool is calculated for other components in that cell.

2.4 Molten Pool Heat Transfer

2.4.1 Contiguous Physical Molten Pools

As previously mentioned, contiguous volumes containing molten pool components constitute coherent molten pools that are assumed to be uniformly mixed by convection so as to have uniform material composition, radionuclide composition, and temperature. Two distinct molten pools (oxide and metallic) are allowed in the lower plenum, and potentially four molten pools can be modeled in the upper core (oxide and metallic in the channel and oxide and metallic in the bypass volume). A search is made in the core and the lower plenum to find the largest contiguous molten pools (by volume), which are then modeled as convecting molten pools. This requirement for contiguity ensures that isolated cells containing molten materials are not mixed with the convecting pools. These convecting molten pools transfer heat to the lower head (or the lower plenum pools); the fluids (water or steam); the substrate material; and the structural components, such as the shroud (PWR). In addition, the transfer of heat and radionuclides occur between stratified molten pools. New models have been added to predict the heat transfer coefficients to the substrate supporting the molten pool, the heat transfer between pools, and the heat transfer to surroundings. Note that isolated volumes of molten pool material are not part of these contiguous molten pools and are not included in the convective mix. They have distinct temperatures and composition, and transfer heat, as discussed in previous sections.

2.4.2 Convection Heat Transfer

In MELCOR 1.8.6 and 2.1, the portion of the vessel that is below the elevation of the BWR baffle plate or the PWR bottom plate (HLST) is treated by the lower head model. In addition, this elevation is used to distinguish convecting molten pools in the lower plenum

from those in the upper core (see Figure 2.3). The convective molten pools contain molten pool material in contiguous COR cells and are therefore free to mix. MELCOR 1.8.6 mixes these molten pools so that they are uniform in temperature and material and radionuclide composition. These molten pools may then transfer heat to their surroundings by convective heat transfer to the supporting substrate; by radiation from the upper surface; by convection to pool or atmosphere at the upper surface, and, in the case of stratified molten pools, by heat transfer between pools. Heat balances for two stratified molten pools are summarized as in Equation (2-116) and Equation (2-118).

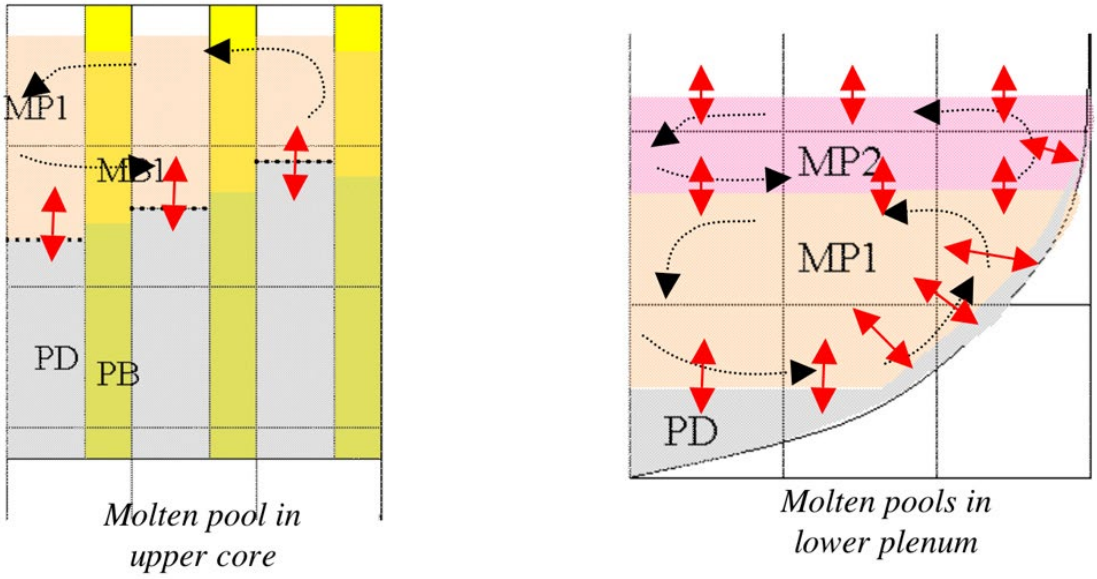


Figure 2.3 Convecting Molten Pools

$$MC_{P,MP1}\frac{T^n_{MP1}-T^o_{MP1}}{\Delta t}=\dot{Q}_{MP1,decay}$$
$$-\sum_{s\in seg} h_{MP1\to s}A_s(T^n_{MP1}-T_s)-h_{MP1\to MP2}A_{1,2}(T^n_{MP1}-T^n_{MP2}) \qquad (2\text{-}116)$$
$$-\left\langle h_{MP1-fluid}A_f\left(T_{MP1}-T_{fluid}\right)-\sigma\varepsilon_{eff}A_{up}(T^4_{MP1}-T^4_{ambient})\right\rangle\cdot\delta(MP2)$$

where

$$\delta(MP2)=0\ when\ molten\ pool\ MP2\ exists,1\ otherwise \qquad (2\text{-}117)$$

$$MC_{P,MP2}\frac{dT_{MP2}^{n}}{\Delta t} = \dot{Q}_{MP2,decay} - \sum_{s\in seg} h_{MPs\rightarrow s}A_s(T_{MP2}^{n} - T_s) + h_{MP1\rightarrow MP2}A_{1,2}(T_{MP1}^{n} - T_{MP2}^{n}) - h_{MP2-fluid}A_f\left(T_{MP2}^{n} - T_{fluid}\right) - \sigma\varepsilon_{eff}A_{up}(T_{MP2}^{4} - T_{ambient}^{4}) \quad (2\text{-}118)$$

Heat transfer coefficients are based on empirical correlations obtained from experiments. These correlations are typically reported in terms of an average Nusselt number calculated from the internal Rayleigh number, as in

$$Nu = C \cdot Ra_i^n \quad (2\text{-}119)$$

$$Ra_i = \frac{\rho^2 g\beta Q H^5}{\lambda\mu^2}Pr = \frac{g\beta Q H^5}{\lambda\nu^2}Pr \quad (2\text{-}120)$$

where

λ = thermal conductivity, kW/m-K,

ρ = density, kg/m^3,

μ = dynamic viscosity, Pa-s,

ν = kinematic viscosity, m^2/s,

β = thermal expansion coefficient, 1/K,

g = acceleration due to gravity, m/s^2,

Q = volumetric heat generation, kW/m^3,

H = height of pool, m, and

Pr = Prandtl number.

The thermal expansion coefficient β_i of material i is taken from MELCOR's material database. The reference temperature $T_{\beta,ref}$ for volume changes is 1673 K. The default values of β_i and molar volume for B4C, B4C-INT, and Silver-Indium-Cadmium are taken to be those of stainless steel. A constant value of β may be specified for molten pools on the MP_PVE input record. If no constant value is specified, the molten pool's mixture thermal expansion coefficient β is calculated as follows:

$$\beta = \frac{\sum_i f_{mol,i}\, v_{mol,i}\, \beta_i}{\sum_i f_{mol,i}\, v_{mol,i}\, (1 + \beta_i(T_i - T_{\beta,ref}))} \quad (2\text{-}121)$$

where $f_{mol,i}$ is the molar fraction of material i in the molten pool, $v_{mol,i}$ is the specific molar volume of material i in the molten pool, and T_i is the temperature of the molten pool.

The dynamic viscosity μ_i of material i of a molten pool is provided using the empirical coefficients $ALPHI$ and $OXMUI$ in an Arrhenius form:

$$\mu_i = OXMUI \cdot \exp\left(\frac{ALPHI}{T}\right) \quad (2\text{-}122)$$

The mixture viscosity μ of an oxidic molten pool is then provided by a third order geometric mean, according to the work of Kendall and Monroe [22], which is weighted by the molar fraction x_i for a given molten pool:

$$\mu = \left(\frac{\sum_i x_i \sqrt[3]{\mu_i}}{\sum_i x_i}\right)^{\frac{1}{3}} \quad (2\text{-}123)$$

The individual material properties and supporting correlations are either inherited from or made consistent with those in the CAV Package and are fully accessible to the MP and COR packages

The viscosity of a metallic molten pool, regardless of composition, is assumed to be that of stainless steel. The CAV Package material database does not contain data for all steels; therefore, for both β_i and μ_i, the COR materials stainless steel, carbon steel, and stainless steel 304 use the properties of iron while steel oxide uses the properties of ferrous oxide (FeO).

The definition of a metallic molten pool is one where most of the conglomerate material's mass in that pool is metal, and conversely for an oxidic pool. The viscous behavior of all default COR materials can be identified by their assignment as being either a metal or an oxide, as is summarized in Table 2-3. An exempted material does not participate in the molten pool viscosity calculation.

Table 2-3 Default Material Types

Material	Material Type
ZIRCALOY	METAL
ZIRCONIUM-DIOXIDE	OXIDE
URANIUM-DIOXIDE	OXIDE
STAINLESS-STEEL	METAL
STAINLESS-STEEL-OXIDE	OXIDE
BORON-CARBIDE	METAL
SIC	METAL
URANIUM-METAL	METAL
ALUMINUM	METAL
ALUMINUM-OXIDE	OXIDE
CADMIUM	METAL
STAINLESS-STEEL-304	METAL
LITHIUM-ALUMINUM	METAL
URANIUM-ALUMINUM	METAL
CARBON-STEEL	METAL
B4C-INT	METAL
ZRO2-INT	OXIDE
UO2-INT	OXIDE
All other materials	EXEMPT

The average Nusselt number can also be calculated from the external Rayleigh number, as in

$$Nu = C \cdot Ra_i^n \tag{2-124}$$

$$Ra = \frac{g\beta\Delta T H^3}{\nu^2} Pr \tag{2-125}$$

Correlations based on the internal Rayleigh number (and the volumetric heat generation rate) assume a steady state condition for the convecting pool. In particular, the heat removed at the boundaries of a molten pool is exactly balanced by the heat generated by decay. The Rayleigh correlation is given in Equation (2-126).

MELCOR, however, must be able to calculate transient convective conditions as molten pools are formed or grow from relocation events. For example, if molten metallic material containing little or no radionuclide mass relocated into the lower plenum, the internal heat generation could be small even though convective heat loss to the boundaries could be quite large. In other words, the initial situation would be far from steady state. Convective currents in the molten pool are driven by density gradients that result both from internal heat generation and from temperature differences across the boundary layers. For transient conditions, the steady state correlations have been adapted to obtain a correlation for the internal Rayleigh number given by Equation (2-126) based on the average of the decay heat and the boundary heat losses.

$$\begin{aligned} Ra_{int,MP1} &= \left(\frac{g\beta H^5}{\lambda\nu^2}Pr\right)_1 \times \dot{Q}_1 \\ &= \left(\frac{g\beta H^5}{\lambda\nu^2}Pr\right)_1 \\ &\times \frac{1}{2}\left[\begin{array}{l} \dot{Q}_{MP1,decay} + \sum_{s\in seg} h_{MP1\to s}A_s|T_{MP1} - T_s| + h_{MP1\to MP2}A_{1,2}|T_{MP1} - T_{MP2}| \\ +\left\langle h_{MP1-fluid}A_f\left|T_{MP1} - T_{fluid}\right| + \sigma\varepsilon A_{up}(T_{MP1}^4 - T_{ambient}^4)\right\rangle \cdot \delta(MP2) \end{array}\right] \end{aligned} \tag{2-126}$$

This equation approaches the steady state Equation (2-120) as the decay heat approaches the boundary heat loss at steady conditions. The following two analogous equations have been developed for the upper molten pool:

Steady State Upper Pool:

$$\begin{aligned} Ra_{\text{int},MP2} &= \left(\frac{g\beta H^5}{\lambda\nu^2}\text{Pr}\right)_2 \cdot \dot{Q}_{MP2,decay} \\ &= \left(\frac{g\beta H^5}{\lambda\nu^2}\text{Pr}\right)_2 \cdot \left[\begin{array}{l} \sum_{s\in seg} h_{MPs\to s}A_s\left(T_{MP2} - T_s\right) + h_{MP1\to MP2}A_{1,2}\left(T_{MP2} - T_{MP1}\right) \\ + h_{MP2-fluid}A_f\left(T_{MP2} - T_{fluid}\right) + \sigma\varepsilon A_u(T_{MP2}^4 - T_{MP2}^4) \end{array}\right] \end{aligned} \tag{2-127}$$

Transient Upper Pool:

$$\begin{aligned} Ra_{\text{int},MP2} &= \left(\frac{g\beta H^5}{\lambda\nu^2}\text{Pr}\right)\cdot\dot{Q}_2 \\ &= \left(\frac{g\beta H^5}{\lambda\nu^2}\text{Pr}\right)\cdot\frac{1}{2}\left[\begin{array}{l} \dot{Q}_{MP2,decay} + \sum_{s\in seg} h_{MPs\to s}A_s\left|T_{MP2} - T_s\right| + h_{MP1\to MP2}A_{1,2}\left|T_{MP1} - T_{MP2}\right| \\ + h_{MP2-fluid}A_f\left|T_{MP2} - T_{fluid}\right| + \sigma\varepsilon A_{up}(T_{MP2}^4 - T_{ambient}^4) \end{array}\right] \end{aligned} \tag{2-128}$$

Note that for the case in which only one molten pool exists, the heat balance for the single pool would include radiation and convective heat losses to pool/atmosphere but would not have an interfacial heat transfer term between molten pools. Calculation of the average bulk heat transfer for a molten pool is further complicated by the possibility of stratified pools with heat transfer at the interface between the two pools. Heat transfer between pools might assist the natural convection currents and thereby enhance heat transfer. This heat transfer term couples the heat balances for the two contacting molten pools, resulting in an iterative scheme for numeric solution. This scheme, outlined in Figure 2.4, consists of two inner iterations for the individual pools within an outer iteration for overall convergence on Rayleigh numbers for both pools.

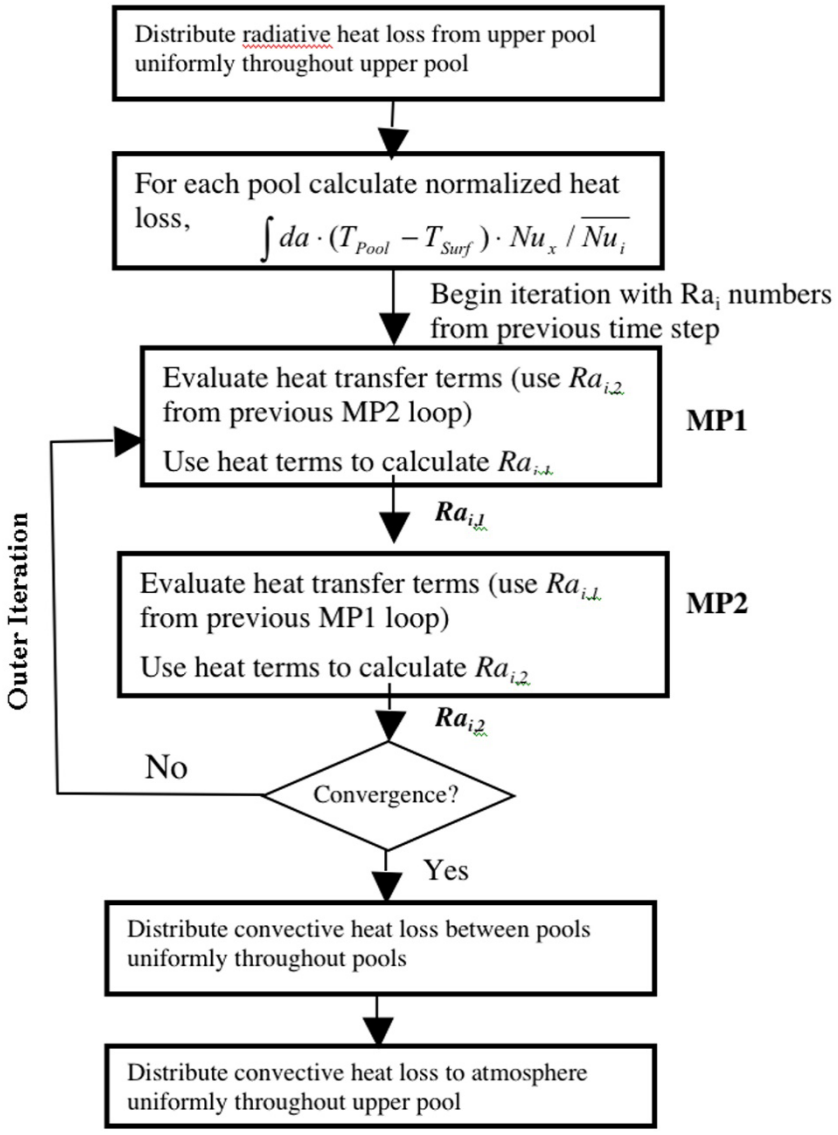


Figure 2.4 Outline of iterative solution for convective heat transfer coefficients

Even though this convergence iteration is based on the average Rayleigh number, which is used to derive average Nusselt numbers, local heat transfer coefficients are used in the pool energy balance. The total heat transfer to the underlying substrate (or lower head), normalized by the average Nusselt number, is calculated before entering the iterative loop.

The average Nusselt number, determined from the Rayleigh number correlation, is then used to remove the normalization and to calculate the total heat loss at the boundary. An empirical correlation must be used to evaluate the normalized local heat transfer profile. An example of such an empirical correlation for relating the local heat transfer to maximum heat transfer coefficient as a function of latitude along the lower head is reported by Bonnet [23] and reproduced in the following equations:

$$\frac{\varphi}{\varphi_{\max}} = \sin(\theta)^{1/3}\left(\frac{1-\cos(\theta)}{k\frac{H}{R}}\right)^{4/3} \quad \textit{for } \theta \le \arccos(1-k\tfrac{H}{R}) \tag{2-129}$$

$$\frac{\varphi}{\varphi_{\max}} = \sin(\theta)^{1/3} \quad \textit{for } \arccos(1-k\tfrac{H}{R}) < \theta \le \arccos(1-\tfrac{H}{R}) \tag{2-130}$$

A family of four curve sets has been generated as a function of the molten pool height, which is correlated by the parameter k (each curve set corresponds to a distinct value of k). The parameter k expresses the height of the stable molten pool. The parameter k is evaluated for each pool, and the distribution is obtained from the interpolation of curve sets.

These correlations are expressed in terms of the peak heat transfer coefficient, making it necessary to recast the equations in terms of the average heat transfer coefficient. To accomplish this, the functions were piecewise curve fit with 2nd order (upper layer) and 3rd order (lower layer) polynomials that could be readily integrated to obtain an average heat flux. These polynomials were then reformulated in terms of the ratio of the local Nusselt number to the average value, as given by Equations (2-131) and (2-132).

Where K = 1-unstable-layer height/total pool height (from tables),

$$\frac{\varphi}{\varphi_{\max}} = c(i)\cdot\theta + b(i)\cdot\theta^2 + a(i)\cdot\theta^3 \quad \textit{for } \theta \le \arccos(1-k(i)\tfrac{H}{R}) \tag{2-131}$$

$$\frac{\varphi}{\varphi_{\max}} = f(i)\cdot\theta + e(i)\cdot\theta^2 + d(i)\cdot\theta^3 \quad \textit{for } \arccos(1-k(i)\tfrac{H}{R}) < \theta \le \arccos(1-\tfrac{H}{R}) \tag{2-132}$$

The correlations, together with the fitted curves employed by MELCOR, are plotted in Figure 2.5. These distributions have been generalized to allow user modification of the polynomial curves through sensitivity coefficient C1290. The user can specify up to four curve sets (each set containing polynomials below and above the inflection point) for four distinct values of *k*. The user specifies the value of *k* for each family of curves together with coefficients a(i), b(i), c(i), d(i), e(i), and f(i) for the associated polynomial fits.

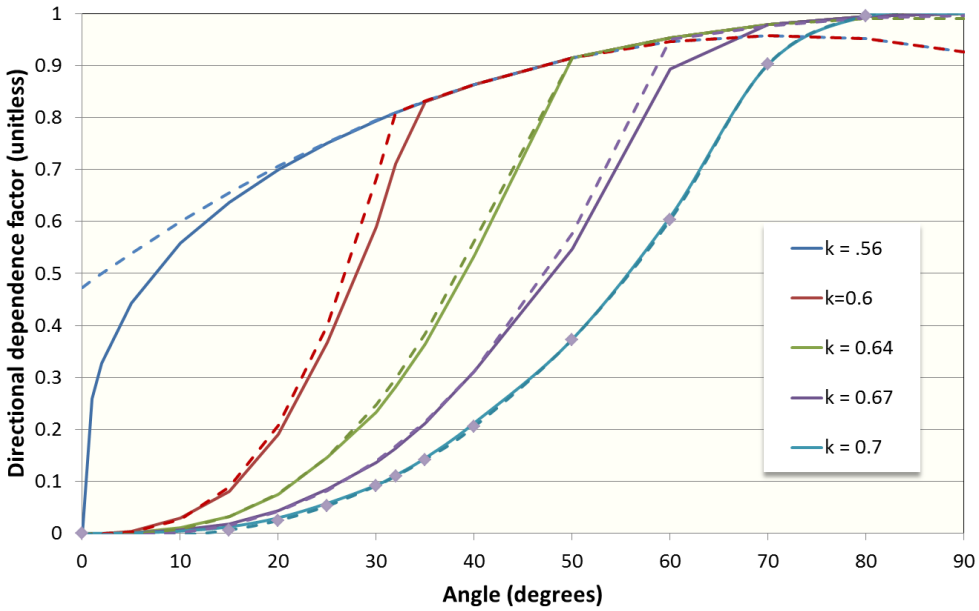


Figure 2.5 Ratio of Local heat transfer to peak heat transfer coefficient

Radiative heat losses as well as convective heat losses to fluid (steam or water) are calculated for the upper surface of the molten pool only. In the case of stratified pools, these radiative losses are calculated only for the upper pool, and heat losses from the upper surface of the lower pool are assumed to be only due to the interface heat transfer. Since the pools are uniformly mixed, the total calculated radiative heat loss is uniformly removed from throughout the pool volume. In addition, the heat transferred between two contacting molten pools is similarly removed uniformly throughout the pool volume.

At the interface, the heat transfer from a molten pool is calculated using a heat transfer coefficient based on the internal Rayleigh number correlation and the temperature difference between the bulk pool and the interface. Note that the formation of an interface crust with conductance temperature drop has not been modeled. The interface temperature is defined implicitly by the assumption that the convective heat loss from one pool is exactly equivalent to the convective heat gain by the other pool. An effective heat transfer correlation relating the bulk temperatures of the two pools can be obtained by eliminating the interface temperature, as in this equation:

$$\overline{htc_{1\to 2}} = \frac{htc_1 \cdot htc_2}{htc_1 + htc_2} \tag{2-133}$$

The corresponding interface temperature satisfies Equation (2-134) and, in most cases, the value need not be used:

$$\overline{htc_{1\to 2}} \cdot (T_{MP1} - T_{MP2}) = htc_1 \cdot \left(T_{MP1} - T_{interface}\right) \tag{2-134}$$

However, if the calculated interface temperature is less than the melting temperature of the residual liquid in the oxide molten pool, the assumption of pure convective heat transfer between the pools cannot be correct, as we would predict the existence of a solid crust at the interface. In this case, the heat transfer coefficient is reduced by using the melting temperature for the interface temperature in the right-hand side of Equation (2-134). This correction accounts for the formation of the interfacial crust, even though the details of that crust are not modeled. In effect, the net heat transfer is limited by convection from the interior of the oxidic pool to the surface of the interfacial crust.

In the case of small molten pools, an implicit calculation of the heat transfer is warranted. In calculating heat transfer between stratified molten pools, the total pool heat capacity, H_{MPi}, is used to arrive at a reduced interfacial heat transfer coefficient, as calculated in the following equation:

$$HA_{effective} = \overline{htc_{1\to 2}} \cdot Area_{MP1MP2} \cdot \frac{C1 \cdot C2}{[(C1+1)\cdot(C2+1)-1]} \tag{2-135}$$

where

$$C1 = \frac{H_{MP1}}{dt \cdot Area_{MP1MP2} \cdot \overline{htc_{1\to 2}}} \tag{2-136}$$

$$C2 = \frac{H_{MP2}}{dt \cdot Area_{MP1MP2} \cdot \overline{htc_{1\to 2}}} \tag{2-137}$$

When both heat transfer coefficients are large, the reduced heat transfer coefficient approaches the effective heat transfer coefficient, $\overline{htc_{1\to 2}}$, derived above. If the heat capacity of one molten pool becomes negligibly small, the heat transfer rate is limited by the heat capacity of that pool.

The calculation of the internal Rayleigh number requires certain material properties (kinematic viscosity and thermal expansion coefficient) for the molten pools that were not previously present in the MP database. However, these properties were already evaluated by the CAV package for ex-vessel molten material in the cavity but were never added to the MP package. Those correlations used by the CAV package were added to

the MP package and are now accessible to the COR package in evaluating the internal Rayleigh number.

A Nusselt number correlation, $Nu = A(j) \cdot Ra^{n(j)} \Pr^{m(j)}$, is assumed for each molten pool surface: oxide pool to radial boundary (j = 1), oxide pool to interface (j = 2), oxide pool to atmosphere (j = 3), metallic pool to lower surface (j = 4), metallic pool to radial surface (j = 5), and metallic pool to upper surface (j = 6). The coefficient A(j) and the exponent n(j) are accessible to the user as sensitivity coefficient C1280(j,1) and C1280(j,2), respectively. The default coefficients for the heat transfer correlations assumed at each boundary are summarized in Table 2-4.

Table 2-4 Assumed convective boundary condition at molten pool surfaces

j	Description	Rayleigh Number	A(j)	N(j)	M(j)	Reference
1	Oxide pool to radial boundary	Internal	.3	.22	0	ACOPO [24]
2	Oxide pool to interface	Internal	.381	.234	0	Bonnet [23]
3	Oxide pool to atmosphere	Internal	.381	.234	0	Bonnet [23]
4	Metallic pool to lower surface	External	.069	.333	0.074	Globe & Dropkin [25]
5	Metallic pool to radial surface	External	.3	.22	0	ACOPO [24]
6	Metallic pool to upper surface	External	.3	.22	0	ACOPO [24]

Because of inertia, some period of time is necessary to establish steady convective currents. The model includes a time constant that can be used to capture this effect, and the internal Rayleigh number calculated for each pool is modified according to

$$Ra_i^{new} = Ra_i^{old} + (Ra_i^{calculated} - Ra_i^{old}) \cdot (1 - e^{\frac{-dtc}{timeconstant}}) \quad (2\text{-}138)$$

The user can specify a distinct time constant for each pool in sensitivity coefficient C1281. The default time constant is zero, for which the new Rayleigh number is equal to the calculated value.

2.4.3 Heat Transfer to Underlying Substrate from Molten Pool (Integral Solution to Stefan Problem)

A molten pool that forms in either the lower plenum or in the core region is supported by a solid substrate material. This substrate may be the lower head or particulate debris, possibly with a dense, impermeable crust at the interface. In any event, an interface between solid and liquid phases exists that may develop and move as material is transported between phases. Thermal properties may vary greatly between the two phases and temperature gradients can be highly nonlinear over the dimension of a COR cell. For this class of problem, the position of the interface is generally not fixed but must be calculated as part of the solution. These 'moving boundary problems' are also referred to as Stefan problems as a result of his work in studying the melting of the polar ice cap in 1890 [26].

An integral model has been developed for calculating transient heat conduction from the molten pool to the underlying substrate. The integral method was selected because it was more conducive to adaptation into the existing MELCOR structure than would be finite difference solutions that would require many nodes to capture temperature gradients and resolve the position of the interface. The integral model reproduces the overall system heat balance by integrating the heat conduction equations over the spatial domain while assuming the shape of the temperature profile in the substrate and applying appropriate boundary conditions. The temperature profile has been generalized from conventional quadratic form. The integral model is capable of handling melting, freezing, and transient heat-up of the substrate. The location of the interface is tracked as it progresses through a cell and into adjacent cells. This Stefan model is a superposition on existing MELCOR heat transfer mechanisms.

The variables carried by the code are the molten pool temperature, T_P, a substrate temperature ahead of the thermal penetration front, T_0, a solid surface temperature, T_s, a conductive heat flux into the solid crust/substrate, q_s, and a surface position, x_s. The boundary condition is convection from the molten pool to the solid surface.

$$q_s = h(T_P - T_s) \tag{2-139}$$

The task, given old values for T_s, q_s, and x_s, is to find new values. There are two important cases to consider: heat transfer with stationary interface or heat transfer with a moving interface such as from ablation or freezing. The equations are first solved assuming stationary interface. If the resulting surface temperature is less than the melting temperature of the solid and greater than the freezing temperature of the liquid, the results are accepted. Otherwise, equations for a moving interface are solved and melting or freezing is determined (Figure 2.6).

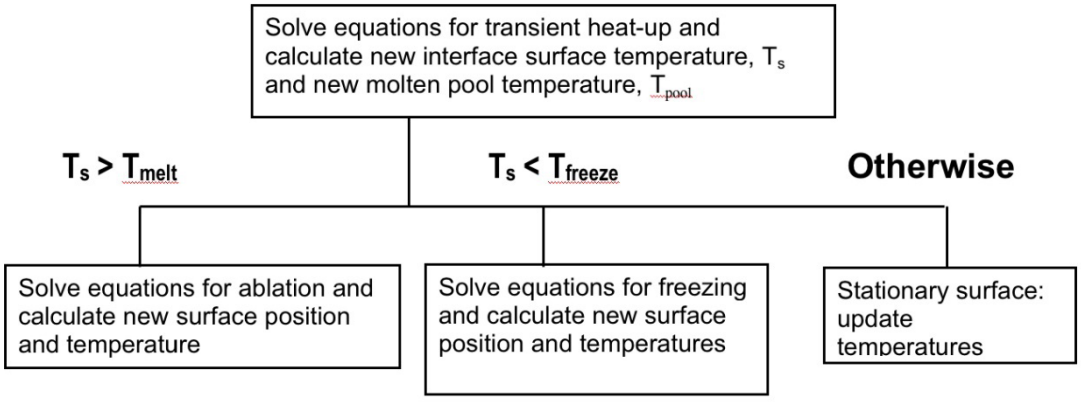


Figure 2.6 Logic for determining moving, or stationary boundary

2.4.3.1 Stationary Surface (Heat-up of Substrate)

For a fixed stationary surface, the one-dimensional finite difference equation for constant heat transfer coefficient is cast in the form of

$$q_s^n = q_{in} = h(T_P - T_s^n) \tag{2-140}$$

The conductive heat flux, q_s is the heat flux at the interface that is conducted into the solid. For the case of the stationary surface the conductive heat flux is equal to the incident convective heat flux from the molten pool. In general, the conductive heat flux need not equal the incident heat flux, the difference accounting for the heat removed or added at the boundary resulting from melting or freezing. The superscripts o and n refer to old and new (end of timestep) values, respectively. The thermal properties of the substrate are summarized in the values of k and κ, which are the thermal conductivity and the thermal diffusivity, respectively. The parameter γ is a parameter that results from the assumed temperature profile in the substrate and is defined by

$$\gamma = \left.\frac{\partial T}{\partial x}\right|_{x=x_s} \int_{x_s}^{\infty} T(x)\ dx / T_s^2 \tag{2-141}$$

For an exponential profile, γ = 1, while for a quadratic one, γ = 2/3 = 0.6667.

A dimensionless group, Z, is introduced into the equation to simplify the solution. When this group is substituted in Equation (2-142) along with the boundary condition, a quadratic equation for Z^n is obtained in Equation (2-143). In the special case in which $q_s^o = 0$, and Z^o is formally indeterminate, the desired result is the one for B = 0 (i.e., $Z^n = \sqrt{D}$). The calculated Z, in Equation (2-143), can then be used to generate the

more physically meaningful variables, such as the conductive heat flux, q_s and the new solid surface temperature, T_s shown in Equation (2-144).

If the new surface temperature that results is less than the melting temperature of the solid and greater than the freezing temperature of the liquid, the results are accepted. If not, the solution for the moving interface is used.

$$Z \equiv \frac{h(T_s - T_0)}{q_s} \tag{2-142}$$

$$(Z^n)^2 - \frac{T_s^o - T_0}{T_P - T_0} Z^o Z^n - \left[\frac{T_s^o - T_0}{T_P - T_0} Z^o + \left(\frac{h}{k}\right)^2 \frac{\kappa \Delta t}{\gamma}\right] = 0 \tag{2-143}$$

$$Z^n = B + \sqrt{B(2+B) + D} \tag{2-144}$$

where

$$B \equiv Z^o \frac{T_s^o - T_0}{2(T_P - T_0)} \tag{2-145}$$

$$D \equiv \left(\frac{h}{k}\right)^2 \frac{\kappa \Delta t}{\gamma} \tag{2-146}$$

$$q_s^n = \frac{h(T_P - T_0)}{Z^n + 1}, \quad T_s^n = T_P - \frac{q_s^n}{h} \tag{2-147}$$

2.4.3.2 Moving Surface (Ablation or Freezing)

For a moving surface, the one-dimensional finite difference equation for constant heat transfer coefficient is cast in the form of

$$\frac{\gamma}{\kappa}\left[\frac{(kT_M)^2}{q_s^n} - \frac{(kT_M)^2}{q_s^o}\right] = \left[\frac{1+\nu}{\nu} q_s^n - \frac{1}{\nu} q_{in}\right] \Delta t \tag{2-148}$$

The solution for melting and freezing are identical and are obtained from the same set of equations. For this case, the conductive heat flux is no longer equal to the incident convective heat flux from the molten pool and the difference determines the movement of the interface

$$x_s^n = x_s^o + \frac{q_{in} - q_s^n}{\rho Q_M} dt \tag{2-149}$$

If the conductive heat flux, q_s^n, is less than the incident heat flux, q_{in}, the excess heat incident on the interface goes into the latent heat of ablation and the surfaces advances into the solid (i.e., x_s^n is greater than x_s^o). If the conductive heat flux is greater than the

incident heat flux, q_{in}, the latent heat released by freezing molten material onto the surface is conducted into the solid and correctly predicts that the surface is regressing such that x_s^n is less than x_s^o.

As was done for the stationary interface case, a dimensionless group, Y, is introduced into the equation to simplify the solution. It is substituted into

$$Y \equiv \frac{k(T_S - T_0)}{q_s} \tag{2-150}$$

along with the boundary condition, and the quadratic equation for Y^n is obtained:

$$(Y^n)^2 - \left[Y^o - \frac{\kappa \Delta t}{\nu\gamma}\frac{h(T_P - T_M)}{k(T_M - T_0)}\right]Y^n - (1+\nu)\frac{\kappa \Delta t}{\nu\gamma} = 0 \tag{2-151}$$

The calculated Y is solved for as

$$Y^n = B + \sqrt{B^2 + C} \tag{2-152}$$

where

$$2B \equiv Y^o - \frac{h(T_P - T_M)}{k(T_M - T_0)}\frac{\kappa \Delta t}{\nu\gamma} \tag{2-153}$$

$$C \equiv (1+\nu)\frac{\kappa \Delta t}{\nu\gamma} \tag{2-154}$$

and can then be used to generate the more physically meaningful variables such as the conductive heat flux, qs and the new solid surface temperature, Ts. as in:

$$x_s^n = x_s^o + \frac{q_{in} - q_s^n}{\rho Q_M}dt \quad q_s^n = \frac{k(T_M - T_0)}{Y^n} \tag{2-155}$$

.

2.4.3.3 Assessment of Integral Method

The integral method was tested by running a set of test cases and comparing calculated results with analytical solutions (where available) or results obtained from a finely noded finite difference calculation. Though the results of many of these calculations are not produced in this report, several cases are shown in Figure 2.7 to Figure 2.11. These plots compare the integral solution to the finite difference solution for a wide range of conditions for heat transfer coefficient, temperature gradient, and internal heat generation. Thermal properties for the molten pool and substrate were representative of conditions expected

for a reactor. The capability of the integral method for calculating results identical with analytical solutions and finely noded finite difference equations is well established.

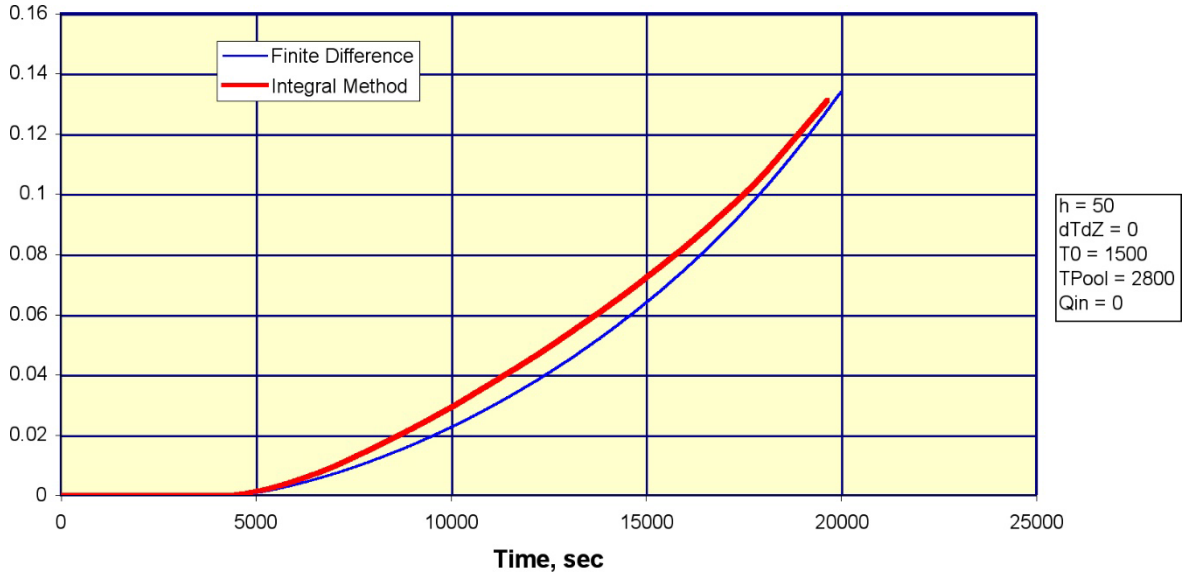


Figure 2.7 Ablation with high heat transfer coefficient

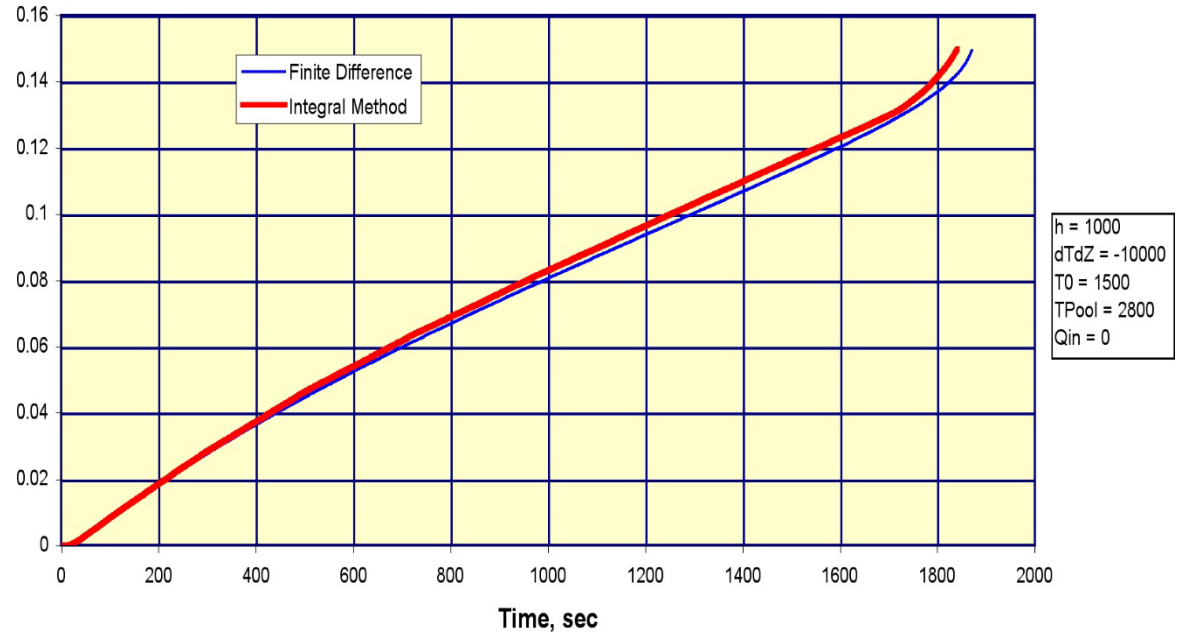


Figure 2.8 Ablation with low heat transfer coefficient

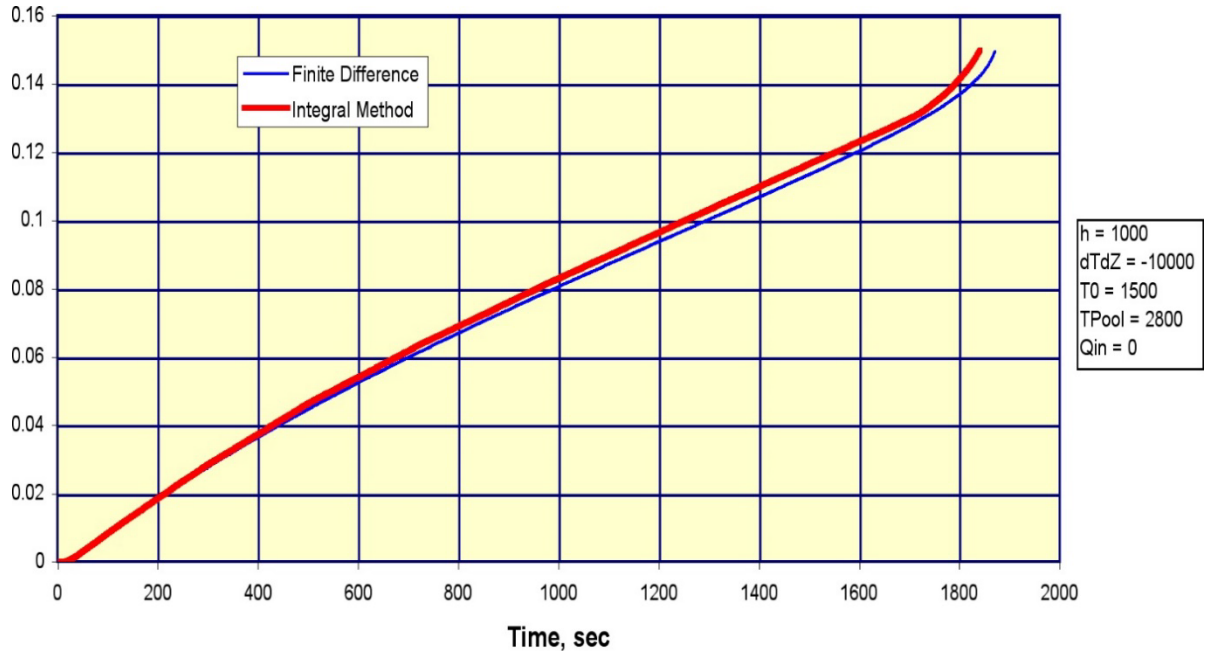


Figure 2.9 Ablation with steep negative temperature gradient

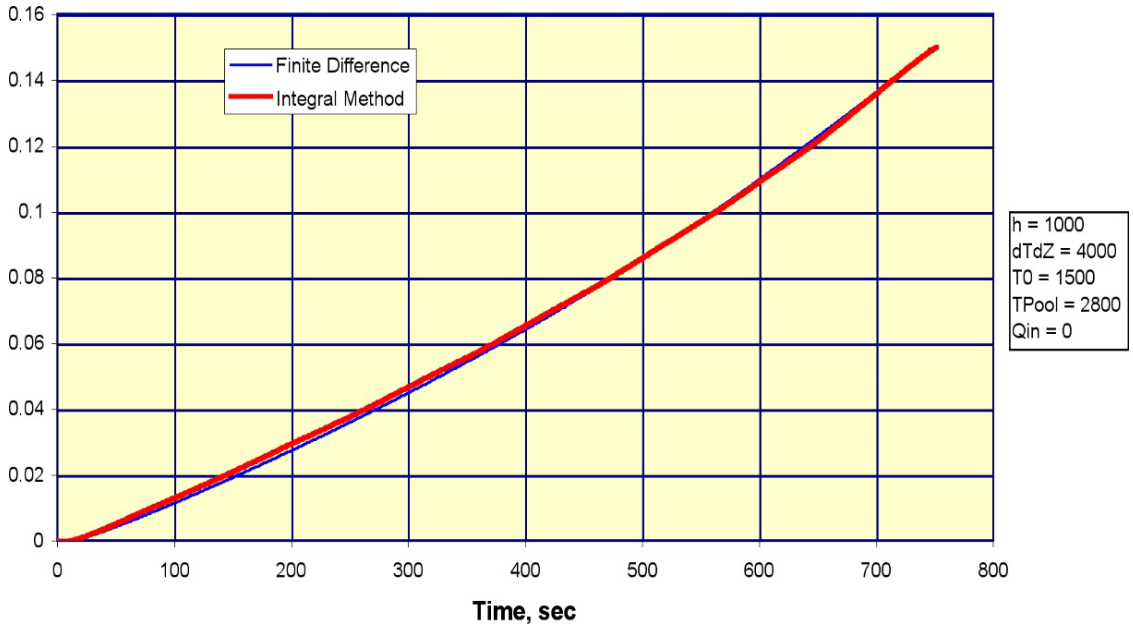


Figure 2.10 Ablation with steep positive temperature gradient

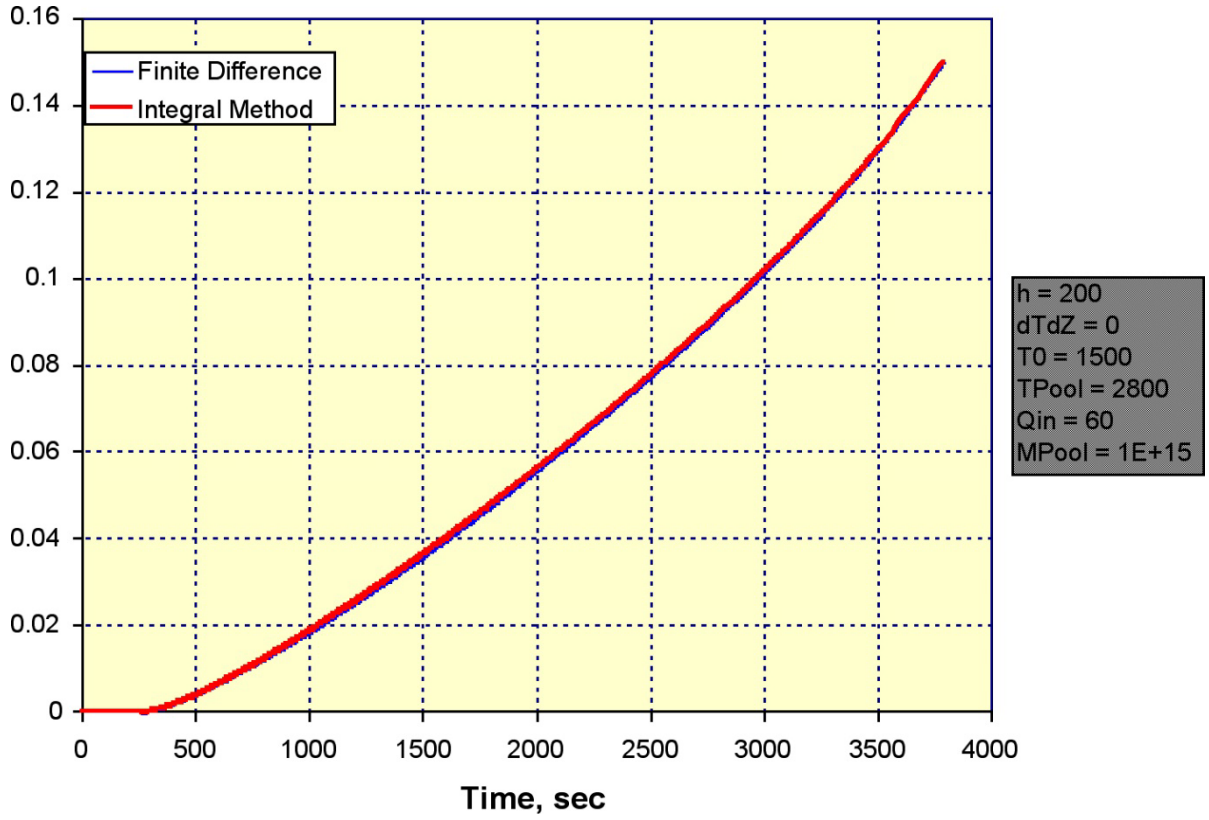


Figure 2.11 Ablation with internal heat generation

The Stefan model solution is implemented into MELCOR as depicted in Figure 2.12. A loop is performed over all physical pools and all substrate nodes in contact with each molten pool. For each node, thermal properties and variables tracked as part of the solution are recalled from the previous timestep. Phase transition properties are then evaluated for the molten pool and the substrate. The transient is then advanced through the solution of the integral finite difference equations, and ablation, freezing, or heating at the interface is determined. If the interface moves across a cell boundary, the meshing is modified. For the case of ablation, the boundary mesh is removed and for the case of freezing a boundary mesh is added while all remaining mesh are shifted. Finally, the sensible heat that is transferred to the substrate is partitioned among the mesh and the end of timestep properties are stored in the database.

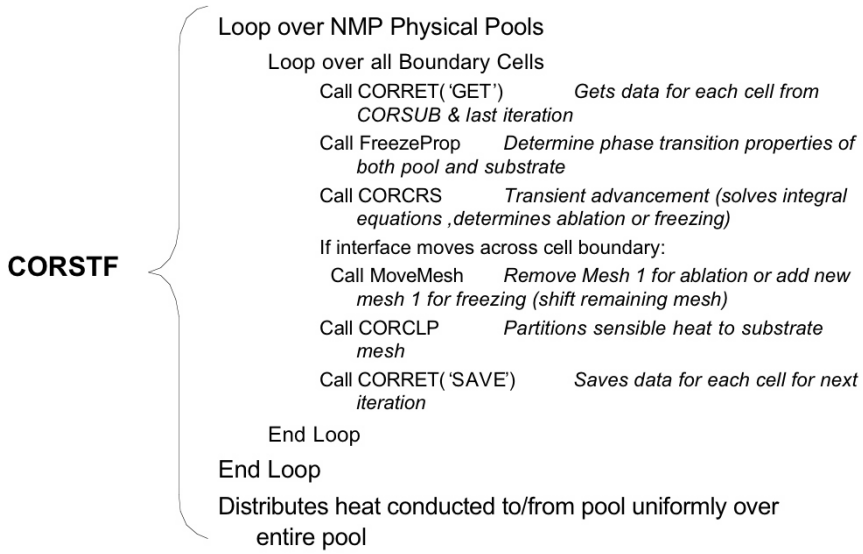


Figure 2.12 Subroutine CORSTF program flow diagram

The molten pool heat transfer is calculated immediately following the calculation of heat transfer and oxidation from other COR components and just before calculating heat transfer to the lower head. The sequence of calculation is important, and a discussion of the logical flow for this model is instructive. First, the molten pools are characterized, and thermal properties are evaluated for all pools that exist in the upper core or in the lower plenum. The existence of molten pool components is detected for each cell, and contiguous convecting molten pools are evaluated. Properties such as pool mass, composition, viscosity, thermal expansion coefficient, thermal conductivity, elevation of pool surfaces, and characteristic lengths are evaluated for each pool. Then, the solid substrate in contact with the molten pool is identified and evaluated. Substrate nodes, node thermal properties, node surface areas, volumes, and thicknesses are evaluated and saved for later use by the Stefan model solver.

Next, using molten pool properties previously evaluated, the transient convective heat transfer coefficients for each pool are evaluated. This is where the nested pool iterations previously discussed are performed. In addition, other heat losses that take place at pool surfaces, such as radiation and convection to the atmosphere or heat transfer between molten pools, are uniformly distributed throughout the pool volumes.

After the molten pools and solid substrate have been characterized and heat transfer coefficients for the molten pools are evaluated, the integral Stefan problem is solved for every interface between molten pool and its surroundings. Node dimensions diminish if material is ablated from the substrate or increase if material freezes. However, material transfer between molten pool components and particulate debris is postponed until temperatures are updated. Heat to be transferred between the convecting molten pool

and particulate debris components is calculated. The heat transferred to the lower head segments is also recorded and is included in the heat transfer calculation for the lower head (Section 6.1).

After updated temperatures are incorporated accounting for heat transfer from the molten pool to the substrate and surroundings, mass transfer of material between molten pool and substrate is accomplished, with the mass and energy conserved. Mass and energy added or removed from a molten pool is added or removed uniformly from the entire pool. Mass and energy added or removed from the substrate is added or removed locally.

2.5 Oxidation

MELCOR computes oxidation of COR materials such as zircaloy, stainless steel, aluminum, and graphite according to either built-in or user-specified oxidation chemistry according to a Generalized Oxidation Model (GOM). When built-in oxidation chemistry is invoked, no GOM user inputs are required. Oxidation of user-defined oxidizable COR materials as configured through the User Defined Materials (UDM) capability (see MP package users' guide) is also computed by the GOM. Built-in boron carbide models are available independent of GOM.

The GOM includes built-in data sets describing oxidation of Zircaloy by both steam (H_2O) and oxygen (O_2) as well as steel by H_2O. These data sets effectively invoke standard parabolic kinetics within the GOM arbitrary kinetics framework and include appropriate rate constant expressions for Zircaloy and steel. The GOM entails a built-in data set describing oxidation of aluminum by steam according to cubic kinetics. The GOM allows for oxidation rate limitation by gaseous diffusion.

There are two options for modeling the oxidation of B_4C, neither involving the GOM. The simple default model developed by ORNL for the MARCON 2.1B code [2] treats only oxidation by H_2O. It gives satisfactory results in oxidizing environments. However, in reducing environments the simple B_4C oxidation model tends to underpredict the methane generation rate which can lead to underestimation for release of volatile methyl iodide. Hence, the use of an optional advanced B_4C reaction model is recommended if reducing atmospheres (high hydrogen concentrations) are expected. The advanced B_4C oxidation model also includes the effects of O_2. The advanced B_4C model, also developed at ORNL, is used in the BWRSAR code, which is the successor to MARCON.

Irrespective of the modeling option, the B_4C reaction does not begin until the steel control blade sheaths have failed (i.e. B_4C is not exposed to steam until failure occurs). Failure is assumed to occur when the mass of intact steel in the control blade component falls below a user-specified fraction of its initial value (adjustable through sensitivity coefficient C1005 with a default value of 0.9). The intact steel is consumed by both steel oxidation and dissolution/melting. Following failure of the steel sheath, the oxidation reaction proceeds when steam or oxygen is present and the B_4C component temperature is above a user-adjustable threshold (sensitivity coefficient C1005, default value 1500 K). Both the

simple and advanced models can be used either with or without the eutectics model described in Section 2.8. If the eutectics model is active, then any B_4C that is dissolved in the eutectic mixture is regarded as unavailable for reaction. The user can arbitrarily limit the fraction of the initial mass of B_4C that is permitted to react. A default maximum reaction consumption fraction of 0.02, specified by sensitivity coefficient C1005, is based on experimental observations [27].

Graphite oxidation is available as a built-in model that does not leverage the GOM explicitly. An empirical model is used to describe graphite oxidation rate for a steam or air oxidant. An empirical model is also used to set the ratio of carbon monoxide to carbon dioxide byproduct.

Zircaloy oxidation is calculated for cladding, for both canister components, and for control rod guide tubes provided their respective components are Zircaloy; steel oxidation is calculated for the other structure (SS or NS) components. Both Zircaloy and steel oxidation are calculated for particulate debris. Oxidation of conglomerate debris (i.e., material that has melted and refrozen onto another existing component) is also modeled but may be selectively deactivated (on MELCOR input record COR_TST) independent of the oxidation of intact components. The oxidation models actived under the GOM framework use surface areas that account for the effects of conglomerate debris refrozen on the components; calculation of these surface areas is described in detail in Section 3.1.6. For BWR cores, oxidation of both sides of the canister walls (which may be exposed to differing environments) is modeled. A control function may be input on record COR_NOX to shut off oxidation on a cell-by-cell basis to simulate, for example, the effects of flow blockage. In addition, minimum and maximum oxidation cutoff temperatures have been implemented as sensitivity coefficient array C1004, with default values of 1100 K and 9900 K, respectively.

The effects of steam (or oxygen) starvation and flow blockage are simulated by explicitly considering the direction of flow within the CVH control volumes representing the core fluids (as determined by the dT/dz model setup described in Section 2.5) and by evaluating the unblocked flow area along the portion of the radial rings located within these CVH volumes. The allocation of steam and oxygen to the rings is based on the fraction of the total unblocked flow area of the CVH volume represented by each ring. Furthermore, oxidizers in each ring are partitioned among the surfaces of each COR cell (see Section 2.5.13) to remove any dependence of oxidation results on the order of surface processing. The partial pressures of steam and oxygen and the amounts available in the control volume interfaced to a COR cell are appropriately decreased, and, in the case of steam, the hydrogen partial pressure and mass are increased. These local gas concentrations are also used in the convection model to obtain local properties for the heat transfer correlations.

The oxidation of unquenched Zircaloy and steel surfaces that are below the pool surface is also considered. The necessary steam is assumed to come from the gas film between

the hot surface and the pool and is therefore limited only by the pool mass. (Optional input allows this model to be disabled for comparison purposes.)

2.5.1 Generalized Oxidation Model

The GOM entails arbitrary oxidation kinetics:

$$\frac{d(W^n)}{dt} = K(T) \tag{2-156}$$

where W is the mass of metal oxidized per unit surface area and *K(T)* is a rate constant expressed as an exponential function of surface temperature, *T*. Integrating analytically over a timestep $(t_f - t_i)$ and assuming a constant temperature (hence constant $K(T)$):

$$\left(W_{t_f}\right)^n = \left(W_{t_i}\right)^n + K(T_{t_i})(t_f - t_i) \tag{2-157}$$

Where,

$$K(T) = \begin{cases} 0 & T < T_{min,1} \\ A_i e^{\left(\frac{-B_i}{T}\right)} & T_{min,i} < T < T_{max,i} \\ \frac{K\left(T_{min,i+1}\right)\left(T - T_{max,i}\right) + K\left(T_{max,i}\right)\left(T_{min,i+1} - T\right)}{\left(T_{min,i+1} - T_{max,i}\right)} & T_{max,i} < T < T_{min,i+1} \end{cases} \tag{2-158}$$

Piecewise temperature sets $T_{min,i}$ to $T_{max,i}$ with corresponding A_i and B_i allow for multiple oxidation rates if necessary (e.g. useful for breakaway oxidation).

2.5.1.1 Oxidation Reactions and Reaction Sets

Oxidation of a given material is described by a reaction set which consists of one or more oxidation reactions. A general set of N reactions in the GOM can be written as:

(1) $$Mat_1 + (\chi_{oxidant})_1(Oxidant) \rightarrow (\chi_{oxide})_1(Oxide) + \left(\chi_{H_2}\right)_1(H_2) + (\chi_{CO})_1(CO) + \left(\chi_{CO_2}\right)_1(CO_2) + \left(\chi_{CH_4}\right)_1(CH_4) + Q_{ox_1}$$

…

(n) $$Mat_n + (\chi_{oxidant})_n(Oxidant) \rightarrow (\chi_{oxide})_n(Oxide) + (\chi_{H_2})_n(H_2) + (\chi_{CO})_n(CO) + (\chi_{CO_2})_n(CO_2) + (\chi_{CH_4})_n(CH_4) + Q_{ox_n}$$

…

(N) $$Mat_N + (\chi_{oxidant})_N(Oxidant) \rightarrow (\chi_{oxide})_N(Oxide) + (\chi_{H_2})_N(H_2) + (\chi_{CO})_N(CO) + (\chi_{CO_2})_N(CO_2) + (\chi_{CH_4})_N(CH_4) + Q_{ox_N}$$

---

overall $$Mat + (\chi_{oxidant})(Oxidant) \rightarrow (\chi_{oxide})(Oxide) + (\chi_{H_2})(H_2) + (\chi_{CO})(CO) + (\chi_{CO_2})(CO_2) + (\chi_{CH_4})(CH_4) + Q_{ox}$$

Within a reaction set defined under the GOM, each constituent reaction is normalized to one mole of oxidizing material:

$$Mat_n + (\chi_{oxidant})\,Oxidant \rightarrow (\chi_{oxide})\,Oxide + (\chi_{H_2})\,H_2 + (\chi_{CO})CO + (\chi_{CO_2})\,CO_2 + (\chi_{CH_4})\,CH_4 + Q_{ox} \quad (2\text{-}159)$$

Where,

Mat_n = Oxidizing material of n^{th} reaction

$Oxidant$ = Oxidant chemical symbol (e.g. O2 or H2O)

$Oxide$ = Solid oxide evolved as a byproduct of Mat oxidation by $Oxidant$

H_2, CO, CO_2, CH_4 = NCG gaseous byproducts of Mat oxidation by $Oxidant$

Q_{ox} = Oxidation energy associated with oxidation of Mat by $Oxidant$

χ_A = Stoichiometric coefficient of species A in reaction [mol A / mol Mat]

The stoichiometric/leading coefficients χ describe - per-unit mole oxidizing material according to a given reaction - the molar consumption of oxidant and molar production of oxide and gas for that reaction. These coefficients are default values when a built-in oxidation model is invoked through UDM/GOM. Alternatively, these coefficients can be user-specified. The noncondensible gases hydrogen, carbon monoxide, carbon dioxide, and methane are the only NCG materials allowed to evolve as products of any oxidation reaction. Note also that while a solid oxide phase is generally produced, it is not necessarily required to exist as some oxidation reactions may produce gaseous byproducts exclusively.

The simplest reaction set describes in one chemical reaction how products (plus oxidation energy) are evolved from reaction of one oxidizing material with one oxidant. Built-in one-equation reaction sets of Zircaloy and aluminum are examples. More complicated reaction sets might describe how the several n constituents Mat_n of an oxidizing material Mat – when exposed to oxidant – yield different products and oxidation energies. The built-in multi-equation reaction set for stainless steel is one such example. Still other reaction sets might describe the two or more outcomes possible when a given oxidizing material ($Mat_n = Mat, for\ all\ n$) is exposed to a given oxidant in different stoichiometric proportions. For example, graphite – when exposed to oxygen – can yield both carbon monoxide and carbon dioxide as products. In any case, reaction sets partition oxidation over the oxidizing material and govern production of NCG gases, oxides, and energy.

2.5.1.2 Reactant and Product Masses

For a given n^{th} reaction in a reaction set, if the oxidizing material mass that reacts according to that given reaction is known, one may readily compute the mass of oxidant consumed, the amount of oxide produced, and the mass of gaseous byproducts:

$$\left(\mathrm{M}_{\mathrm{P|R}}\right)_n = \left(\frac{\left(\chi_{\mathrm{P|R}}\right)_n \left(\mathrm{MW}_{\mathrm{P|R}}\right)(P|R)_n}{\mathrm{MW}_{Mat}}\right)\mathrm{M}_{Mat,n} \tag{2-160}$$

Or,

$$\left(\mathrm{M}_{\mathrm{P|R}}\right)_n = \left(\frac{\left(\chi_{\mathrm{P|R}}\right)_n \left(\mathrm{MW}_{\mathrm{P|R}}\right)(XR)_n}{\mathrm{MW}_{Mat}}\right)\mathrm{M}_{Mat,n} \tag{2-161}$$

Where,

n = Reaction identifier in a set, $n = 1 \ldots N$

$\mathrm{M}_{\mathrm{P|R}}$ = Mass of product (P) generated or reactant (R) consumed [kg $\mathrm{P|R}$]

$\chi_{\mathrm{P|R}}$ = Stoichiometric coefficient of P or R [mol $\mathrm{P|R}$ / mol Mat]

MW_X = Molecular weight of indicated substance X [g X / mol X]

M_{Mat} = Mass of oxidizing material consumed [kg Mat]

$P|R$ = Product or reactant scalar (introduced below) [-]

XR = Weight/mass fraction of given reaction (introduced below) [-]

A single chemical reaction may only describe how some fraction of the total oxidizing material Mat reacts in the context of a larger reaction set or may only describe how one constituent Mat_n of the total oxidizing material responds to an oxidant. Also, oxidants and solid/gaseous byproducts may appear in more than one reaction of a larger reaction set.

It is therefore useful to fractionally apportion reactants and products to each individual reaction of the set. This can be done in one of two ways with the GOM:

- Product and reactant scalars, or
- Reaction mass fractions and molecular weights of constituents and oxides
- These two methods are mutually exclusive. Either:
- Product and reactant scalars are given while reaction mass fractions are held at unity across the reaction set, or
- Reaction mass fractions are given while product and reactant scalars are held at unity across the reaction set

2.5.1.3 Product and Reactant Scalars

Product and reactant scalars help to ascertain mass ratios (of products/reactants to oxidizing material) that – when multiplied by the total mass of oxidizing material – yield the total amounts of oxidant consumed and product generated. With reference to a constituent n of a reaction set of N equations, product scalars P_n and reactant scalars R_n are defined by:

$$\begin{aligned} Mat + \ & (\chi_{oxidant})_n(Oxidant)(R_n) \\ & \rightarrow \ (P_n)\Big((\chi_{oxide})_n(Oxide) + (\chi_{CO})_n(CO) \\ & + \left(\chi_{CO_2}\right)_n(CO_2) + \left(\chi_{CH_4}\right)_n(CH_4) + (\bar{R})_n(H_2)\Big) \end{aligned} \tag{2-162}$$

The residual term $\bar{R}$ resolves hydrogen mass from the imbalance given all other reactants and products (i.e. hydrogen production is not a user input but is surmised from user input):

$$\bar{R}_n = \frac{M_{Mat} + M_{Oxidant} - M_{Oxide} - M_{CO} - M_{CO_2} - M_{CH_4}}{M_{H_2}} \tag{2-163}$$

The product scalar of reaction n can be interpreted as a scalar uniformly applied to all product stoichiometric coefficients (of reaction n) to ascertain total product generation from each n^{th} reaction. Likewise, the reactant scalar of reaction n can be interpreted as a scalar applied to the reactant (i.e. oxidant) stoichiometric coefficient (of reaction n) to ascertain oxidant consumption of each n^{th} reaction.

Product and reactant scalars are most useful when describing the competing oxidation reactions occurring for a single oxidizing material. An example is carbon (graphite) oxidation in air (oxygen), where a single elemental species (carbon) can oxidize to form either carbon monoxide or carbon dioxide.

2.5.1.4 Reaction Mass Fractions

Reaction mass fractions help to ascertain mass ratios (of products/reactants to oxidizing material) that – when multiplied by the total mass of oxidizing material – yield the total amounts of oxidant consumed and product generated. With reference to a constituent n of a reaction set of N equations, a reaction mass fraction should reflect the weight (or mass) fraction, XR_n [dimensionless] represented by the constituent material mat_n [kg] within the reacting material Mat [kg]. An oxidant would presumably encounter each constituent material in proportion to its weight (or mass) fraction. The total mass of the reacting material is represented by the sum of its constituents:

$$Mat = \sum_{n=1}^{N} XR_n * mat_n \tag{2-164}$$

When using the reaction mass fraction method, the user must carefully define the reaction rate constant(s) to account for all constituent materials. The reaction rate is computed by constituent, and values for A, B, and n are defined per reaction (or per constituent). Summing over all reactions/constituents captures the full picture of material oxidation. New-time oxidized mass is simply the sum over all contributions accounting for weight fraction:

$$\begin{aligned} \left(W_{t_f}\right)^n &= \sum_{n=1}^{N} \left[(XR)_k \left(W_{t_f,k}\right)^n\right] \\ &= \sum_{n=1}^{N} \left[(XR)_k \left(W_{t_i,k}^n + A(XR)_k{}^n e^{-B/T}\left(t_f - t_i\right)\right)\right] \end{aligned} \tag{2-165}$$

Note the difference between this case and that for which product and reactant scalars are recommended. Here, the oxidation of a material (e.g. stainless steel) is described by a set of oxidation reactions each having unique oxidizing material constituents and oxides (and possibly other byproducts). The unique oxidizing material constituents comprise some weight fraction of the oxidizing material. Note molar masses of the material constituents and their oxides must be known.

2.5.2 Oxidation Energy

Oxidation energy from any given reaction is either prescribed by the user (GOM MELGEN inputs) or is assumed (built-in oxidation models). Under the GOM framework, oxidation energy is computed in view of reactant/product scalars or reaction mass fractions.

Reaction energies are calculated for COR materials from the enthalpies of the reactants and products. Because the equations of state used for the core materials currently do not have reference points consistent with each other or with the CVH and NCG equations of

state for fluid materials, the following treatment must be used to obtain the reaction energies for arbitrary temperature T given reference temperature T_0 :

$$Q_{ox}(T) = Q_{ox}(T_0) + H_{rp}(T) - H_{rp}(T_0) \quad (2\text{-}166)$$

$$H_{rp}(T) = H_r(T) - H_p(T) \quad (2\text{-}167)$$

Where

Q_{ox} = Reaction energy generated,

H_{rp} = Enthalpy of reactants (r), and products (p)

The heats of reaction are provided below for all oxidizable default materials.

Table 2-5 Default heats of reaction

MATERIAL	Oxidant	Heat of Reaction
ZIRCALOY	Steam	5,797 kJ/kg$_{Zr}$
STAINLESS-STEEL	Steam	$\frac{-249.5X_{Fe} + 2{,}442X_{Cr}}{X_{Fe} + X_{Cr}} kJ/kg_{SS}$
ALUMINUM	Steam	15,180 kJ/kg$_{Al}$
GRAPHITE	Steam	-10,850 kJ/kg$_{gr}$
ZIRCALOY	Air	12,065 kJ/kg$_{Zr}$
STAINLESS-STEEL	Air	$\frac{4{,}869X_{Fe} + 1{,}091X_{Cr}}{X_{Fe} + X_{Cr}} kJ/kg_{SS}$
ALUMINUM	Air	3,106 kJ/kg$_{Al}$
GRAPHITE	Air	9,273 kJ/kg$_{gr}$ (CO) 3,280.7 kJ/kg$_{gr}$(CO_2)

2.5.3 Gaseous Diffusion Limitation

For very low oxidant concentrations, gaseous diffusion may limit the reaction rate. A mass transfer coefficient is calculated via a heat-mass transfer analogy from the heat transfer correlations in Section 2.3 by substituting the Schmidt number for the Prandtl number and

the Sherwood number for the Nusselt number. The oxidation rate when limited by gaseous diffusion is given by:

$$\frac{d\,W}{dt} = \frac{MW \;\; k_c \;\; P_{ox}}{n \;\; R \;\; T_f} \tag{2-168}$$

where

MW = Molecular weight of metal being oxidized,

k_c = Mass transfer coefficient,

P_{ox} = Partial pressure of oxidant (H_2O or O_2),

n = Number of moles of oxidant (H_2O or O_2) consumed per mole of metal,

R = Universal gas constant, and

T_f = Gas film temperature, $(T \; + T_{gas})$ / 2

The gaseous diffusion oxidation rate is used if it is less than the rate calculated by Equation (2-158). Although the molecular weight MW and the number of moles n of H_2O consumed are defined by the reaction, the quantity (MW/nR) has been implemented for reactions with H_2O as sensitivity coefficient array C1003 to allow the user a measure of separate control over the gaseous diffusion oxidation rate. That sensitivity coefficient is multiplied by two internally in the code to obtain an equivalent value for gaseous diffusion of oxygen ($n_{H2O} = 2n_{O2}$).

2.5.4 Zircaloy Oxidation

Zircaloy has built-in oxidation models consisting of one reaction per oxidant. They are trivial in terms of the GOM framework, i.e. the product and reactant scalars are identically one or, alternatively, zircaloy has a weight fraction of unity with known molar mass for Zr and ZrO_2. Steam oxidation behaves according to:

$$Zr \; + \; 2 \; H_2 \, O \; \rightarrow \; ZrO_2 \; + \; 2 \, H_2 \; + \; Q_{ox} \tag{2-169}$$

While air oxidation reaction behaves according to:

$$Zr \; + \; O_2 \; \rightarrow \; ZrO_2 \; + \; Q_{ox} \tag{2-170}$$

A parabolic model (n=2 under GOM framework) as described in equation (2-158) is invoked for zircaloy oxidation kinetics, where default reaction rate parameters for steam and oxygen are based on the Urbanic-Heidrich model and are provided in Table 2-6. These coefficients are accessible to the user through sensitivity coefficients as described in the COR package user guide.

Table 2-6 Zircaloy oxidation kinetics parameters

Material	Oxidant	T_{min}	T_{max}	A	B	n
Zircaloy [28]	Steam	1100.0	1853.0	29.6	16820.0	2
Zircaloy	Steam	1873.0	9900.0	87.9	16610.0	2
Zircaloy [29]	Oxygen	1100.0	10000.0	50.4	14630.0	2

Heats of reaction are given previously in Table 2-5. For oxidation in environments containing both H_2O and O_2, the maximum calculated oxidation rate is used:

$$dW/dt = MAX\left[(dW/dt)_{H_2O}, (dW/dt)_{O_2}\right] \tag{2-171}$$

There are two options for partitioning oxidant consumption. The default (recommended) does not permit consumption of steam until all available oxygen is consumed. This option is equivalent to assuming that all hydrogen produced by steam oxidation is instantaneously converted back to steam by combustion with available oxygen. The default option should prevent timestep reductions associated with the normal combustion of in-vessel hydrogen by the BUR package. For the second option, the reactions given by Equations (2-169) and (2-170)are proportioned by relative rates:

$$f_{H_2O} = \left(\frac{dW}{dt}\right)_{H_2O} \Big/ \left(\left(\frac{dW}{dt}\right)_{H_2O} + \left(\frac{dW}{dt}\right)_{O_2}\right) \tag{2-172}$$

And

$$f_{O_2} = 1 - f_{H_2O} \tag{2-173}$$

2.5.5 Optional Zircaloy Oxidation Models

As previously noted, the parameters for the reaction kinetics term are fully accessible to the user via control functions, making it possible to customize the reaction rate to replicate standard oxidation models other than the default Urbanic-Heidrich model. However, for user convenience, there are additionally a number of standard oxidation models that are easily accessed by the user through alternate input where the user specifies the model desired. These alternate models are available through the COR_OX record as described in the COR user guide. The equation form is nearly identical to equation (2-158) which is reproduced in equation (2-19) below where the coefficients are provided in Table 2-7 and Table 2-8.

$$K(T) = \begin{cases} A_1 e^{\left(\frac{-B_1}{T}\right)} & T < T_1 \\ \left(A_1 e^{\left(\frac{-B_1}{T_1}\right)} \cdot (T_2 - T) + A_2 e^{\left(\frac{-B_2}{T_2}\right)} \cdot (T - T_1) \right) / (T_2 - T_1) & else \\ A_2 e^{\left(\frac{-B_2}{T}\right)} & T_2 < T \end{cases} \quad (2\text{-}174)$$

Table 2-7 Optional models for oxidation of zircaloy in steam

Model	T_1	T_2	A1	B1	A2	B2
Cathcart-Pawel/ Urbanic-Heidrick	1853	1873	294.2	20111	87.9	16610
Leistikov-Schanz/Prater-Courtright	1800	1900	426.0	20953.	26778.5	26461.
Leistikov	1100	9900	426.01	20953.	426.01	20953.
Urbanic-Heidrick	1853	1873	29.6	16820.	87.9	16610.
Sokolov	1763	1783	1291.2	23040.	798.5	20800.
Grosse*	1223	1273	6633.6	25310	72.2	18066.6

*Note that above 1673K, the Cathcart-Pawel/Urbanic-Heidrick model is employed

Table 2-8 Optional models for oxidation of zircaloy in air

Model	T_1	T_2	A1	B1	A2	B2	A3	B3
*Hoffman-Birchley	1363	1383	25494.4	25792.	74.8	17895.	-	-
Hayes-Roberson/Leistikov-Berg	1173	1523	0.935	13759.	46819.6	26668.	50.39	14630.
Powers	273	273	251124	28485	30898.8	28485	-	-
MELCOR	273	273	50.4	14630.	50.4	14630.	-	-
Mozart	1263	1283	199.8	21507	1.83E9	41449.	-	-

*Note that above 1623K, the Cathcart-Pawel/Urbanic-Heidrick model is employed

Note that for the Hayes-Roberson correlation there are additional coefficients and the equation takes the slightly modified form shown in (2-175)

$$K(T) = \begin{Bmatrix} A_1 e^{\left(\frac{-B_1}{T}\right)} & T < T_1 \\ min\left(A_2 e^{\left(\frac{-B_2}{T}\right)}, A_3 e^{\left(\frac{-B_3}{T}\right)}\right) & T_1 \le T \le T_2 \\ max\left(A_1 e^{\left(\frac{-B_1}{T}\right)}, A_2 e^{\left(\frac{-B_2}{T}\right)}\right) & T_2 < T \end{Bmatrix} \quad (2\text{-}175)$$

2.5.6 Stainless Steel Oxidation

Stainless Steel has a built-in oxidation model consisting of a reaction set – one per constituent element of stainless steel to include iron, chromium, nickel, and carbon. Under the GOM framework, stainless steel is a perfect candidate for oxidation modeling via the reaction weight fraction route. For the built-in model, these constituent/reaction weight fractions are obtained from MP package input/defaults. For steam oxidation, the sequence of reactions is:

$$Fe + H_2O \rightarrow FeO + H_2 + Q_{ox} \quad (2\text{-}176)$$

$$Cr + \frac{3}{2} H_2O \rightarrow \frac{1}{2} Cr_2O_3 + \frac{3}{2} H_2 + Q_{ox} \quad (2\text{-}177)$$

$$Ni + H_2O \rightarrow NiO + H_2 + Q_{ox} \quad (2\text{-}178)$$

$$C + H_2O \rightarrow CO + H_2 + Q_{ox} \quad (2\text{-}179)$$

For parabolic kinetics, the rate constant for stainless-steel oxidation in steam is parameterized as:

Table 2-9 Stainless steel reaction rate constant parameters

Material	Oxidant	T_{min}	T_{max}	A	B	n
SS [30]	Steam	1100.0	9900.0	2.42e9	42400.0	2

Heats of reaction are given previously in Table 2-5.Though separate elements comprise the constituents of stainless steel, neither these constituents, (Fe, Cr, Ni, or C) nor their solid oxides (FeO, Cr2O3, and NiO) exist as material within the COR package. The gaseous product CO, however, exists along with H2 and CH4 within the NCG package. CO mass produced by reacting C is passed to the CVH package. Oxidation of the material - stainless-steel - produces the oxide material stainless-steel oxide in the COR package.

2.5.7 Aluminum Oxidation

Aluminum has a built-in oxidation model consisting of one reaction for steam oxidation. This model is trivial in terms of the GOM framework, i.e. the product and reactant scalars are identically one or, alternatively, aluminum has a weight fraction of unity with known molar mass for Al and Al_2O_3. Steam oxidation behaves according to:

$$Al + \frac{3}{2} H_2 O \rightarrow \frac{1}{2} Al_2 O_3 + \frac{3}{2} H_2 + Q_{ox} \quad (2\text{-}180)$$

A cubic model (n=3 under GOM framework) is invoked for aluminum oxidation kinetics, and reaction rates are parameterized as:

Table 2-10 Aluminum reaction rate parameters

Material	Oxidant	T_{min}	T_{max}	A	B	n
Aluminum	Steam	1100.0	9900.0	66.667	36990.0	3

2.5.8 Graphite Oxidation

Built-in graphite oxidation models are based on work by Richards at INL[31, 32]. These models were extensively compared to data by Gelbard [33] and were found to give good agreement. Models exist for steam and air oxidation.

2.5.8.1 Air Oxidation

Two air oxidation reactions are considered:

$$C + \frac{1}{2} O_2 \rightarrow \mathrm{CO} + Q_{ox} \quad (2\text{-}181)$$

$$C + O_2 \rightarrow \mathrm{CO_2} + Q_{ox} \quad (2\text{-}182)$$

The intrinsic reaction rate for oxidation of reactor-grade graphite by oxygen is:

$$R_{ox} = 1.7804 * 10^4 * e^{\left[\left(\frac{-20129}{T}\right)\left(\frac{P}{0.21228*10^5}\right)^{1/2}\right]} \quad (2\text{-}183)$$

Where

R_{ox} = Reaction rate [1/s],

T = Temperature [K],

P = Oxygen partial pressure [Pa]

Note this reaction rate requires no multiplication by surface area to ascertain graphite consumption rate. This equation for the reaction rate has been compared to numerous sets of data for the temperature range 800-1050 K and gives good agreement. Results are similar to those from the equation reported by Fuller and Okoh [34] for IG-110 graphite for the range 720-1020 K in [35].

Graphite oxidizes with O_2 to produce both CO and CO_2. The mole ratio of these reaction products varies with temperature. CO_2 dominates at low temperatures and CO dominates at higher temperatures. The CO/CO_2 ratio is given by an empirical model [36]:

$$f_{CO/CO_2} = 7396 * e^{(-69604/RT)} \quad (2\text{-}184)$$

Where

f_{CO/CO_2}= CO/CO_2 mole ratio

R = Gas constant = 8.314 [J/mol-K]

T = Temperature [K]

The overall graphite oxidation reaction can be written as

$$1\,C + \chi_{O_2} O_2 \rightarrow \chi_{CO} CO + \chi_{CO_2} CO_2 \quad (2\text{-}185)$$

Where

χ_{O_2} = Moles of O2 per mole C oxidized

χ_{CO} = Moles of CO per mole C oxidized

χ_{CO_2} = Moles of CO2 per mole C oxidized

Assuming chemical equilibrium at the graphite surface temperature, a balance gives:

$$\chi_{O_2} = \frac{2 + f_{CO/CO_2}}{2\left(1 + f_{CO/CO_2}\right)} \quad (2\text{-}186)$$

$$\chi_{CO} = \frac{f_{CO/CO_2}}{1 + f_{CO/CO_2}} \quad (2\text{-}187)$$

$$\chi_{CO_2} = \frac{1}{1 + f_{CO/CO_2}} \tag{2-188}$$

Built-in graphite oxidation does not directly exercise the GOM inputs but could serve as an illustration of the product/reactant scalar method in the GOM framework. There is a reaction set consisting of two equations that share an oxidizing material (C) and an oxidant (O_2). Each yields a unique gaseous product (with no solid oxide products). Including product and reactant scalars into the reaction set with each reaction normalized to one unit (e.g. mole) of oxidizing material:

$$C + R_1 \mathrm{N}_{O_{2,1}} O_2 \rightarrow P_1 N_{CO,1} \mathrm{CO} \tag{2-189}$$

$$C + R_2 \mathrm{N}_{O_{2,2}} O_2 \rightarrow P_2 N_{CO_2,2} \mathrm{CO}_2 \tag{2-190}$$

Where the stoichiometric coefficients are $N_{O_2,1} = 1/2$; $N_{CO,1} = 1$; $N_{O_2,2} = 1$; $N_{CO_2,2} = 1$

In the aggregate (combining oxidation reactions), 1 mole of C oxidized results in:

- χ_{CO} moles CO produced (by reaction (1) uniquely)
- χ_{CO_2} moles CO_2 produced (by reaction (2) uniquely)
- χ_{O_2} moles O_2 consumed (by both reactions (1) and (2))

Such that:

$$C + \chi_{O_2} O_2 \rightarrow \chi_{CO} CO + \chi_{CO_2} \mathrm{CO}_2 \tag{2-191}$$

Product scalars in this situation can be surmised from their definition by inspection since only reaction (1) produces CO and only reaction (2) produces CO_2. The CO that is produced from graphite oxidation comes from reaction (1), and likewise all CO_2 that is produced from graphite oxidation come from reaction (2). Per 1 mole of graphite oxidized, a total molar CO inventory χ_{CO} is produced, and the only stoichiometric coefficient on CO is in reaction (1) as $N_{CO,1} = 1$. Then, the statement to satisfy is $P_1 N_{CO,1} = \chi_{CO}$ which is true for $P_1 = \chi_{CO}$. By similar reasoning, $P_2 N_{CO_2,2} = \chi_{CO_2}$ which is true for $P_2 = \chi_{CO_2}$. Note the molar units on P_1 could be interpreted as $mol\ CO/mol\ C$ and the molar units on P_2 could be interpreted as $mol\ CO_2/mol\ C$.

Reactant scalars are less obvious in this situation because the oxidant (reactant) is common to reactions (1) and (2). From the O_2 stoichiometric balance overall:

$$\chi_{O_2} = 2\chi_{CO} + \chi_{CO_2} \tag{2-192}$$

Thus, per 1 mole of graphite oxidized, χ_{O_2} moles of oxygen are consumed in all with an amount $2\chi_{CO}$ consumed along (1) and an amount χ_{CO_2} consumed along (2). In terms of molar units:

$$R_1\left[\frac{mol\ O_{2,1}}{mol\ C}\right] = \left(2\chi_{CO}\left[\frac{mol\ O_{2,1}}{mol\ O_2}\right]\right)\left(\chi_{O_2}\left[\frac{mol\ O_2}{mol\ C}\right]\right) \quad (2\text{-}193)$$

$$R_2\left[\frac{mol\ O_{2,2}}{mol\ C}\right] = \left(\chi_{CO_2}\left[\frac{mol\ O_{2,2}}{mol\ O_2}\right]\right)\left(\chi_{O_2}\left[\frac{mol\ O_2}{mol\ C}\right]\right) \quad (2\text{-}194)$$

Note that:

$$R_1 N_{O_2,1} + R_2 N_{O_2,2} = \left(2\chi_{CO}\chi_{O_2}\right)\left({}^1/_2\right) + \left(\chi_{CO_2}\chi_{O_2}\right)(1) = \chi_{O_2}\left(\chi_{CO} + \chi_{CO_2}\right) = \chi_{O_2}$$

Since the C stoichiometric balance overall dictates $\chi_{CO} + \chi_{CO_2} = 1$

2.5.8.2 Steam Oxidation

One steam oxidation reaction is considered:

$$C + H_2O \ \rightarrow \mathrm{CO} + H_2 + Q_{ox} \quad (2\text{-}195)$$

The intrinsic reaction rate for oxidation of reactor-grade graphite by steam is:

$$R_{ox} = \frac{k_4 P_{H_2O}}{1 + k_5 {P_{H_2}}^{1/2} + k_6 P_{H_2O}} \quad (2\text{-}196)$$

Where

R_{ox} = Reaction rate for steam oxidation [1/s]
P_{H_2O} = Partial pressure of H_2O [Pa]
P_{H_2} = Partial pressure of H_2 [Pa]
k_i = i^{th} reaction coefficient, $K_i e^{\left({}^{-E_i}/_{RT}\right)}$
K_i = i^{th} precoefficient
E_i = i^{th} activation energy [J/kmol]
R = Gas constant = 8314 [J/kmol-K]
T = Temperature [K]

Pre-coefficients and activation energies are given below for steam oxidation[37]

Table 2-11 Default coefficients for steam oxidation of graphite

	K_i	E_i/R
k_4	2.646x10 $^{-4}$ [1/s-Pa]	16455 K
k_5	1.075x10 $^{-12}$ [1/Pa1/2]	30596 K
k_6	4.887x10 $^{-21}$ [1/Pa]	20129 K

2.5.9 Simple Boron Carbide Reaction Model

In the simple default *B4C* reaction model, the *B4C* in BWR control blades reacts with steam, using the model from MARCON 2.1B [2]. This model uses three reaction equations:

$$B_4C + 7H_2O \rightarrow 2B_2O_3 + CO + 7H_2 + Q_1 \quad (2\text{-}197)$$

$$B_4C + 8H_2O \rightarrow 2B_2O_3 + CO_2 + 8H_2 + Q_2 \quad (2\text{-}198)$$

$$B_4C + 6H_2O \rightarrow 2B_2O_3 + CH_4 + 4H_2 + Q_3 \quad (2\text{-}199)$$

Chemical equilibrium of reaction products is assumed, and the model uses the steam and hydrogen partial pressures and *B4C* temperature to determine the relative extent of each reaction. The equilibrium *CO/CO2* and *CO/CH4* mole ratios, $y_{CO/CO2}$ and $y_{CO/CH4}$ respectively, are given by the expressions:

$$y_{CO/CO_2} = \frac{P_{H_2}}{P_{H_2O}} \exp\left[\frac{-3605.0}{T} + 3.427\right] \quad (2\text{-}200)$$

$$y_{CO/CH_4} = \frac{P_{H_2O}}{(P_{H_2})^3} \exp\left[\frac{-27350.0}{T} + 30.50\right] \quad (2\text{-}201)$$

where the steam and hydrogen partial pressures are in atmospheres. The extents of reactions (2-197)

$$B_4C + 7H_2O \rightarrow 2B_2O_3 + CO + 7H_2 + Q_1 \quad (2\text{-}197)$$

– (2-199), expressed as relative percentages of *CO*, *CO2*, and *CH4* produced (x_{CO}, x_{CO2}, and x_{CH4}, respectively), can then be given in terms of the *CO*/*CO2* and *CO*/*CH4* mole ratios as

$$x_{CO} = \frac{1}{1 + 1/y_{CO/CO_2} + 1/y_{CO/CH_4}} \quad (2\text{-}202)$$

$$x_{CO_2} = x_{CO}/y_{CO/CO_2} \quad (2\text{-}203)$$

$$x_{CH_4} = 1 - x_{CO} - x_{CO_2} \quad (2\text{-}204)$$

The reaction energies (in J/kg-mole B_4C reacted) for reaction Equations (2-197)–(2-199) are given by the equations:

$$Q_1 = 8.238(10^8) - 58380.0\ T \quad (2\text{-}205)$$

$$Q_2 = 8.674(10^8) - 67060.0\ T \quad (2\text{-}206)$$

$$Q_3 = 1.056(10^8) - 61430.0\ T \quad (2\text{-}207)$$

The gaseous reaction products are transferred to the CVH package, while the B_2O_3 generated is transferred to the Radionuclide (RN) package as an aerosol. All of the energy generated by the *B_4C* reaction is added to the CVH package. The reaction energies calculated by Equations (2-205) to (2-207) are inconsistent with reaction energies that would be calculated using the present equations of state for the noncondensible gases and the *B_4C* and *B_2O_3* (i.e., the temperature dependence implied by those equations is not consistent with the actual temperature dependence of the equations of state used). This discrepancy is ignored at present, due to the lack of reliable enthalpy data for *B_4C* and *B_2O_3*.

The *B_4C* oxidation rate is given as a fractional change per second in the initial (intact) B_4C mass by

$$\frac{d\left(M_{B_4C}/M^o_{B_4C}\right)}{dt} = \frac{9.973(10^6)}{60.} \exp\left(\frac{-22647.2}{T}\right) \quad (2\text{-}208)$$

The constants in Equation (2-208) are programmed as sensitivity coefficients C1006. Given the amount of *B_4C* reacted, the amounts of the various products are calculated from Equations (2-197) – (2-199).

2.5.10 Advanced Boron Carbide Reaction Model

In the optional advanced *B_4C* reaction model, the *B_4C* in BWR control blades is reacted with vapors in the surrounding atmosphere using the model from BWRSAR and SCDAP [38]. This model determines the equilibrium composition in each control volume that is achieved when the Gibbs free energy of the system is minimized. The difference between the initial composition in the control volume and the equilibrium composition determines the rate of consumption of the reactants. The algorithm that is used to determine the

composition that minimizes the free energy is based on the Swedish SOLGASMIX computer code [39]. In this method, the quantity

$$\frac{G}{RT} = \sum_i n_i \left[\left(\frac{g^o}{RT} \right)_i + ln(a_i) \right] \quad (2\text{-}209)$$

is minimized with respect to the variables n_i for constant temperature and pressure values, where G denotes the total free energy of the system, R the gas constant, T the thermodynamic temperature, n_i the number of moles of the i^{th} species, g^o the standard chemical potential, and a_i the activity. The values for n_i corresponding to the equilibrium mixture must be non-negative, and the mass balance constraints must be satisfied. Lagrange's method of undetermined multipliers is used for determining the constrained minimum, and the logarithmic equations thus obtained are expanded in a Taylor series about initially estimated n_i-values, neglecting terms involving second- and higher-order derivatives. The linear equations represent approximations of the exact expressions, and so a series of iterations is performed to obtain the final solution.

The advanced B_4C reaction model assumes that chemical equilibrium is achieved between the reactants during each timestep. The mass of reactants considered during each timestep is linearly dependent on the size of the timestep, so that as the timestep size goes to zero, the rate of reaction goes to zero. The mass of B_4C available for reaction during each timestep is determined by Equation (2-208) as in the simple model. The availability of all other reactants is limited by the rate of steam diffusion to the reaction surface during the given timestep. For example, if only 5% of the steam in the control volume can diffuse to the surface during the timestep, then only 5% of all the other reactants in the control volume (except B_4C) are considered to be available for equilibration with the steam (5% of control volume total) and B_4C. The model considers 18 species that contain one or more of the five elements: argon, oxygen, hydrogen, boron and carbon (argon occurs only in elemental form and is included for simulation of fuel damage experiments that employ this inert gas). The 18 species are as follows:

H_2 (g)	CO_2 (g)	B_2O_3 (P,s)
H_2O (g)	B_4C (s)	$H_3B_3O_6$ (g)
C (s)	O (g)	HBO_2 (g)
CO (g)	B_2O_3 (g)	BH_3 (g)
CH_4 (g)	Ar (g)	B_2H_6 (g)
O_2 (g)	B (s)	BOH (g)

The quantity g^o/RT is determined from the thermodynamic relationship g = h – Ts. The enthalpy h and entropy s are calculated as integrals of the specific heat capacity,

$$h = \int_{298}^{T} C_p dT' + h_{298}^0 \quad (2\text{-}210)$$

$$s = \int_{298}^{T} \frac{C_p}{T'} dT' + s_{298}^0 \quad (2\text{-}211)$$

and the specific heat capacity for each species is expressed as a function of temperature over various temperature ranges:

$$c_p = a + b\,T + c\,T^2 + d/\,T^2 + e/\,T^3 \quad (2\text{-}212)$$

Deviations from ideality are not modeled, so that the activities of all gaseous species are equal to their respective mole fractions in the gas mixture. The condensed species are treated as a mechanical mixture only; their activities are set to unity, and they have no effect on the minimization of the Gibbs free energy in Equation (2-209).

Because a thermochemical reference is used, the heat of reaction is simply the difference between the total enthalpy of the products and that of the reactants.

The reaction products are passed to either the CVH or RN package for tracking and subsequent use as input to the chemical equilibrium routine. Steam, oxygen, hydrogen, carbon monoxide, carbon dioxide, methane and argon are tracked by the CVH package, while elemental boron and carbon and all the boron compounds are tracked by the RN package.

2.5.11 Boron Carbide Control Rod Oxidation Model

The improved rod oxidation model is intended to represent the release and oxidation of B_4C from control rods in a more mechanistic manner than the existing B_4C oxidation model. The release and oxidation scenario reflected in the improved boron carbide oxidation model was taken from observations noted in several experimental reports on B_4C oxidation. [40,41,42] These reports describe a quite different scenario than the existing MELCOR model.

2.5.11.1 Description of Control Rods and Failure Scenario

The B_4C control rod is constructed something like a fuel rod, with boron carbide pellets clad in a stainless-steel (SS) sheath filled with helium. The control rod is inserted in a guide tube made of stainless steel or Zircaloy.

When the control rod heats up to about 1500 K, the boron carbide starts to form a eutectic with the SS clad. This generally causes the control rod clad to fail at around 1500 K instead of 1700 K, the SS melting point. The eutectic then drains down the outside and

inside of the control rod sheath, stopping the oxidation as the eutectic drains to cooler parts of the core. There is typically enough SS to liquefy most of the boron carbide at a given axial location.

2.5.11.2 Model Implementation

The revised boron carbide oxidation model represents the formation of the SS-B4C eutectic by modifying the material properties tables for B4C; specifically, the pure B4C melting point of 2600 K is reduced to an effective eutectic liquefaction temperature of 1700 K to represent the temperature at which eutectic interaction between the SS-cladding and the B4C pellet produces gross slumping and relocation of the control material. Upon reaching this temperature, the oxidizing control material relocates by candling to lower regions where temperatures are lower, effectively terminating further oxidation. Before this, the boron carbide can oxidize when temperatures exceed 1500 K, the temperature at which the boron carbide pellet is assumed to be exposed to steam upon first liquefactions induced by the eutectic reaction. In other words, boron carbide oxidation is allowed to commence when temperatures exceed 1500 K, and the material is allowed to fully liquefy and relocate when the temperature exceeds 1700 K. This treatment more mechanistically represents the physical processes affecting boron carbide oxidation than does the simple fractional oxidation treatment in the previous model.

2.5.12 Comparison to Experiment

2.5.12.1 TestDemo Comparison

The new B_4C model was tested using the MELCOR testdemo input deck. Comparison of results was made with (a) the original model, (b) the original model with the maximum oxidation fraction set via C1031 to 0.9999, and (c) the new model. Because this test was done using the testdemo deck, the CO/CO_2 production, which would normally be a good diagnostic, is dominated by release from core-concrete interactions. Therefore, Class 13, B_2O_3, was used as the comparison reaction product.

The original model produces a total of 50 kg of B_2O_3, which is released over the period from the beginning of rod failure at 1200 s until failure of the core support plate at 4200 s. Using C1005(2) to set the maximum oxidation fraction up to 0.9999 from 0.02 increases the total B_2O_3 produced to 662 kg, or 20% oxidation of the B_4C on a boron mole basis. This is again released in the period from1200 to 4200 s. The new model releases a total of 610 kg of B_2O_3, or 18%, but the timing is different—most of the release occurs from when the rod temperature reaches the lower oxidation point of 1500 K at around 1000 s until control rod sheath failure occurs at 2000 s. While the results obtained by the revised model applied to the simple “testdemo” problem are not too different from that obtained by allowing 99.99% oxidation of the boron carbide locally, the revised model is believed

to arrive at this result from a more mechanistic standpoint. This observation may not apply to all problems. Some timing differences are also noted.

2.5.12.2 QUENCH-07 Comparison

A second test of the new model was performed using a comparison to the QUENCH-07 experiment. QUENCH-07 was a bundle test carried out in the QUENCH facility, using a bundle with 20 heated fuel rods and a single boron carbide control rod. A MELCOR input deck was assembled based on the QUENCH-06 input deck used for ISP45. [43] In QUENCH-07, the gaseous reaction products were measured directly, so the amount of CO, CO2, and CH4 produced was used as the basis of the comparison.

In general, the overall calculated temperature histories and failure times for the various components matched the experiment with one notable exception: the amount of hydrogen produced after the beginning of the cooldown phase. In the experiment, 182 g of hydrogen (corrected) were produced, 120 g of which were in the cooldown phase. Although the 62 g of hydrogen produced in the pre-cooldown period is matched by MELCOR, the large amount produced during cooldown is unaccounted for. It should be mentioned that all code calculations performed so far on this experiment have failed to account for the hydrogen, so this is not just a MELCOR problem. It has been suggested that some change in the morphology of the protective cladding oxide layer occurred during the quench period, resulting in the exposure of new Zircaloy surface during cooldown. If valid, this phenomenon is not currently modeled by MELCOR, nor by other severe accident codes.

2.5.13 Steam/Oxygen Allocation

As mentioned earlier, steam (and oxygen) from the core region CVH volumes is supplied to the COR Package component surfaces for oxidation purposes in a manner that takes into account the effects of both steam (and oxygen) starvation and flow blockage. To account for the effect of flow blockage within each core CVH volume, the minimum unblocked flow area for each of the rings interfaced to the volume is evaluated and then summed across all constituent rings. The CVH volume steam allocated to each constituent ring is the fraction of total unblocked flow area of the CVH volume represented by the ring. If the user desires, the calculation of unblocked flow areas may be bypassed (input record COR_RAF). The mass of steam within each ring is decremented as oxidation consumes the steam and no sharing of steam among the constituent rings is permitted during a COR subcycle. Thus, the components of some rings may completely consume the ring inventory of steam while other rings may remain steam rich.

To account for the effect of steam starvation on a ring-by-ring basis, the processing of oxidation effects is conducted for each radial ring of the CVH volume in the direction of flow. The direction of flow is determined from CVH results or from the evaluation of a user-prescribed control function (see input record COR_RP). Therefore, if the flow direction is upward, the progression of oxidation processing in the axial direction is from

bottom to top. For the up-flow condition, the entire ring inventory of steam is initially allocated to the surfaces of the lowermost axial cell in the ring adjacent to the CVH volume, the inventory is adjusted to account for oxidation, and the remaining steam is supplied to the components in the overlying cell in the ring. This axial marching is repeated until the uppermost axial segment of the ring within the CVH volume has been processed. All rings associated with the CVH volume are processed in this manner for each COR subcycle.

A second level of oxidant partitioning is performed at the cell level (axial segment-IA, ring-IR) within the CVH volume during the axial marching process. The object is to make results independent of the order in which the various oxidation reactions are evaluated. A fraction of the total available oxidant (steam or oxygen) available in this level of this ring is allocated to each possible oxidation reaction on each surface in proportion to the area available for that reaction. The reactions may include oxidation of zirconium, steel, aluminum, graphite, and/or B_4C (Sections 2.5.1, through 2.5.11). The portion of each intact component surface that is not blocked by candled materials (conglomerate debris) and the surface of the conglomerate debris on that component are each considered separately.

Because oxidation is calculated using rate equations subject to availability of steam, it is possible that all of the oxidant allocated to some surfaces may be consumed while only some of the oxidant allocated to other surfaces is consumed. In this situation, the oxidant that was not consumed is reallocated (using the same algorithm) among the starved surfaces, and the oxidation calculations for these surfaces are repeated. This process is repeated (a maximum of 10 times) until either

1. the ring oxidant inventory is exhausted, or
2. for each surface, either the metal content of each surface is consumed or the limit established by rate considerations is reached.

If the ring oxidant inventory is not exhausted, the calculation proceeds to the next cell in the direction of flow.

Because COR package calculations may result in total blockage (and thus steam/oxygen deprivation) of rings, the effect upon oxidation results and upon accident progression may be significant. Due to this dependence, sensitivity coefficient C1007 has been defined to provide a lower limit on the unblocked area fractions to be used in the partitioning of CVH volume oxidant inventories among the associated rings.

If the calculated unblocked area fraction for a ring falls below the corresponding limit specified for that ring by sensitivity coefficient C1007, then the fraction of CVH volume oxidant inventory allocated to the ring is held at the limit, and the remainder of the oxidant is divided among the remaining unblocked rings. If all rings are blocked, then the oxidant is divided among the rings according to the limits prescribed by sensitivity coefficient C1007, and any remaining oxidant is unavailable for oxidation. A check is made during

input processing to ensure that the sum of the ring fractions prescribed by sensitivity coefficient C1007 does not exceed unity.

2.6 Control Volume Temperature Distribution (dT/dz) Model

To accurately model the heat transfer to the gas from multiple COR cells interfaced to a single control volume, an estimate of the temperature distribution in the control volume atmosphere must be made in the COR package. Approximate local fluid temperatures are calculated for cells above the uppermost liquid level in the core; the remaining cells use control volume pool and atmosphere temperatures.

The dT/dz model used for this approximation assumes steady gas flow through the channel or bypass with known or specified inlet gas temperature and no cross-flow between core rings within any single CVH control volume. The model uses time-smoothed ("relaxed") CVH steam and/or oxygen outflow at the top of the core to determine whether the flow direction is upwards or downwards during each COR package subcycle. The flow relaxation time constant is adjustable through sensitivity coefficient C1030(2), which has a default value of 0.1 s. (The user can prohibit the consideration of downward flow, in imitation of earlier versions of MELCOR, by changing the default value of sensitivity coefficient C1030(1), but this degrades the calculation.) Because fluid temperatures are defined in the CVH package only as volume-averaged quantities and are not defined at particular flow path locations, various methods have been implemented to obtain a suitable inlet temperature for a control volume.

The default treatment is to take the inlet temperature as the temperature of the atmosphere flow actually entering the control volume, as calculated by CVH. If the CVH nodalization permits more than one such flow, a heat-capacity-weighted average temperature of the actual inflows is used. If water is boiling in the CVH control volume, the steam generation is treated as an "inflow" at the saturation temperature.

The default treatment includes the effects of cross flows between control volumes representing different radial portions of the core when a detailed CVH nodalization is used. It also minimizes the discrepancies between the calculated dT/dz temperatures and the CVH temperatures. (Note that donor differencing is used in the hydrodynamic equations, so that fluid is advected out of a control volume with enthalpy corresponding to the CVH temperature. For a core volume, this temperature should therefore correspond to the exit temperature for the portion of the core contained in that volume.) Because CVH and COR equations are not solved simultaneously, imperfections in the coupling may result in apparent discontinuities in the profile of dT/dz temperatures between core cells in different CVH volumes. We have found the consequences to be relatively minor, particularly in comparison to the consequences of major discrepancies between dT/dz and CVH temperatures, which cause termination of an execution if a temperature becomes nonphysical.

MELCOR 1.8.3 and earlier versions required the user to specify the definition of inlet temperature. This model has been extended slightly to allow consideration of downflow and is still available (input of IDTDZ = 1 on input record COR_MS is required), but its use is now strongly discouraged. (Consideration of downflow may also be disabled, allowing the 1.8.3 model, using sensitivity coefficient array C1030(1).) In this older model, the inlet temperature to the control volume atmosphere is taken as the saturation temperature if a pool is present and flow is upwards. Otherwise, there are several options available to the user (via the COR_TIN input record) to control how the inlet temperature to a control volume is determined:

(1). As a first option, the user may specify that the inlet temperature for any control volume be taken as the exit temperature from the control volume directly upstream of it, in the direction of assumed axial flow, as calculated with the dT/dz model described below. This option is the default except for the bottommost and topmost control volumes in the reactor vessel that contain core cells, for which it is not applicable.

(2). Alternatively, the user may specify that the inlet temperature for a control volume be taken as the CVH atmosphere temperature of some other control volume (or itself), as defined by the user. This option could be used for the lower head volume, for example, where the downcomer atmosphere temperature might be appropriate.

(3). As a third alternative, the user may specify that the value of a control function be used as the inlet temperature for a control volume. This option allows the user great flexibility in defining the inlet temperature and may be appropriate for complex flows or geometries, such as flows from more than one control volume entering the channel or bypass.

(4). The model may also be disabled, or the current default treatment selected, for specified volumes.

Once the inlet temperature for a control volume is determined, the temperature at each successive COR cell axial location, moving through the core or lower plenum in the direction of flow, is obtained by performing a simple energy and mass balance. The basic energy balance relates the change in energy in a cell, ΔE_{stored}, during a timestep to the enthalpy flow through the cell, H_{flow}, and any energy sources, q:

$$\Delta E_{stored} + H_{flow}\,\Delta t = q\,\Delta t \qquad (2\text{-}213)$$

The terms in Equation (2-213) are expressed in terms of masses, mass flow rates, and temperatures at the entrance and exit to the cell (note the canceling quantities):

$$\Delta E_{stored} = m^n h^n - m^o h^o = m^o c_p (T^n - T^o) + (\dot{m}_{in} - \dot{m}_{out}) h^n \Delta t$$

$$H_{flow} = \dot{m}_{out}\, h^n - \dot{m}_{in}\, h_{in}^n = \dot{m}_{in}\ c_p\ (T^n - T_{in}^n) - (\dot{m}_{in} - \dot{m}_{out})\, h^n \qquad (2\text{-}214)$$

$$q = (h^* A)_e\ (T_{s,e} - T_{out}^n) + q_{sou} \qquad (2\text{-}215)$$

where

Δt	=	timestep [s],
m	=	fluid mass in cell [kg],
$\dot{m}$	=	mass flow rate [kg/s],
C_p	=	gas specific heat [J/(kg K)],
h	=	enthalpy,
T	=	cell temperature [K],
$(h^*A)_e$	=	effective average heat transfer coefficient times surface area for the various cell components in contact with the current CVH control volume [W/K],
$T_{s,e}$	=	effective surface temperature [K] for cell components, and
q_{sou}	=	source heat rate [W], from fission product decay heat and B_4C reaction energy deposited in the atmosphere and from heat transfer from heat structures,

and superscripts "*n*" and "*o*" represent new and old time values, respectively.

In the interest of stability, mass flows calculated by CVH are relaxed (smoothed) before use in Equations (2-214) – (2-215).

$$\dot{m}_{dT/dz}^n = \dot{m}_{CVH} + min\left[exp(-dt/\tau_{RLXZ}), f_{max,old}\right]\left(\dot{m}_{dT/dz}^o - \dot{m}_{CVH}\right) \qquad (2\text{-}216)$$

where τ_{RLXZ} is a relaxation time, coded as sensitivity coefficient C1030(2) with a default value of 0.1 s, and $f_{max,old}$ is the maximum permitted weight for the old dT/dz flow, coded as sensitivity coefficient C1030(5) with a default value of 0.6. The use of $f_{max,old}$ is new in MELCOR 1.8.6 and is intended to deal with difficulties encountered when the time scale for flow changes is much less than the relaxation time defined by τ_{RLXZ}.

The dT/dz model in MELCOR 1.8.6 has also been modified to improve coupling between calculations in COR and those in CVH under conditions of little or no flow. The modification involved the assumption that there is a characteristic time for recirculation of fluid *within* each CVH volume, independent of flows *through* the volume, given by

sensitivity coefficient C1030(4) with a default value of 10 s. The effect is to add a fraction dt/C1030(4) of the mass in the atmosphere to $\dot{m}_{dT/dz}$.

The model solves for the value of T^n, which is then used as T_{in}^n for the next higher cell. Control volume average values for mass and mass flow rates are currently used at the inlet to the control volume and are updated for the effects of oxidation for each cell. For multiple core rings within the same control volume, the inlet mass flow rate is multiplied by the fraction of the total flow area for each ring, thus partitioning the flow across all rings.

For the dT/dz model to function correctly and model the phenomena appropriately, it is important that the heat structures representing the radial core boundary (e.g., core shroud) communicate with the fluid temperatures calculated by this model. The outer ring core cells must be specified as the fluid temperature boundary on input record HS_LBF/HS_RBF (see the HS Package Users' Guide) unless the IHSDT option switch provided on input record COR_MS has been set to 1.

The heat transfer rates obtained by using the dT/dz temperatures in conjunction with the core component surface areas and temperatures in all of the core cells associated with each CVH control volume within the core are summed and compared to the value that would be obtained if the CVH vapor temperature in that volume had been used instead of the dT/dz temperatures. If the heat transfer rates thus obtained are of opposite sign, then it is assumed that the dT/dz model is malfunctioning (probably because prevailing conditions are outside the scope of its intended application) and the dT/dz temperatures are overwritten by relaxing their beginning-of-step values with the value of the CVH vapor temperature in the corresponding CVH volume. Hence, if the model is malfunctioning, then relaxed CVH vapor temperatures are used instead, and the relaxation time constant for the CVH temperatures is adjustable through sensitivity coefficient C1030(3). Also, if the dT/dz model is deactivated by user input, then relaxed CVH temperatures are always used in place of results from the deactivated model.

2.7 Power Generation

2.7.1 Fission Power Generation

For ATWS accident sequences (or for fission-powered experiments), fission power is generated in addition to the decay heat. The COR package contains a simple model that calculates the fission power as a function of downcomer liquid level using the Chexal-Layman correlation [44]:

$$q_f = 0.037\ (C_u H_r)^{0.7}\ (P/P_r)^{0.3}\ (H/H_r)^{0.7} \qquad \text{(2-217)}$$

In this model, *H* is defined in terms of the downcomer liquid level L, relative to the top of active fuel and the distance L_f below the top of active fuel, where fission power drops to zero:

$$H = max\left[0.0, \left(L + L_f\right)\right] \quad (2\text{-}218)$$

$$L_f = 2.4384\ (P/P_r)^{0.45} \quad (2\text{-}219)$$

and

q_f = fraction of full operating power, which is defined by the Decay Heat package on input record DCH_FPW,

C_u = dimensional constant = 3.28084 m^{-1},

H_r = arbitrary reference height, selected as 1 m,

P = system pressure [MPa],

P_r = reference pressure, with default value 7.65318 MPa, and

L = height of downcomer water relative to the top of active fuel [m].

The Chexal-Layman correlation is based largely on work presented in Reference 17, in which steady state power levels were calculated using coupled, 3-D neutronic and thermal-hydraulic models of the reactor power and fluid flow. The correlation assumes that the core inlet enthalpy is always at saturation. The constants in this correlation are implemented in sensitivity coefficient array C1301.

The downcomer liquid level must be calculated by a control function specified on record COR_TP. Alternatively, this control function may directly calculate the fission power without use of the Chexal-Layman correlation, as discussed in the input description for record COR_TP.

The energy generated in the fission power model (as well as the decay heat if the RN Package is inactive) is distributed over the core cells using the radial and axial relative power densities input on records COR_RP and COR_ZP. The user has the option (as described in the input description for record COR_TP) for the fission energy to be deposited in the intact fuel components of all core cells (not lower plenum cells) or only in the intact fuel component of cells that are fully or partially liquid covered. In the latter case, the radial and axial relative power densities for these cells are renormalized to achieve this distribution.

Further, because this energy is not all deposited at the point of the fission (some of it is carried by energetic particles and radiation, e.g., gamma rays), the user has the option of specifying the distribution of the total fission power in a core cell over the components and materials within that cell, using sensitivity coefficients arrays C1311 and C1312.

(Direct transport of fission power to adjacent core cells is not modeled.) These coefficients specify relative absorbing efficiencies for the core materials and core components for a fraction, f_{esc}, of the fission power that is specified to "escape" the fuel. A single absorption efficiency is used for steel and steel oxide, and a single coefficient for Zircaloy, ZrO_2, and Inconel in grid spacers. The default values of these coefficients were modified in MELCOR 1.8.4 to model generation of fission power in components other than intact fuel to simulate gamma and neutron heating in nonfuel components. Thus,

$$P_{i,j,k} = P_T F_{cell} \left[(1 - f_{esc}) \frac{M_{i,j,k,UO_2}}{\sum_{k'} M_{i,j,k',UO_2}} + f_{esc} F_{i,j,k} \right] \tag{2-220}$$

is the fission power deposited in component *k* in cell *i,j* (radial ring *i*, axial level *j*), where

$$F_{cell} = \frac{f_i f_j \sum_k M_{i,j,k,UO_2}}{\sum_{i'} \sum_{j'} f_{i'} f_{j'} \sum_k M_{i',j',k,UO_2}} \tag{2-221}$$

is the fraction of the total fission power, P_T , born in cell *i,j*. Note that it is assumed that the fraction of that power born in component *k* is proportional to the UO_2 mass in component *k*, and that the term involving

$$F_{i,j,k} = \frac{\sum_m f_m M_{i,j,k,m}}{\sum_{k'} \sum_m f_m M_{i,j,k',m}} \tag{2-222}$$

represents the absorption by materials in that component of fission power not initially deposited in UO_2. For these equations,

- f_i = radial relative power density (input record COR_RP),
- f_j = axial relative power density (input record COR_ZP),
- $M_{i,j,k,m}$ = mass of material m in component k in cell i,j,
- f_m = relative material absorbing efficiency for escaping fission energy (sensitivity coefficient array C1311), and
- f_{esc} = fraction of fission energy escaping UO_2 (1 - C1312(1) from sensitivity coefficient array C1312).

The sum on *k′* in Equation (2-222) extends only over active components, as specified by the remainder of sensitivity coefficient array C1312. The sum over cells in Equation (2-221) extends only over the core region, that is, only over axial levels

j>number of axial levels in the lower plenum, and it is to be understood that $P_{i,j,k}$ is non-zero only for active components (as specified by sensitivity coefficient array C1312) in the core region. Therefore, no fission power is associated with components in the lower plenum. Note that the sum of $F_{i,j,k,m}$ over all materials and components is unity, as is the sum over components of UO_2 mass fractions, so that the sum of $P_{i,j,k}$ over all components is simply $P_T F_{cell}$,, the total fission power generated in that cell.

2.7.2 Decay Power Distribution

A model for distribution of decay power was added to MELCOR 1.8.4 to account for the distribution of gamma ray energy from fission product decay to components other than intact fuel. This model resembles the fission power distribution model described in the preceding subsection with two important exceptions: the calculation of average specific power (W/kg-UO_2) in the cell differs and decay power is distributed among components within cells throughout the entire lower plenum and core region. In addition, separate sensitivity coefficient arrays, analogous to C1311 and C1312, are used in the calculation. Implementation of the model, including determination of default values of the model parameters for BWRs and PWRs, is described in detail in Reference 16.

Decay heat generated in the core is produced by unreleased fission products, which are assumed to remain with the UO_2 material when it is relocated from intact fuel pellets to other components. As with the model for fission power, a fraction of the decay power is assumed to remain with the component containing the fission products, with the remainder absorbed by various materials in that and other components in the same cell. The net decay power deposited in component *k* in cell *i,j* is calculated as

$$DH_{i,j,k,net} = \left(1 - f'_{esc}\right) DH^0_{i,j,k} + f'_{esc}\ DH^0_{i,j}\ F'_{i,j,k} \tag{2-223}$$

where $DH^0_{i,j,k}$ is the decay heat born of fission products associated with component *k* in cell *i,j*

$$DH^0_{i,j} \equiv \sum_k DH^0_{i,j,k} \tag{2-224}$$

and

$$F'_{i,j,k,m} = \frac{\sum_m f'_m M_{i,j,k,m}}{\sum_{k'}\sum_m f'_m M_{i,j,k',m}} \tag{2-225}$$

represents the absorption by materials in component *k* of decay power escaping the UO_2 in which it was born.

Here,

$M_{i,j,k,m}$ = mass of material m in component k in cell ij as in Section 2.7.1, f'_m = relative material absorbing efficiency for escaping decay gammas (sensitivity coefficient array C1321) and

f'_{esc} = fraction of decay energy escaping UO_2 (1 – C1322(1) from sensitivity coefficient array C1322),

and the sum on k' extends only overactive components, as specified by the remainder of sensitivity coefficient array C1322.

When the RN package is active, the decay power $DH_{i,j,k}{}^0$ is calculated from the fission product inventories tracked for each component in each cell, using the specific power attributed to each radionuclide class as a function of time by the Decay Heat (DCH) package. As a result, the decay heat per unit mass of UO_2 is not the same for all components. In particular, the decay power in intact fuel pellets in various core cells reflects differences in initial fission product inventories corresponding to the power densities in those cells, while the decay power in particulate and conglomerate debris reflects the initial inventories in the fuel pellets that originally contained the UO_2. In addition, all decay power densities reflect differences in release resulting from the differing temperature histories of the UO_2 carrying the fission products.

If the RN package is not active, information on the distribution of fission products is not available. In this case, the total decay heat can only be approximately distributed over the UO_2 content of the active core components and debris in the cavity. The radial and axial power densities are considered for the UO_2 remaining in intact fuel pellets, but because of the absence of tracking information, the average specific power must be assigned to UO_2 in all other locations. This average specific decay power (W/kg-UO_2) is calculated from the whole core decay power provided by the DCH package as

$$DH(t) = \frac{DH_T(t)}{M_{UO2,cor}(0) + M_{UO2,cav}(0)} \qquad \text{(RN package not active)} \qquad (2\text{-}226)$$

where

DH_T = whole core decay power (Watts),

$M_{UO2,cor}$ = total UO_2 mass in the core (kg), and

$M_{UO2,cav}$ = total UO_2 mass in the cavity (kg).

The decay heat attributed to UO_2 in the various components in cell i,j is then calculated as

$$DH^0_{i,j,FU} = \mathrm{DH(\,t\,)}\ \frac{f_i\ f_j\ M_{i,j,FU,UO_2}}{\sum_{i'}\sum_{j'} f_{i'}\, f_{j'}\, M_{i',j',FU,UO_2}}\ \sum_{i'}\sum_{j'} \mathrm{M}_{i',j',FU,UO_2} \qquad (2\text{-}227)$$

$$DH^0_{i,j,k} = DH(\,t\,)\ M_{i,j,k,UO_2} \qquad (k \neq FU) \qquad (2\text{-}228)$$

where

f_i = radial relative power density (input record COR_RP) and

f_j = axial relative power density (input record COR_ZP),

as in Section 2.7.1. Note that

$$\sum_i \sum_j \sum_k DH^0_{i,j,k} = DH(\,t\,)\ \ M_{UO_2,cor}(\,t\,) \qquad (2\text{-}229)$$

so that the average decay power density in UO_2 in the core and lower plenum is simply the average power density *DH(t)* from the DCH package.

2.8 Material Interactions (Eutectics)

The material interactions model is invoked by entering integer 1 on input record COR_MS. When the model is active, the conglomerate debris materials associated with any component are treated as part of a coherent mixture. In the formulation of the model, some of the materials are treated as mutually miscible while all the others are considered mutually immiscible and treated as they are when the model is inactive (i.e. they melt and relocate independently of one another). As currently implemented, when the model is active all the materials are part of the miscible mixture. The material interactions model can only be activated during MELGEN execution and cannot be deactivated on a restart.

2.8.1 Mixture Formation

Molten material can enter the conglomerate debris mixture in one of three ways: (1) as a normal liquid formed when an intact solid reaches its melting point, (2) as a eutectic reaction product formed when two intact solids in mechanical contact within a core component reach their eutectic temperature, or (3) through the dissolution of an intact solid by an existing liquid mixture in the same core cell (e.g., the dissolution of UO_2 fuel by the liquid mixture associated with the cladding in the same core cell as the fuel). Currently, three eutectic reactions are considered that lead to early failure of fuel and control rods: (1) the eutectic reaction between Zircaloy cladding and Inconel grid spacers can lead to early failure of fuel rods, (2) the eutectic reaction between Zircaloy guide tubes and steel cladding can lead to early failure of PWR control rods and (3) the eutectic reaction between B_4C powder and steel cladding can lead to early failure of BWR control

rods. The threshold for the first two reactions is taken at 1400 K, and that for the B_4C-steel reaction at 1520 K, based on References [45] and [46], but these temperatures may be modified independently with sensitivity coefficients C1011. The molten material is placed in the conglomerate debris array associated with the component.

2.8.2 Mixture Properties

The properties of the mixture are mass-weighted averages of the constituent properties. The solidus and liquidus temperatures of the mixture depend upon the composition of the mixture and are currently calculated as a mole-weighted combination of the solidus temperatures determined by considering every binary combination of material pairs in the mixture. That is, the mixture solidus temperature is given by

$$TS_{mix} = \frac{\sum_{j}\sum_{i\neq j} f_i \quad f_j \quad TS_{ij}}{\sum_{j}\sum_{i\neq j} f_i \quad f_j} \tag{2-230}$$

where the f's are mole fractions and TS_{ij} is the solidus temperature for a mixture of materials *i* and *j* with the same relative proportions as in the total mixture. TS_{ij} can be obtained from pseudo-binary phase diagrams or simple mole weighting of the individual solidus temperatures. Presently, TS_{ij} is given by the mole-weighted average of the two solidus temperatures for all material pairs except for those listed in Table 2-12. For the pairs listed in the table, the solidus temperature is given by the mole-weighted average of the eutectic temperature and solidus temperature of the component present in excess of the eutectic molar composition. (The molar ratios and eutectic temperatures in Table 2-12 are currently hardwired and not implemented as sensitivity coefficients.) Equation (2-230) correctly reduces to TS_{ij} when only materials *i* and *j* are present in the mixture.

Table 2-12 Core eutectic reactions [45, 46]

Material Pairs		Molar Ratio	Eutectic Temperature
Zr	Inconel	0.76 / 0.24	1210
Zr	steel	0.76 / 0.24	1210
ZrO_2	UO_2	0.50 / 0.50	2800
Zr	B_4C	0.43 / 0.57	1900
Steel	B_4C	0.69 / 0.31	1420
Zr	Ag-In-Cd	0.67 / 0.33	1470

The liquidus temperature is set equal to the solidus temperature plus 0.01 K (an artificially small melting range is used to avoid the separation of a two-phase mixture into a solid

and liquid of vastly different temperatures, which may occur under the assumption of congruent melting that requires the solid and liquid to have the same composition).

The specific enthalpy is calculated in three temperature ranges as follows (refer to Figure 2.13) [47]:

(1). For temperatures less than the calculated solidus, the mass-weighted individual enthalpies are summed with the exception that extrapolated solid enthalpies are used for any material that would ordinarily be liquid.

(2). For temperatures greater than the calculated liquidus, the mass-weighted individual enthalpies are summed with the exception that extrapolated liquid enthalpies are used for any material that would ordinarily be solid.

(3). Otherwise, linear interpolation in enthalpy is used between the solidus and liquidus. The difference in enthalpy is the latent heat of fusion.

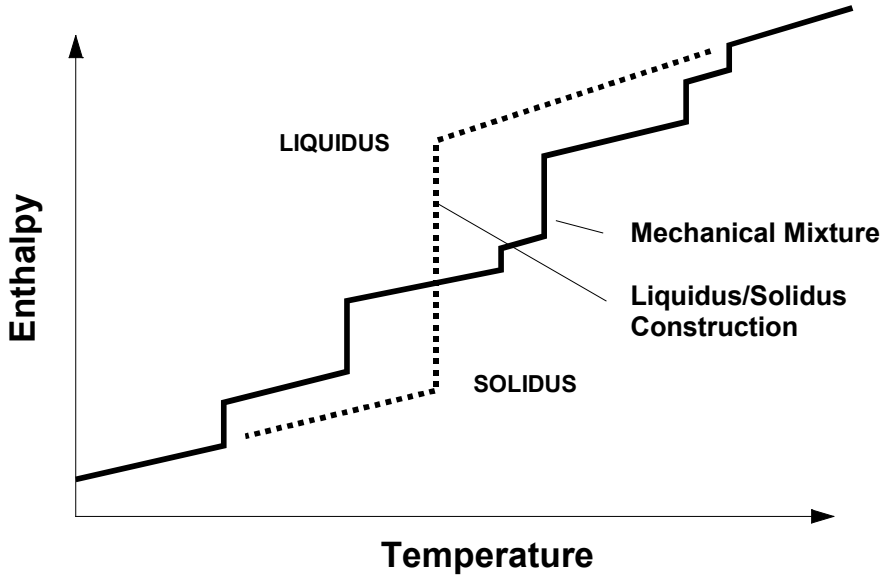


Figure 2.13 Two-phase construction for material mixture [47]

The Zircaloy and steel included in the mixture oxidize unless disabled by user input on record COR_TST. The oxidation reduces the metallic content of the mixture and increase the oxidic content.

2.8.3 Chemical Dissolution of Solids

If the enthalpy of the molten mixture exceeds its liquidus enthalpy, then the mixture begins to dissolve certain solids if they are present in the same core cell. The dissolution of solids proceeds sequentially, and at most two distinct solids may be attacked by the mixture

associated with a component on any given timestep. Table 2-13 lists the hierarchy used in determining which solids are dissolved by the mixtures associated with each core component (intact fuel does not have a mixture associated with it). Note that certain solids are attacked only if the oxide shell surrounding the component has been breached, while others are attacked only if the shell is intact. Holdup by oxide shells is described in detail in Section 3.1.3. The hierarchy listed is based upon the assumed arrangement of materials in intact core components. For example, it is assumed that a eutectic mixture that escapes from a PWR control rod must dissolve the ZrO_2 oxide shell that surrounds fuel rods before it can dissolve the UO_2 pellets within. Similarly, mixtures originating from BWR control blades encounter canisters. It should be noted that most intact components are eventually converted into particulate debris, so that even though the eutectic associated with BWR control blades is not assumed to reach intact fuel, after the blade becomes particulate debris the eutectic may have access to UO_2.

Table 2-13 Solid dissolution hierarchy

Component	Solids Dissolved by Mixture
Cladding	UO2 from intact fuel
	ZrO2 from intact cladding
Canister	ZrO2 from intact canister
	ZrO2 from intact cladding (A)
	UO2 from intact fuel
Other structure SS or NS (steel only)	steel oxide from the same other structure
Other structure NS (BWR control rod)	steel oxide from the same other structure
	ZrO2 from intact canister (A)
	Zr from intact canister (A)
Other structure NS (PWR control rod)	steel oxide from the same other structure (B)
	Zr from the same other structure
	ZrO2 from intact cladding (A)
	UO2 from intact fuel (A)
Particulate debris	UO2 from particulate debris
	ZrO2 from particulate debris
	ZrO2 from intact cladding
	UO2 from intact fuel

(A) indicates solid is attacked only if there is no holdup of the mixture in the component.

(B) indicates solid is attacked only if the mixture is being held up by the component

Dissolution proceeds until the addition of solid lowers the updated gross mixture enthalpy to the liquidus enthalpy associated with the updated mixture composition or until the parabolic rate limitation associated with the dissolution reaction has been exceeded for

the given timestep. The solution is iterative, and the parabolic rate limitations are given by [45]

$$(x_j^f)^2 = (x_j^i)^2 + K_j \ \Delta t \tag{2-231}$$

$$K_j = A_j \ exp\left(B_j / T\right) \tag{2-232}$$

where

x_j^f = final mass fraction of material j,

x_j^i = initial mass fraction of material j,

Δt = timestep (s), and

T = component temperature (K),

and the constants A_j and B_j may be adjusted through sensitivity coefficient array C1010. Default values for ZrO_2 and UO_2 are taken from Reference [18]:

$A_{ZrO2} = 1.47 \times 10^{14}$ $A_{UO2} = 1.02 \times 10^{15}$

$B_{ZrO2} = 8.01 \times 10^{4}$ $B_{UO2} = 8.14 \times 10^{4}$

These constants are based upon experiments using molten Zircaloy to dissolve UO_2 and ZrO_2, but the limits are applied to the dissolution of those solids by any mixture, irrespective of its composition. Consequently, as the fraction of Zircaloy in the mixture becomes small, the results from the model become suspect, and users are urged to conduct sensitivity studies to determine the effect of variations in the values of the constants in Equation (2-232). For the remaining materials, parabolic rate correlations have not been identified and no limitation is applied, although a limitation could be activated by supplying appropriate values for the sensitivity coefficients in Equation (2-232).

3. Core/In-Vessel Mass Relocation Models

This section describes the mass relocation models in the COR package. Candling of molten core materials, the transport of additional unmolten materials with the molten material, the radial relocation of molten pools, and the formation of flow blockages and molten pools are described in Section 3.1. The models for the radial relocation of molten pools and particulate debris are described in Section 3.1.5. Formation of particulate debris by various means from intact component, radial spreading of this debris, and its axial relocation by gravitational settling and collapse of supporting components are described in Section 3.2. The model that limits volumes available to accept the relocation of particulate debris (new in MELCOR 1.8.5) is described in Section 3.2.4.

3.1 Candling

The term *candling* is used here to refer to the downward flow of molten core materials and the subsequent refreezing of these materials as they transfer latent heat to cooler structures below. The COR package candling model is semimechanistic, based on fundamental thermal/hydraulic principles, but with incorporation of user-specified refreezing heat transfer coefficients defined for each material on record COR_CHT. The model is adaptable to steady flow of either films or rivulets (with smaller contact area than a film) by appropriate adjustment of these refreezing coefficients.

The model does not solve a momentum equation for a flow velocity. Instead, it assumes steady generation and flow of molten material, with all material generated within a timestep reaching its final destination within that step. For a steady melt generation rate, the amount of material entering into the candling model is proportional to the timestep, and, for small timesteps, the amount of material that refreezes at a particular location is also approximately proportional to the timestep. In other words, if for a given timestep, a certain amount of molten material is calculated with varying amounts refreezing at different axial locations, the assumption is that for a timestep twice as large, twice as much molten material would be generated and approximately twice as much would refreeze at each location. Thus, the cumulative behavior of the model should be relatively independent of timestep history. For situations involving release of a larger amount of molten material built up over several timesteps, alternative assumptions are used regarding the flow of that material and its contact time with structural surfaces to avoid timestep dependencies, as described in Section 3.1.

3.1.1 Steady Flow

Following the heat transfer and oxidation calculations, molten material may exist on the surfaces of components in various locations in core. It is assumed that this molten mass has been generated at a constant rate over the timestep, Δt. The candling model follows it as it flows down (because of gravity) through a column of cells. (A model to hold up molten material by an oxide shell until it is breached is described below.)

The amount of mass that refreezes on each lower cell component is determined by integrating the heat transfer rate between the molten film and the component:

$$q = h_m \; P_w \; \Delta z \; (T_m - T_s) \qquad (3\text{-}1)$$

over the timestep Δt, where

h_m = user-specified refreezing heat transfer coefficient [W/m^2-K],

Δz = cell height [m],

P_w = film or rivulet width (area of contact divided by Δz) [m],

T_m = temperature of the molten film [K], and

T_s = temperature of the component [K].

As energy is transferred between the melt and the component, their temperatures change. To account for this, implicitly projected new temperatures are used in Equation (3-1)

$$T_s = T_s^o + \frac{q\Delta t}{C_{ps}} \tag{3-2}$$

$$T_m = max\left(T_m^o - \frac{q\Delta t}{M_m c_{p,m}}, T_{mp}\right) \tag{3-3}$$

where

T_m^o = temperature of the component before candling,

C_{ps} = total heat capacity of the component [J/K],

M_m = molten mass that enters the cell on surface s [kg]

$c_{p,m}$ = molten film specific heat capacity [J/(kg K)],

T_m^o = temperature of molten film entering the cell [K], and

T_{mp} = melting point of film material [K],

and Equation (3-3) reflects the fact that although the molten film may carry a superheat, it is not cooled below its melting point.

$$Q_{sh} = M_m\ c_{p,m}\ (T_m - T_{mp}) \tag{3-4}$$

Equations (3-1) to (3-3) may be solved in the form

$$Q \equiv \mathrm{q}\,\Delta t = h_m\ P_w\ \Delta z \max(\Delta T_1, \Delta T_2)\Delta t \tag{3-5}$$

$$\Delta T_1 = \frac{C_{ps} M_m c_{p,m}}{C_{ps} M_m c_{p,m} + \left(C_{ps} + M_m c_{p,m}\right)\ h_m\ P_w\ \Delta z\ \ \Delta t}\left(T_m^o - T_s^o\right) \tag{3-6}$$

$$\Delta T_2 = \frac{C_{ps}}{C_{ps} + C_{ps}\ h_m\ P_w\ \Delta z\ \ \Delta t}\left(T_{mp} - T_s^o\right) \tag{3-7}$$

If Q is less than Q_{sh}, sensible heat is transferred but no mass is refrozen. If Q is greater than Q_{sh}, a mass

$$\Delta M_m = \frac{Q - Q_{sh}}{H_f} \tag{3-8}$$

is refrozen as conglomerate debris on the component surface, and then thermally equilibrated with the component.

If the underlying component is cladding, its effective heat capacity from Equation (2-93) is used for C_{ps}. This includes the effects of coupling to underlying fuel pellets. However, the candling calculation is performed after other heat transfer has been evaluated, so that the results are not included in the implicit fuel cladding gap heat transfer calculation described in Section 2.2.8. Therefore, only the appropriate fraction of the candling heat transfer to cladding from Equation (3-5) is transferred to the cladding, with the remainder going directly to the underlying fuel, as shown by

$$Q_{CL} = \frac{C_{CL}}{C_{CL,eff}} Q \tag{3-9}$$

$$Q_{FU} = Q - Q_{CL} \tag{3-10}$$

Molten mass is relocated downward in stepwise fashion according to Equation (3-8), until it has all refrozen on components in one or more lower cells). Figure 3.1 illustrates several steps in this process. The material refrozen on a component is termed *conglomerate* debris (as opposed to particulate debris) and becomes an integral part of that component.

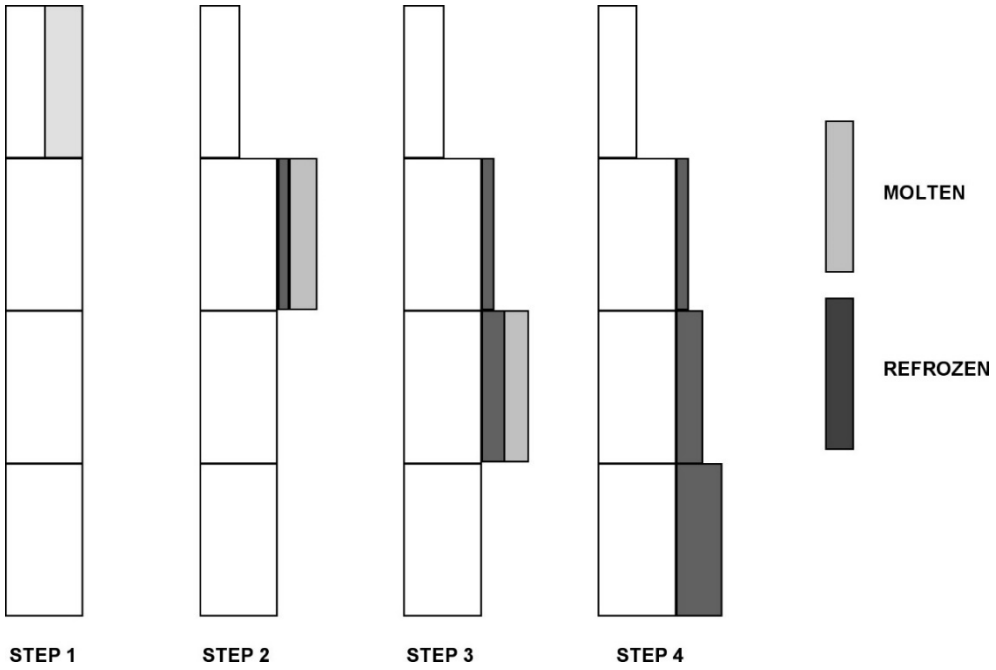


Figure 3.1 Candling process steps

If the material interactions (eutectics) model is not active, materials candle independently whenever their melting point is reached; otherwise, the molten portion of the conglomerate debris mixture candles as a congruently freezing mixture (i.e., when it freezes, the solid formed has the same composition as the liquid remaining).

Molten material originating in one type of component refreezes on the same component type in lower cells unless that component does not exist in those cells. If the originating component type does not exist in a cell, the molten material refreezes on an alternate component that depends on the originating component type and whether the cell is in the core or lower plenum. The definition of alternate refreezing components is summarized in Table 3-1. As indicated there, in the core the alternate refreezing component for material originating in all components, except particulate debris, is particulate debris, in either the channel or the bypass, as appropriate to the originating component. For the canister components, CN and CB, there is an opportunity for some part of the conglomerate to refreeze onto the clad component, which is described below. The alternate component for material originating in particulate debris in the channel is cladding, and for particulate debris in the bypass it is NS (presumably representing a control blade). In the lower plenum, a second alternate refreezing component is taken as SS (presumably representing CRGTs), if present. If neither the originating component nor an alternate refreezing component is found in a cell, the molten material falls through to the next lower cell.

Table 3-1 Alternate refreezing components

Cell Location	Originating Component Type				
	CL	**CN/CB**	**XS (A)**	**PD**	**PB**
Core	PD fallthrough	CL or PD fallthrough	PB/PD (B) fallthrough	CL fallthrough	NS fallthrough
Lower Plenum	PD SS fallthrough	PD SS fallthrough	PB/PD (B) SS fallthrough	CL SS fallthrough	NS SS fallthrough

(A) XS denotes any of SS or NS

(B) PB/PD denotes PB if there is a distinct bypass, otherwise PD

The volume occupied by molten and refrozen material during candling is tracked, and any related changes in component volumes are communicated to the CVH package as virtual volume changes. (The term virtual volume refers to space occupied by relocatable non-CVH materials in a control volume. Changes in virtual volume affect such things as liquid levels and pressure calculations. For a detailed discussion of virtual volume concepts, see the CVH Package Reference Manual.)

The two canister components, CN and CB, used in the BWR reactor model, can candle once a portion of their mass has become molten, and later refreeze onto lower, cooler

materials. The space into which the canister can candle can be thought of as the periphery of the fuel assembly and is characterized by a cross sectional area (PERA), declared on the COR_BFA record.

Once molten conglomerate debris is generated, it relocates to the cell immediately below and preferentially onto the same component. Thus, if the CN component becomes molten, it preferentially relocates to the CN component below and if the CB component becomes molten, the preferred host is CB. For each relocation, the molten conglomerate has the opportunity to refreeze a portion of its mass onto the underlying component.

The peripheral volume of the fuel assembly is divided into two, so that one part interfaces CB and the other CN. The fraction of the surface area of canister component CX (either CN or CB) to the total canister area scales the peripheral volume in a suitable proportion. There are two ways in which the peripheral volume of the fuel assembly (Δz.PERA), which is tracked for each COR cell, can be consumed: refreezing canister molten conglomerate onto a canister component or through the generation of excess conglomerate associated with the clad component that has fully occupied the clad's interstitial volume. The peripheral volume, V_P, is then calculated as:

$$V_{P,CX} = Max\left(\frac{A_{CX}}{A_{CB}+A_{CN}}\Delta z.PERA - V_{DC,ex} - V_{RF,CX}, 0\right) \quad (3\text{-}11)$$

where

CX = generic canister component representing either CN or CB,

A_{CB} = surface area of the canister component facing the blade,

A_{CN} = surface area of the canister component not facing the blade,

$V_{P,CX}$ = volume of the peripheral region between the canister and the rod matrix,

$V_{RF,CX}$ = volume of conglomerate debris from a canister component, CX, and

$V_{DC,ex}$ = volume of excess conglomerate that fills the rod's interstitial space.

If the volume of $V_{P,CB}$ reduces to zero and $V_{P,CN}$ is not zero, any clad conglomerate excess is allocated to $V_{P,CN}$; clad excess conglomerate always remains associated with the clad component. As volume $V_{P,CX}$ is reduced, the fluid volume also decreases simultaneously by the same amount. $V_{P,CX}$ does not serve to restrict fluid flow and is only used in the determination of a suitable host component for canister candling and in limiting the surface area of the canister conglomerate debris.

The conglomerate canister surface area is derived from the assumption of uniform relocation along a rectangular plate. The canister conglomerate surface area is calculated separately for CN and CB and for a generic canister component CX for each COR cell.

Secondary materials transport is accounted for, expressed as an explicit sum of mass - density quotients. The equation for conglomerate canister surface area is:

$$A_{CXDC} = 2\Delta z \sqrt{A_{CH,fl} + \frac{1}{\Delta z}\left(V_{rod} - \sum_{m}^{NMAT} \frac{M_{CX,m}}{\rho_m} - Min\left(\sum_{m}^{NMAT} \frac{M_{CXDC,m}}{\rho_m}, V_{P,CX}\right)\right)} \quad (3\text{-}12)$$

where

A_{CXDC} = surface area of conglomerate debris originating from component CX,

$A_{CH,fl}$ = fluid flow area in the channel,

V_{rod} = total volume of the fuel rods,

Δz = height of the COR cell,

m = index of the materials constituting component CX,

$M_{CX,m}$ = mass of material m of component CX,

$M_{CXDC,m}$ = mass of material m belonging to the conglomerate originating from CX, and

ρ_m = density of material m.

The minimum canister conglomerate area is equal to the perimeter of the peripheral volume cross section, multiplied by the height of the cell.

Canister candling differs from the candling of other intact COR components because there is a possibility that the material migrates onto the control rods. The direction of the candled material depends on the volume associated with the periphery, $V_{P,CX}$, and the degree of blockage, thus presenting two possible ways in which the canister conglomerate can migrate onto the fuel rods.

The first consideration is the event in which the volume of conglomerate canister is too large for the associated volume of the fuel assembly periphery, $V_{P,CX}$. This may occur with or without the underlying refrozen conglomerate, but the most likely occurrence is when some part of the peripheral volume is already occupied. This scenario is depicted in Figure 3.2. The event starts with the formation of conglomerate canister in the uppermost COR cell in (A). Since clad is usually hotter than canister, it is not depicted as adjacent to canister in the top node. Some previously refrozen canister conglomerate debris exists in a lower cell that occupies some of the peripheral fuel assembly volume such that a reduced $V_{P,CX}$ remains.

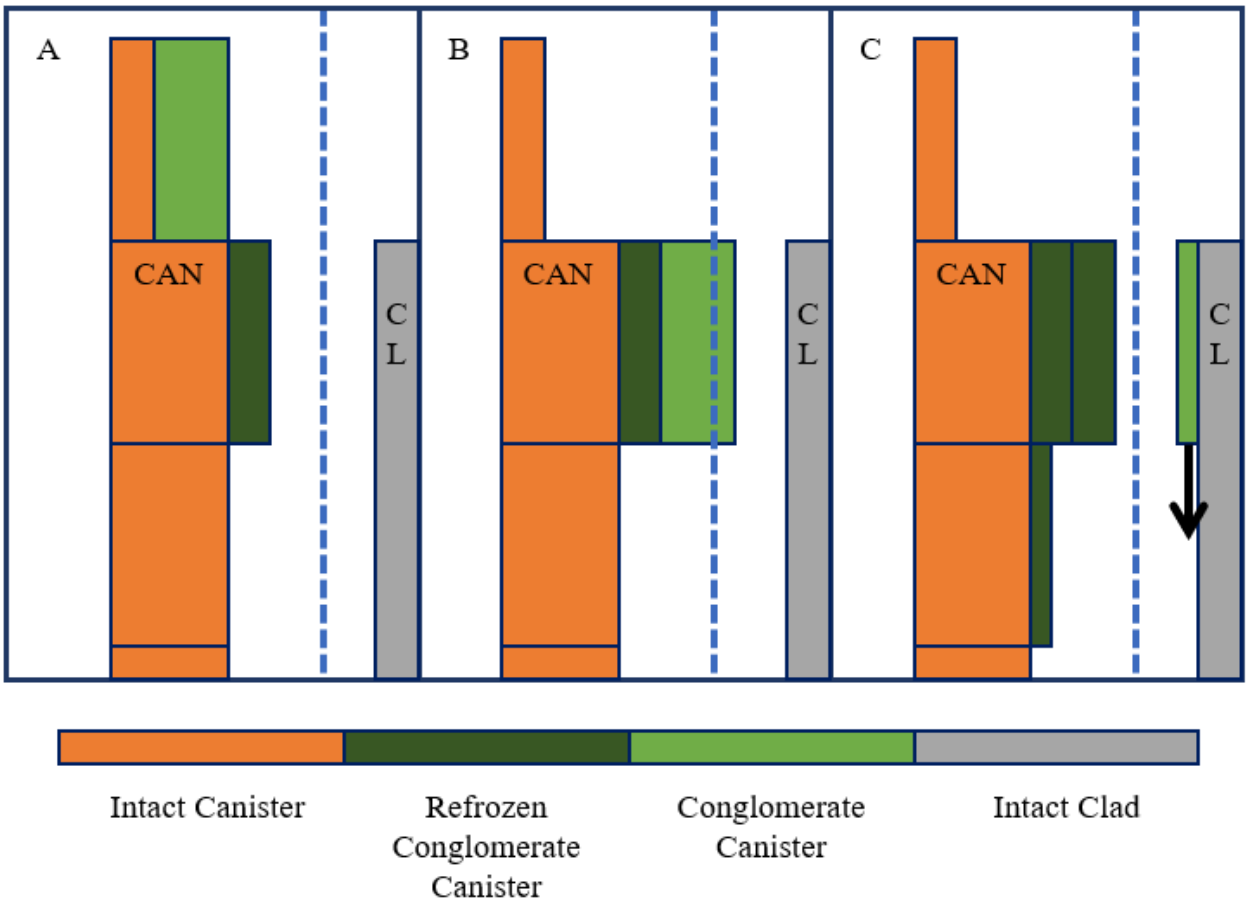


Figure 3.2 Canister conglomerate carryover to fuel assembly periphery

In (B) the canister conglomerate is relocated to the cell below, where there is insufficient volume in the remaining peripheral space V_P. The excess volume is then relocated onto the clad component where it becomes indistinguishable from clad conglomerate as indicated in (C). If the clad component does not exist, the newly identified clad conglomerate candles onto the denuded fuel component. The new clad conglomerate contributes fully to the rod interstitial volume and bridging model as described in Section 3.1.6 of the MELCOR Reference Manual for the COR package. If neither clad nor fuel exists, it candles onto particulate debris or falls through to the next lower cell. The portion of canister conglomerate that remains in the peripheral volume is then free to continue candling along the same canister component and refreeze there, as is indicated in (C).

The second consideration is when the canister candles to a lower volume that has become blocked at the sub-grid level, as depicted in Figure 3.3. A crust that forms in the peripheral volume cannot support a molten pool unless a crust, formed by the conglomerate of all components associated with the channel region, extends across the whole channel area. If a canister crust extends across the peripheral region but has not

yet formed a blockage in the interstitial area associated with the fuel rods, then some portion of the conglomerate mass in that cell is able to refreeze, but only up to a maximum volume that is permitted above the bridged grid level. All remaining conglomerate debris is then transferred to the fuel rod for further candling and refreezing as before.

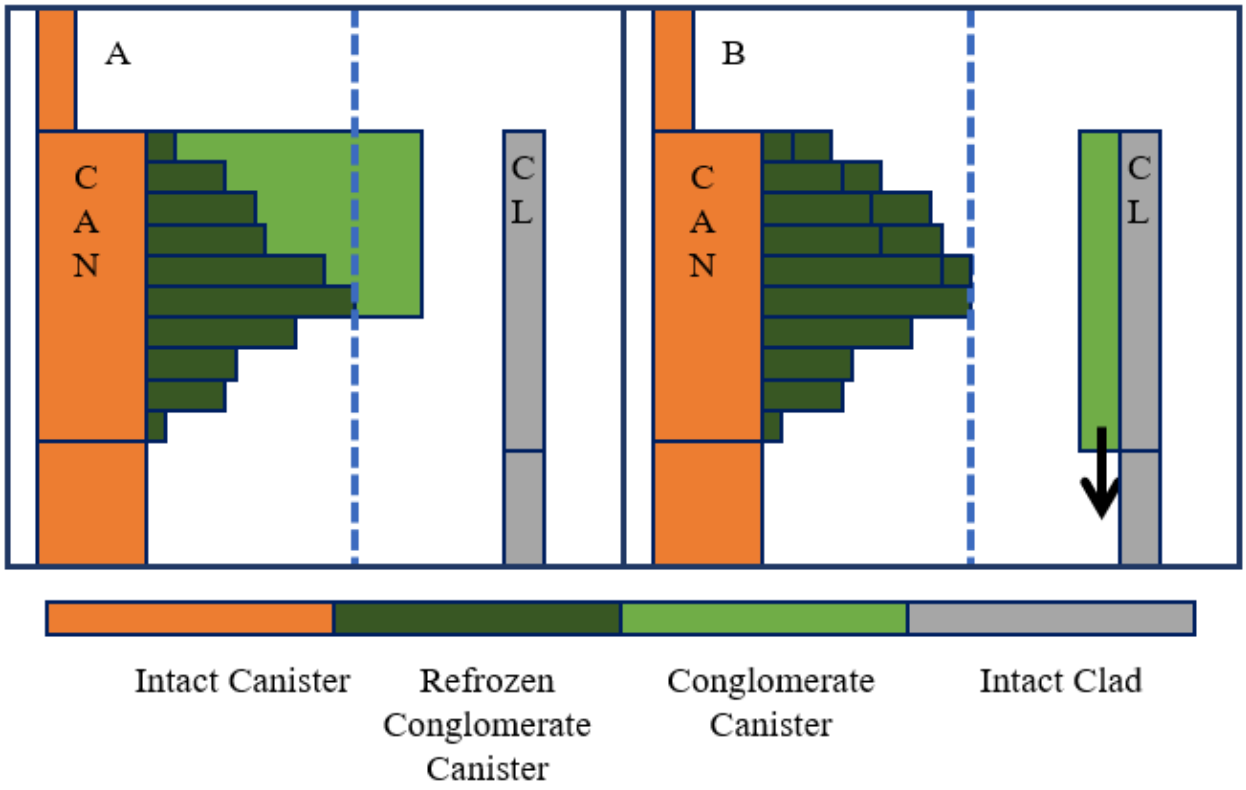


Figure 3.3 Canister conglomerate carryover due to blockage formation

3.1.2 Flow Blockages

Important changes were made to the candling model in version 1.8.6 to improve the ability to predict the formation of flow blockages. As shown in Figure 3.4, for MELCOR 1.8.5, flow blockages can only occur when refrozen material completely fills the available volume in a COR cell. This, of course, can lead to node size dependence because large COR cells would be more difficult to fill than smaller cells. In MELCOR 1.8.6, the vertical distribution of material refrozen on components within a core cell is tracked so that a local blockage can form. (The number of sub nodes in this distribution function is accessible to the user via a sensitivity coefficient.) By default, ten sub nodes are assumed for each COR cell. When a local blockage is detected, any remaining unfrozen candling material becomes molten pool material. Molten pool material is created from unfrozen candling material.

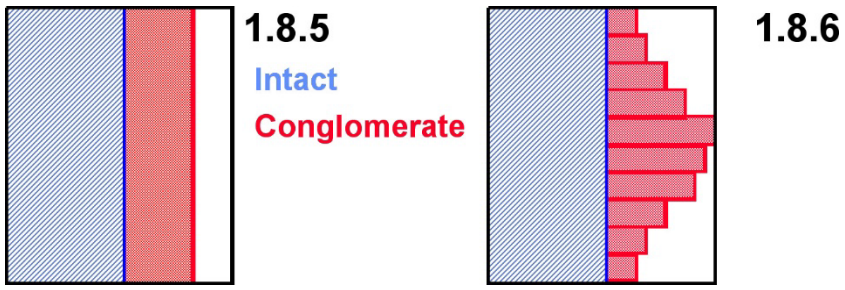


Figure 3.4 Flow blockage for a cell, as predicted by the COR candling model

As molten material candles it transfers heat to the underlying component. Eventually it loses its latent heat and refreezes onto the surface, or it is blocked and forms a molten pool. Generally, in MELCOR 1.8.6, refrozen material is distributed non-uniformly over the surface of the underlying component where before it was assumed to be uniformly distributed over an intact surface. This distribution modifies every time candled material refreezes on components and whenever refrozen material melts or relocates.

The formation of a local blockage in the upper core obstructs downward-relocating molten materials, even though there may otherwise be available volume for relocation. It also obstructs the downward relocation of solid materials. When solid particulate debris relocates into a suspended molten pool, it is retained above the blockage. The distinction between particulate debris above and below the blockage is maintained by adding the former to the “intact” (as distinct from “conglomerate”) portion of the molten pool component. As a consequence, the relocating particulate debris is thermally equilibrated with the existing molten pool.

Molten material is transferred between radial rings to achieve a uniform surface level across the pool as discussed in Section 3.3.2. Candling of molten pools accumulated above a blockage after failure of that blockage is discussed in Section 3.1.

Relocation of core materials may result in a reduction of area and an increase of flow resistance, or even a total blocking of flow, within various parts of the core. The effects on hydrodynamic flows may be modeled by using the core flow blockage model in the hydrodynamics package, which requires input of FL_BLK records for the associated flow paths. In addition to modeling the change in flow area, this model calculates the change in flow resistance. The resistance is based on a model for flow through porous media when particulate debris is present; otherwise, the input flow resistance for intact geometry is simply modified to account for any change in flow area. This model, described in the CVH/FL Reference Manual, uses a porosity based on the ratio of available hydrodynamic volume to total volume (see Section 3.2.4); a minimum porosity is imposed by sensitivity coefficient C1505(1), with a default value of 10^{-5}.

MELCOR 1.8.5 also includes a model for the opening of a flow path between the channel and bypass regions of the core upon failure of the canister in a BWR.

Activation of these models is *not* automatic. Input on FL_BLK records is required to specify which core cells are associated with each flow path involving the core. Furthermore, because only CVH and FL model the flow of water and gases, the effects of blockages on circulation can be modeled only to the extent that the CVH/FL nodalization can resolve that circulation. For more details, see discussion the COR Package Users' Guide and input instructions in the FL Users' Guide.

3.1.3 Holdup by Oxide Shells

A model has been implemented in the COR package for an oxide shell to hold up molten material until the shell is breached. Molten material is held up within a component if the oxide thickness is greater than a critical value Δr_{hold}, if the component temperature is less than a critical value T_{breach}, and if no candling from the component in that cell has yet taken place. The parameters Δr_{hold} and T_{breach} may be set independently for steel and for Zircaloy in cladding and in canisters via sensitivity coefficient array C1131. The default values for these sensitivity coefficients are currently set so that there is holdup by Zircaloy oxide but not by steel oxide.

When an oxide shell is first breached, or when a flow blockage or crust first fails, the assumption built into the candling model of constant generation of melt over the timestep is no longer valid. Behavior of the model related to the amounts of mass refrozen in lower core cells, as described in Section 3.1, would thus be highly dependent on the size of the current timestep. Therefore, for those situations involving the sudden release of a large mass of molten material, M_m, built up over perhaps several previous timesteps, application of the candling model is modified slightly. For breach of an oxide shell, a constant timestep Δt_{break} is used. For failure of a flow blockage holding up a molten pool, a timestep, $\Delta t_{contact}$. This timestep is calculated as a function of a parameter, Γ_{max}, that represents a maximum flow rate (per unit surface width) of the molten pool after breakthrough:

$$\Delta t_{contact} = max \left[\Delta t, \frac{M_m \ \Delta z}{\Gamma_{max} A_{surf}} \right] \tag{3-13}$$

In other words, a large molten pool is allowed to discharge at a maximum rate of Γ_{max}, and the amount refreezing onto structures below is a linear function of the total mass of the pool. Both Δt_{break} and Γ_{max} are accessible in sensitivity coefficient array C1141; their default values of 1 s and 1 kg/m–s have been set so that this model is only active for large molten pools breaching a crust.

3.1.4 Solid Material Transport

A simple model has been implemented to allow transport of unmolten secondary materials (currently ZrO_2, UO_2, steel oxide, and control poison) via the candling process.

This model could be used to treat the breaking off of pieces of thin oxide shells that are carried with the molten material or to simulate the dissolution of UO_2 by molten Zr. On input record COR_CMT, the user may specify relocation of a secondary material, ΔM_s, as either an input fraction F_1 of the molten mass ΔM_m deposited on a component:

$$\Delta M_s = F_1 \Delta M_m \tag{3-14}$$

or in fractional proportion to its existing fraction within a component:

$$\Delta M_s = F_2 \frac{M_{s,total}}{M_{m,total}} \Delta M_m \tag{3-15}$$

where F_2 is an input parameter specifying the fraction of direct proportional relocation, $M_{s,total}$ is the total secondary material mass in the component in the cell of origin, $M_{m,total}$ is the total material mass (molten and solid) in the cell of origin, and ΔM_m is the secondary material mass deposited with refrozen material ΔM_m.

This model is inactive if the COR materials interactions (eutectics) model, which is described in Section 2.8 and treats dissolution mechanistically, is active.

3.1.5 Radial Relocation of Molten Materials

There are two radial relocation models: the first relocates molten core material that still exists following the candling/refreezing algorithm just described. The second, which relocates particulate debris, is essentially similar. Both models are intended to simulate the gravitational leveling between adjacent core rings that tends to equalize the hydrostatic head in a fluid medium. Either of the two radial relocation models can be deactivated by user input on MELCOR input record COR_TST, but they are both active by default.

The molten material radial relocation model considers each axial level of the core independently, and is invoked after the axial relocation (candling) model. A simple algorithm loops over all adjacent pairs of radial rings between which relocation is possible and compares the calculated liquid levels in the two. If the levels are unequal, then a calculation is performed to determine the volume of molten material, V_{eq}, that must be moved between the rings to balance the levels. Furthermore, when the stratified molten pool model is active, molten material may reside in both the oxide molten pool component and the metallic molten pool component. Leveling is performed for each component and displacement of metallic molten pool material by assumed heavier oxide molten materials is considered. Furthermore, the nonuniform axial variation of the cell volume for core cells adjacent to the curved lower head is used in determining pool heights. It is assumed that the radial relocation is blocked by the presence of an intact BWR canister structure in either ring. In addition, radial relocation is not allowed within a core plate. The actual

implementation prevents such relocation to or from a core cell containing supporting structure modeled as a plate.

The relocation rate has a time constant of τ_{spr}, which may be adjusted by user input, so that the actual volume relocated, V_{rel}, during the core timestep, Δt_c, is given by

$$V_{rel} = V_{eq} \left[1 - \exp\left(-\Delta t_c / \tau_{spr}\right)\right] \quad (3\text{-}16)$$

The default value of 60 s for τ_{spr} was chosen as an order-of-magnitude value based on engineering judgment and recommendations of code users. It is accessible as sensitivity coefficient C1020(2).

If the volume of the material that must be relocated is trivial (specifically, less than 0.01 m^3/kg times the mass below which any component is eliminated, C1502(1), which has a default value of 1.0 x 10^{-6} kg), then no relocation is performed during that timestep; otherwise, the fraction of the molten material that must be transferred from the “deep“ ring to the “shallow“ ring is determined by dividing the mass of melt that must be relocated by the total mass of melt in the deep ring. That fraction of molten mass is then transferred from each core component in the deep ring to the conglomerate debris associated with the particulate debris component in the shallow ring, and the component volumes in each ring are adjusted accordingly. Any fission product transfers or virtual volume adjustments resulting from the relocation are performed by calls to interface routines with the RN package and CVH package, respectively.

Radial relocations are directed inward preferentially; that is, at each axial level the algorithm begins at the innermost ring, marches radially outward and transfers molten material from ring i to ring i-1 if the liquid level in ring i exceeds that in ring i-1. Following the march from ring 1 outward, a reverse march is made inward from the outermost ring to perform any outward relocations from ring i to ring i+1 still required to achieve a uniform liquid level across the axial level.

3.1.6 Surface Area Effects of Conglomerate Debris

The addition of conglomerate debris refrozen on component structures affects the surface area exposed to fluid convection, oxidation, and further refreezing during candling. For fuel rods and particulate debris, conglomerate debris can fill interstitial spaces, thus occluding some or all of the surface of the underlying component. The following paragraphs describe in detail a model specifically developed for fuel bundles. The general form of this model is incorporated into the COR package for <u>all</u> core components, but with different coefficients for each. With the default values of these coefficients, it is actually used only for fuel rods and particulate debris.

Consider the candling process idealized for a fuel rod unit cell, as shown in Figure 3.5. Molten debris refreezing on the rod is assumed to begin forming a half-cylinder on the rod at the point directly adjacent to the next rod [Figure 3.5 (a,b)]. As this half-cylinder of conglomerate continues to grow, its surface area expands, and the intact area shielded also grows, albeit at a lesser rate. Eventually it meets the conglomerate on the adjacent rod and forms a bridge between the two rods [Figure 3.5 (c)]. As additional material is added, more of the intact rod is covered by the conglomerate, until a cylindrical void region centered in the interstitial region among a set of four rods is created [Figure 3.5 (d)]. This central void then shrinks to nothing as the interstitial area is completely plugged up [Figure 3.5 (e)].

For purposes of calculation, the above-described process is divided into three stages. The first stage lasts until the conglomerate debris half-cylinders bridge the gap between rods, as shown in Figure 3.5 (b). The second stage lasts until that bridge has widened to cover the entire surface area of the fuel rods, forming a central cylindrical void, as shown in Figure 3.5 (c). The third stage continues until the central void is completely plugged up as shown in Figure 3.5 (e). The surface area of the conglomerate debris in the unit cell is calculated in approximate fashion from the fraction of the interstitial volume that it occupies.

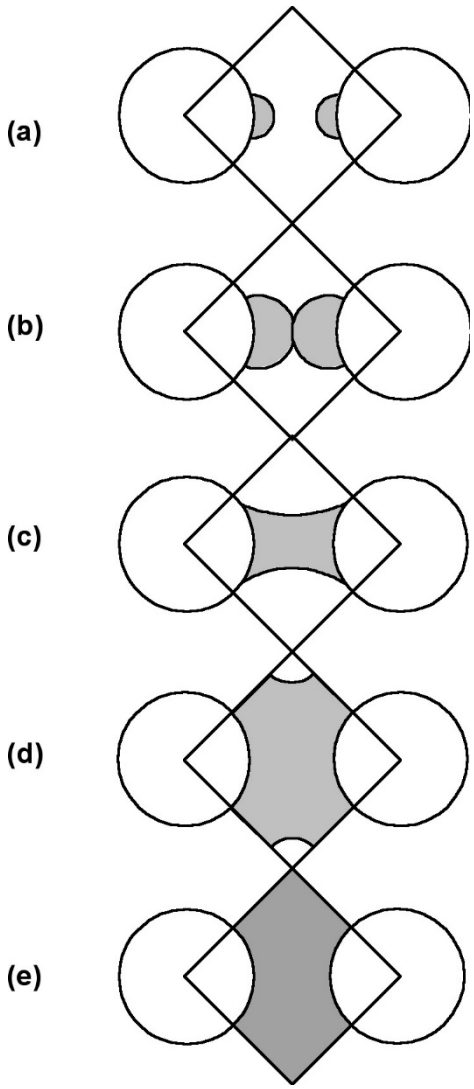


Figure 3.5 Conglomerate debris geometry in fuel rod bundles

It is convenient to define areas and volumes relative to the unit cell rod surface area A_i and initial interstitial volume V_i. The latter is related to the volume of the rods by

$$\frac{V_i}{V_{rod}} = \frac{P^2 - \pi R^2}{\pi R^2} = \frac{\varepsilon_{bundle}}{1 - \varepsilon_{bundle}} \tag{3-17}$$

where P is the rod pitch and R is the rod radius, as defined by the COR_GP input record, and ε_{bundle} is an effective porosity of the rod bundle.

During the first stage, the surface area of the conglomerate debris A_{cd} grows as the square root of its volume V_{cd} up to some critical volume V_{c1} with surface area A_{c1}.

With the definition of Equation (3-17), it may be shown that

$$A_{cd} = A_i \left(\frac{A_{c1}}{A_i} \right) \left(\frac{V_{cd} / V_i}{V_{c1} / V_i} \right)^{1/2} \tag{3-18}$$

$$\frac{A_{c1}}{A_i} = F_{A1,\max} = \frac{P - 2R}{R} \tag{3-19}$$

$$\frac{V_{c1}}{V_i} = F_{V1,\max} = \frac{\pi}{2} \frac{(P - 2R)^2}{(P^2 - \pi R^2)} \tag{3-20}$$

During the third stage, beyond some critical volume V_{c2} with surface area A_{c2}, the surface area of the conglomerate debris decreases as the square root of the empty volume ($V_i - V_{cd}$). In terms of area and volume fractions,

$$A_{cd} = A_i \left(\frac{A_{c2}}{A_i} \right) \left(\frac{1 - V_{cd} / V_i}{1 - V_{c2} / V_i} \right)^{1/2} \tag{3-21}$$

$$\frac{A_{c2}}{A_i} = F_{A2,max} = \frac{P / \sqrt{2} - R}{R} \tag{3-22}$$

$$\frac{V_{c2}}{V_i} = F_{V2,max} = 1 - \pi \frac{(P / \sqrt{2} - R)^2}{(P^2 - \pi R^2)} \tag{3-23}$$

A minimum area fraction $F_{A,min}$ may be imposed for the third stage to prevent the surface area of central void from being completely reduced to zero. In any case, the surface area of conglomerate debris is not reduced below a minimum surface-to-volume ratio as described below.

During the second stage, the surface area of the conglomerate debris is interpolated linearly with volume between A_{c1} and A_{c2}.

The area of the intact rods wetted by the conglomerate, and thus blocked from further oxidation and convection, is treated in two stages. For volumes greater than V_{c2}, the fraction of intact surface area A_i blocked is set to a maximum value:

$$F_b = F_{b,max} \tag{3-24}$$

For volumes less than V_{c2}, the fraction blocked is linearly interpolated:

$$F_b = F_{b,max} \frac{V_{cd}}{V_{c2}} \tag{3-25}$$

The same form is applied for all components. For particulate debris, the user-input porosity of the debris bed is used to replace ε_{bundle} in Equation (3-17); for all other components, the interstitial volume, V_i, is taken as zero. The parameters $F_{A1,max}$, $F_{V1,max}$, $F_{A2,max}$, $F_{V2,max}$, $F_{A,min}$, and $F_{b,max}$ are accessible for each component as sensitivity coefficient array C1151. Currently, all components have default values based on typical BWR rod geometries with pitch 16 mm and rod radius 6.26 mm. However, they are used only for fuel rods and particulate debris.

For conglomerate debris that does not occupy interstitial volume (either the component does not have interstitial volume via the porosity input or the debris overflows what is available), a simple surface area-to-volume ratio is applied to the excess conglomerate debris volume $V_{cd,excess}$:

$$A_{cd,excess} = V_{cd,excess} R_{SV} \tag{3-26}$$

The parameter R_{SV} is also accessible in sensitivity coefficient array C1151, with a default value of 100. The surface area of the excess debris is added to the area calculated from Equations (3-18) to (3-25) The total surface area of conglomerate debris (excess plus interstitial) cannot fall below the value obtained by multiplying the debris volume V_{cd} by R_{SV}.

Furthermore, to avoid overheating a vanishing CVH fluid, the sum of the surface areas of the intact component and its associated conglomerate debris, which constitutes the total effective surface area for heat transfer to CVH, cannot exceed

$$A_{tot,\max} = \max\left(V_{CVH} R_{SVf}, \varepsilon_{\min} V_{COR}\right) \tag{3-27}$$

where R_{SVf} is a limiting surface-to-volume ratio, accessible as sensitivity coefficient C1152(1) with a default value of 1000 m^{-1}, ε_{min} is a minimum porosity in the core, accessible as sensitivity coefficient C1505(2) with a default value of 0.05, and V_{CVH} and V_{COR} are the total volumes of fluids and COR materials, respectively.

3.2 Particulate Debris

After core components collapse, the materials that composed them are treated as particulate debris. After it has been formed, this debris can spread radially and/or settle vertically, subject to the availability of free volume and the presence or absence of support.

3.2.1 Formation of Particulate Debris

The COR package contains several simple models that consider the structural integrity and support of intact components and convert them to particulate debris when either is lost. Most are logical models rather than structural models; no stress calculations are performed for any component other than supporting structure (SS). Even for SS, such a calculation is optional. Complex debris formation mechanisms, such as quench-induced shattering, have not been implemented into the COR package at this time.

All components other than fuel rods (FU and CL) are immediately converted to particulate debris whenever the unoxidized metal thickness is reduced below a user-defined minimum value. The thickness criterion is also used for cladding (CL), which is assumed to support fuel pellets (FU), but other criteria are also considered for fuel rods. On record COR_CCT, the user may define one minimum thickness parameter, $\Delta r_{cl,min}$, with a default of 0.1 mm, that is used for Zircaloy in the cladding (CL) and the two canister components (CN and CB).

For the nonsupporting structure component (NS), the structural metal may be taken either as steel (the default) or as Zircaloy. The default minimum thickness is also 0.1 mm. Both the structural metal to which it is applied, and the minimum thickness may be specified independently for each core cell containing NS.

Setting any Δr_{min} to zero prevents collapse of the associated components by this mechanism, although MELCOR may still predict their collapse using one of the other criteria described below. If the user has specified electric heating element material in the fuel rods, formation of particulate debris is suppressed, and the minimum thickness parameter $\Delta r_{cl,min}$ must be set to zero.

Unoxidized metal thickness is reduced both by oxidation and by melting and candling of metal. It is expected to increase, except for the case of cladding, by refreezing of metal candled from above. If candling of molten material is not possible because of a flow blockage or holdup by an oxide shell (Sections 3.1.2 and 3.1.3), the retained metal is considered as part of the unoxidized thickness. In effect, the component is considered to be supported by the oxide shell that contains the held-up melt or by the surrounding pool of molten material. Particulate debris is formed from CN or CB whenever the temperature of the component reaches the melting temperature of the associated oxide (ZrO_2 for CN and CB). The temperature at which NS is converted to particulate debris, independent of metal thickness, may be independently specified for each core cell containing NS, with a default value of the melting point of the structural metal identified for NS in that cell.

Fuel rods, composed of cladding and fuel pellets (the CL and FU components), are treated somewhat differently. Oxidized rods are assumed to retain their identity until the cladding reaches 2500 K and to collapse unconditionally if the fuel temperature reaches 3100 K (the approximate melting temperature of UO_2). In MELCOR 1.8.4, the former

temperature was taken as 2800 K, the approximate melting temperature of the UO_2/ZrO_2 eutectic, but experience with Phebus has shown that the lower temperature is more appropriate for irradiated fuels. Both temperatures are accessible to the user through sensitivity coefficient array C1132. It is possible for a fuel rod to be hot but unoxidized, either as a result of heating in an inert environment or following total loss of ZrO_2 through candling involving secondary transport (Section 3.1.4) or eutectics (Section 2.8). As currently coded, such a rod is converted to particulate debris when the remaining metal thickness falls below $\Delta r_{cl,min}$.

Finally, an intact component is converted to particulate debris whenever that component's support is lost. This support may be provided by either the same component or the unfailed supporting structure (SS) component in the cell below; the portion of a fuel rod in level n supports the portion in level n+1, and the core support plate is considered to support all components above it.

When a component of the core of a BWR collapses to form particulate debris within the core region, this debris can occupy space either inside or outside the channel boxes. In earlier versions of MELCOR, only a single particulate field was available, and all components collapsed to form particulate in the channel. In MELCOR 1.8.5, particulate debris in the bypass (PB) is distinguished from that in the channel (PD). In any core cell with a distinct bypass, the structural components SS and NS are modeled as collapsing to form PB, while all others collapse to form PD. As this debris is later relocated, it may—depending on geometry—occupy the channel or bypass region of other cells or be split between them.

Particulate debris is characterized by user-specified particle diameters, D_{pd}, and D_{pb}, entered as hydraulic diameters on input record COR_EDR. The two diameters are equal by default, but this is not required. However, there is no provision at this point for considering more than a single representative diameter for either. The surface area of the particulate portion of each type of debris is calculated from D_{px}, and the total volume of the particulate, V_{px}, as

$$A_{s,pd} = \frac{6\ V_{px}}{D_{px}} \tag{3-28}$$

where *x* can be *d* or *b*. The fraction of this area used for oxidizing the Zircaloy portion of the particulate debris is the fraction of the particulate debris volume that is Zircaloy plus ZrO_2. The fraction of this area used for oxidizing the steel portion of the particulate is the fraction of the particulate volume that is steel plus steel oxide. ZrO_2 and steel oxide in particulate debris are modeled to exist as layers covering the Zr and steel, respectively. The particulate areas of the debris are further modified by the addition of conglomerate debris, according to the model described in Section 3.1.6 to obtain actual areas for oxidation and heat transfer.

Note that when intact components are converted to particulate debris, the distribution of refrozen conglomerate associated with the intact component is redistributed uniformly throughout the newly created particulate debris.

3.2.2 Time-at-Temperature Fuel Rod Failure

A simple time-at-temperature (TaT) model has been created to model the complex physio-chemical processes resulting in an eventual failure of the fuel. Fuel failure is characterized by the loss of normal fuel rod geometry whereby the code simulates the instant transition from intact rod geometry to particulate debris. The TaT criterion was introduced in an attempt to avoid non-physical cliff-edge effects that are observed during a calculation when fuel temperatures are predicted to remain slightly below a failure temperature criterion for extended periods. The TaT model determines fuel failure by applying a life-time criterion based on the local, i.e., core cell, temperature.

The TaT model [48] incorporates data from the VERCORS experiments as well as expert elicitations from the SOARCA UA to determine suitable fit coefficients (A and B) for the assumed Arrhenius equation:

$$\frac{1}{L(T)} = A\ exp(B\ T) \tag{3-29}$$

Where,

L(T) is the predicted life-time of the fuel rod in a core cell at a given temperature (T) and the coefficients $A = 2.16 \times 10^{-11}\ \mathrm{sec}^{-1}$, and $B = 7 \times 10^{-3}\ K^{-1}$ corresponds to the 50th percentile of possible Arrhenius curves.

The accrued damage as a function of time, $DF(t)$, is given below. A cumulative value of 1.0 corresponds to an instantaneous transition of the fuel in the core cell from intact rod geometry to particulate debris.

$$DF(t) = \sum \frac{1}{L(T)} \Delta t \tag{3-30}$$

3.2.3 Debris Addition from Heat Structure Melting

During degraded core conditions, many reactor vessel structures that are modeled by the HS package in MELCOR are subjected to intense radiative and convective heating, and may be expected to melt. These structures are often designated on input records COR_RP and COR_ZP as the radial and axial boundary heat structures for heat transfer from the core. An example of such a structure is the BWR core shroud, a relatively thin (5 cm) structure that surrounds the entire core and extends into the upper plenum.

Although the HS package does not model melting in general, the melting of these structures may be calculated by special application of the HS package degassing model,

using material type SS (see the HS_DG input records), and the resulting molten steel passed to the COR package. The melting model tracks the mass and volume changes associated with the molten steel added to the core. The model requires that any melting steel HS structure lie either along the core, corresponding directly with one of the axial segments represented in the COR package, or above the core.

The molten steel produced from the degassing model is passed to the outermost radial ring (NRAD) in the axial segment corresponding to the origin of the melt. It is entered as particulate debris with energy corresponding to fully molten steel with no superheat. The model is flexible to the extent that additional HS package structures above the core can also be identified to melt via the degassing model, with material passed to the uppermost axial segment (NAXL) in the outer ring. The candling model described in Section 3.1 and the particulate debris relocation logic discussed below performs any subsequent relocation of the molten steel from its initial core position.

3.2.4 Exclusion of Particulate Debris

Core cells need not be completely filled to block entry of particulate debris; debris can enter a core cell only if there is “free” volume in that cell. The free volume can be less than the fluid volume, because a component is allowed to exclude particulate (but not fluid or molten materials) from a volume greater than its physical volume. This can represent the natural porosity of a rubble bed, which does not allow other rubble to enter the pores. It can also represent an assumption that other interstitial spaces, such as those within fuel rod bundles, are too small to allow rubble to enter.

All intact components automatically exclude debris from the physical volume that they occupy. In a BWR, all initial components except the control blades are considered to occupy space in the channel region, with the blades occupying space in the bypass. (To be strictly precise, the core support plate is sometimes viewed as occupying space in the bypass, but only in a region where channel and bypass are not distinguished.) Particulate debris can occupy space in the channel (as PD), in the bypass (as PB), or in both.

Particulate debris is treated as forming a porous debris bed, which excludes other particulate debris from an effective bed volume, V_{bed}.

$$V_{bed} = \max\left(V_{material} , \frac{V_{unmelted}}{1-\varepsilon} \right) \tag{3-31}$$

Here, $V_{material}$ is the total volume of material in the particulate, $V_{unmelted}$ is the volume of that portion of the material that has never been melted, and ε is a user-defined porosity. The physical picture is that the unmelted particulate forms a debris bed with porosity ε, but that molten, or once-molten, materials may fill some or all of the pores. For a BWR, this treatment is applied separately to particulate in the bypass and in the channel.

MELCOR 1.8.5 includes a flexible and relatively straightforward capability to model the exclusion of particulate debris from other interstitial spaces. The model allows all components to exclude particulate debris from some minimum fraction of an associated total volume (channel or bypass) by their simple presence. (In cases where the associated volume is the one occupied by the component, particulate debris continues to be excluded from the total physical volume, if it is greater.)

The free volume in a core cell (or in the channel or bypass region of a core cell) represents the volume available for additional particulate debris to relocate into that cell. Such debris may relocate either from the cell above or from an adjacent cell on the same axial level. The free volume is defined as

$$V_{free} = max\left[V_{total} - \sum_{k} max\left(V_{material,k}, V_{excluded,k}\right), 0\right] \quad (3\text{-}32)$$

where the sum is over all components. For particulate debris, $V_{excluded,k}$ is the bed volume given by Equation (3-31). For all other components, it is a user-defined fraction of the total volume of the cell, if the component exists in the cell.

Based on examination of the geometry of typical US reactors, we would expect that no particulate debris could enter fuel bundles while there are intact fuel rods present. In BWRs, we would expect that particulate could not enter the unbladed bypass while there is intact CN, nor enter the bladed bypass while there is intact NS representing control blades. After the control blades have failed, this debris is free to enter the bladed bypass but not the unbladed bypass (assuming that CN is still intact).

The default exclusion fractions, selected in accordance with this picture, are shown in Table 3-2. In the table, RD means fuel rod; the exclusion is associated with the presence of FU, CL, or both. By default, the presence of intact fuel rods in a core cell excludes particulate debris from the entire channel region but has no effect on the bypass. The presence of intact CN excludes particulate from 30% of the bypass, representing the unbladed portion, and intact NS representing control blades exclude it from the remaining 70%. NS representing PWR control rods and SS have no effect.

Table 3-2 Exclusion of particulate debris by core components

Excluded fraction	Channel	Bypass
RD (FU, CN)	1.0	0.0[a]
CN		0.3[b]
CB		0.0[a]
NS[c]	BWR: 0.7[b] PWR: 0.0[d]	
SS[c]	0.0[e]	

[a] The default values for RD and CB allow failed control blades to slump without melting. A value of 1.0 for CB would exclude particulate from the bypass region while CB survives. A value of 1.0 for RD would exclude it from the *original* bypass region—even after the canisters have failed—while there are intact fuel rods. Such values could be used to prevent the slumping of unmelted rubble from control blades until the canisters, fuel rods, or both have failed.

[b] By default, CN excludes PB from the unbladed portion of the bypass while NS representing control blades excludes it from the bladed portion. These numbers are intended to represent a typical partition between unbladed and bladed bypass volumes in a BWR, taken here as 30%/70%.

[c] If there is a separate bypass region, SS and NS occupy that bypass and the exclusion fraction is applied to its volume. If there is none, as for NS representing PWR control rods and most cases of SS representing plates or control rod guide tubes, the fraction is applied to the total (i.e., channel) volume.

[d] In a PWR, NS is used to represent control rods. In Western designs, these rods have little ability to exclude debris in the absence of fuel rods. For a VVER (Russian PWR design), there are control assemblies that take the place of certain fuel assemblies when the reactor is shut down. If the exclusion fraction for NS is set to 1.0, particulate debris is prevented from entering these control assemblies until the control elements fail.

[e] SS is used to model core plates and BWR control rod guide tubes. This value allows particulate to enter core plates and be supported there and to fill around BWR control rod guide tubes without restriction.

Other analysts might want to examine the consequences of other assumptions when applying MELCOR to different reactor designs. For example, when VVER reactors are shut down, some of the fuel bundles are lowered out of the main core, with their place taken by control elements. Under the assumptions appropriate to U.S. designs, these control elements would have no capability to exclude debris and, when the upper core starts to collapse, the resulting debris would immediately spread into the rings containing these elements and fall to the lower plenum. Therefore, the default exclusion fractions can be modified globally, level by level, ring by ring, or cell by cell through user input. Consequences of default and alternate values are indicated in footnotes to the table.

These constraints on availability of space are considered in the models for radial and axial relocation of debris described in the following subsections. The absence of free volume is not allowed to prevent particulate debris from being *formed* in a core cell. For example, whenever a control rod or blade disintegrates, it is converted to particulate debris in place. The debris must be allowed to occupy the space previously occupied by the blade, even if geometric restrictions might have prevented any rubble from falling into that space. In addition, if a support plate has failed and lost the ability to support particulate debris, the absence of free volume within the plate is not allowed to prevent the passage of

particulate debris *through* it. This allows such particulate to continue to relocate downward to space available below the plate. Note that free volume, in the sense discussed here, is not relevant to the relocation of molten materials, which can fill all volume not physically occupied by materials; rather, it is the volume available to fluids.

3.2.5 Radial Relocation of Particulate Debris

The particulate debris leveling model is very similar to the molten material leveling model described in Section 3.1.5 except that, in the former, material is moved only from the particulate debris component in the deep ring to the particulate debris component in the shallow ring. Particulate debris is permitted to displace molten pool material in adjacent rings, and molten material backfills volume previously occupied by slumping solid particulate debris. The time constant for particulate debris relocation has an ad hoc default value of 360 s and is accessible as sensitivity coefficient C1020(1). There is no consideration of an angle of repose; debris is completely leveled across the core. Particulate debris relocation is subject to the same constraints concerning BWR canisters and core support plates as is molten material relocation. Component volumes and associated fission products are adjusted following relocations.

When particulate debris spreads radially, the distribution of refrozen conglomerate for particulate debris in both the donor and receiver cell is modified (Section 3.1.3). Conglomerate is moved from the top of the axial distribution in the donor cell to the top of the distribution in the receiver cell.

In the lower head, the particulate conglomerate volume that can spread radially inward is limited in order to retain a minimum volume in the cell to represent the volume of crust material refrozen on the vessel surface. This crust volume is calculated from component/vessel contact area, derived from component volume and cell geometry (curved lower head), and from the crust thickness obtained from the Stefan model. As material in the molten pool is frozen, it is transferred to particulate debris, and this minimum volume for retention increases. Similarly, as the crust freezes, this minimum volume decreases.

3.2.6 Gravitational Settling

The downward relocation of particulate debris by gravitational settling is modeled in MELCOR as a constant-velocity process whose velocity is given by VFALL from the COR_LP input record. Each ring in the COR nodalization is treated independently. For a given ring, each core cell containing particulate debris is considered in turn as a potential source of falling debris, working from the bottom up. The model first determines how far particulate from that cell can fall during the timestep, subject to limitations of available volume and support. Falling debris can be stopped by the absence of available space due to a blockage or by encountering a structure that can support it, typically, a support plate with the capability to support particulate. Note that, because of the debris exclusion

model discussed in Section 3.2.4, core cells can be blocked without being completely filled.

After the lowest core cell that particulate can reach has been determined, the algorithm fills the available space from that level until the debris in the source cell has been exhausted or all available volume has been filled. It then moves on to consider the next higher cell in the ring as a possible source of slumping debris, subject to the updated availability of space.

The model accounts for the distinction between particulate in the channel (or what was originally the channel), PD, and that in the bypass (or what was originally the bypass), PB. It allows particulate debris to slump from the channel or bypass of one cell into the channel and/or bypass of the cell below, depending on the conditions in those cells. The situation is complicated by the fact that the distinction between channel and bypass does not exist everywhere in the core. For example, cells such as those in the lower plenum that never contained canisters—and therefore can have no “bypass” region—are permitted to contain only PD. Thus, any debris that slumps into such a cell as PB must be considered there as PD. In addition, the distinction is almost entirely lost for cells that originally contained canisters once those canisters fail, and all particulate debris in such a cell is considered to be well-mixed and equilibrated. (However, separate volumes in the channel and the bypass must be calculated for such cells in order to define the volumes displaced in the associated CVH control volumes, which do remain distinct.)

The details of the algorithm implemented are as follows; it is to be understood that intact canister means component CB present in the cell:

(1). The split between channel and bypass regions is preserved when particulate debris slumps from a core cell with an intact canister into another core cell with an intact canister. That is, debris in the channel passes into the channel while debris in the bypass passes into the bypass;

(2). All particulate debris that originates in, or enters, a core cell where there is no intact canister to separate channel and bypass is treated as mixed, and any distinction between origin as PD or PB is lost;

(3). Particulate debris that slumps from a core cell without an intact canister into one with an intact canister is split between channel and bypass in proportion to the available cross-sectional areas of the two regions;

(4). If the fall of particulate is blocked in the channel or the bypass in a core cell that contains an intact canister, it fills that region from the blocked level up. If both are blocked, it fills each independently, based on debris entering the corresponding region. If there is enough debris to fill all available volume in the channel or bypass to a point above which there is no intact canister, any remaining debris is used to fill from that cell upwards;

(5). If the fall of particulate is blocked in a core cell that does not contain an intact canister, it fills from that level up. If channel and bypass are distinguished in that cell because it originally contained a canister, the particulate debris is divided between PD and PB in proportion to the available cross-sectional areas of the two regions. If there is enough debris to fill all available volume to a point above which there is an intact canister, the two regions are filled independently, based on the split of debris between them at the point where falling particulate entered the region containing a canister.

When particulate debris slumps into the adjacent cell beneath it, the distributions of refrozen conglomerate for particulate debris in both the donor and receiver cells are modified (Section 3.1). To represent the movement of particulate debris and conglomerate slumping from the bottom of the donor cell, the axial distribution in the donor cell is shifted downwards, and the distribution of the moved conglomerate is appended at the top of the conglomerate distribution in the receiving cell.

3.3 Molten Pool

In MELCOR 1.8.5, molten material was included as part of the particulate debris component, requiring that molten material be in thermal equilibrium with solid particulate. In addition, although a distinct composition was maintained for never-melted particulate debris, molten and refrozen debris were required to have the same composition. It was therefore impossible to distinguish adequately between molten and solid relocated material within a COR cell. Consequently, in MELCOR 1.8.6, two new components have been added to the COR package to represent oxide (MP1) and metallic (MP2) molten pool materials. In addition, there are corresponding components (MB1 and MB2) to track molten pool material in the bypass regions of a BWR (see Figure 3.6).

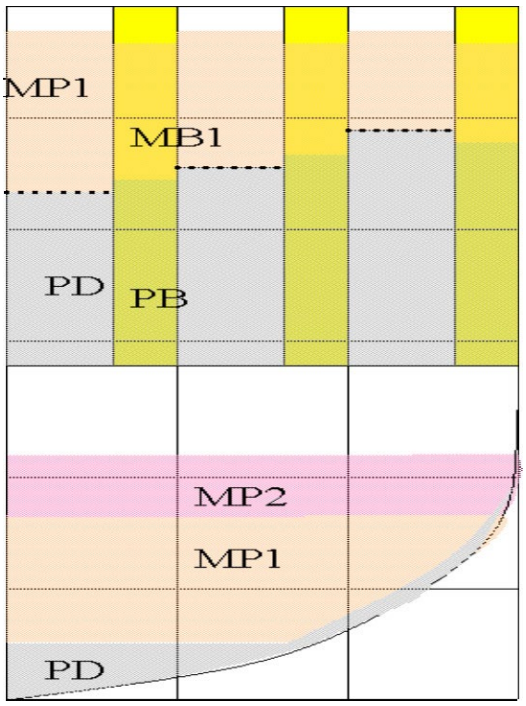


Figure 3.6 New molten pool components

Molten pool is formed as candling molten material is blocked due to support material or as a local blockage is formed from previously frozen conglomerate. Models were developed for these new components to define the characteristics of heat transfer and relocation. Unsupported molten pools slump in a manner analogous to particulate debris. However, now it is important to allow for displacement of molten pool materials by particulate debris. In addition, molten pool material is to be supported by a substrate (particulate debris) when the presence of a crust is detected. A crust is formed as a local blockage due to refrozen material. If a crust is not present, then molten pool material is free to relocate downward or is allowed to fill any available interstitial particulate debris volume where it is equilibrated and moved to the particulate debris component.

Contiguous volumes containing these components comprise physical molten pools that are assumed to be uniformly mixed by convection and have uniform composition and temperature.

3.3.1 Slumping and Displacement

Several assumptions have been made regarding stratification and displacement of materials in the degraded core. It is assumed that particulate debris sinks into a molten pool, displacing the molten pool volume. If the molten pool volume is part of a contiguous convecting pool retained behind a local blockage, it does not relocate below the local blockage and is moved from particulate debris to the intact component of the molten pool

component. Furthermore, if the pool stratification model is enabled, it is also assumed that the oxide and metallic materials are immiscible and separate into distinct molten pool components. The oxides (MP1) are assumed to be denser than the metallic materials (MP2) and therefore to displace the metallic molten pool material. In a BWR, this displacement is complicated by the existence of canister walls separating channel and bypass volumes. No material can flow through the canister walls or through an intact crust. However, as the canister is degraded, flow paths may open and relocating material may be partitioned (by available area) between channel and bypass volumes.

The algorithm to account for the displacement of components is illustrated by the simple case of MP1 displacing MP2 in cells (see Figure 3.7). For each cell containing MP1, the volume presently occupied by MP2 in cells is made available for further downward relocation of MP1. The net transfer is accomplished first by relocating MP2 upwards into the volume to be vacated by MP1. In the next step, MP1 is transferred downward into the volume vacated by MP2 Checks are made for small volumes to prevent numerical problems, and special logic is applied to split flow through canister openings based on areas. The detailed procedure is as follows:

(1). Check for small volumes (less than roundoff) to prevent subsequent numerical problems

(2). Consider separate flow paths for channel and bypass volumes (allow mixing at openings)

(3). Determine volume available for MP1 relocation (fluid volume occupied by MP2)

(4). Split flow at canister openings based on area

(5). Find lowest elevation for relocation

(6). Relocate MP2 upward into volume to be vacated by downwardly relocating MP1

(7). Increase MP2 masses, volumes, and enthalpies in ‘To’ cells

(8). Check for small (less than roundoff) volumes left behind

(9). Set up all RN moves and make them

(10). Remove MP2 masses, volumes, and enthalpies in ‘From’ cells

(11). Relocate MP1 downward into volume just vacated by MP2

(12). Increase MP1 masses, volumes, and enthalpies in ‘To’ cells

(13). Check for small (less than roundoff) volumes left behind

(14). Set up all RN moves and make them

(15). Remove MP1 masses, volumes, and enthalpies in ‘From’ cells

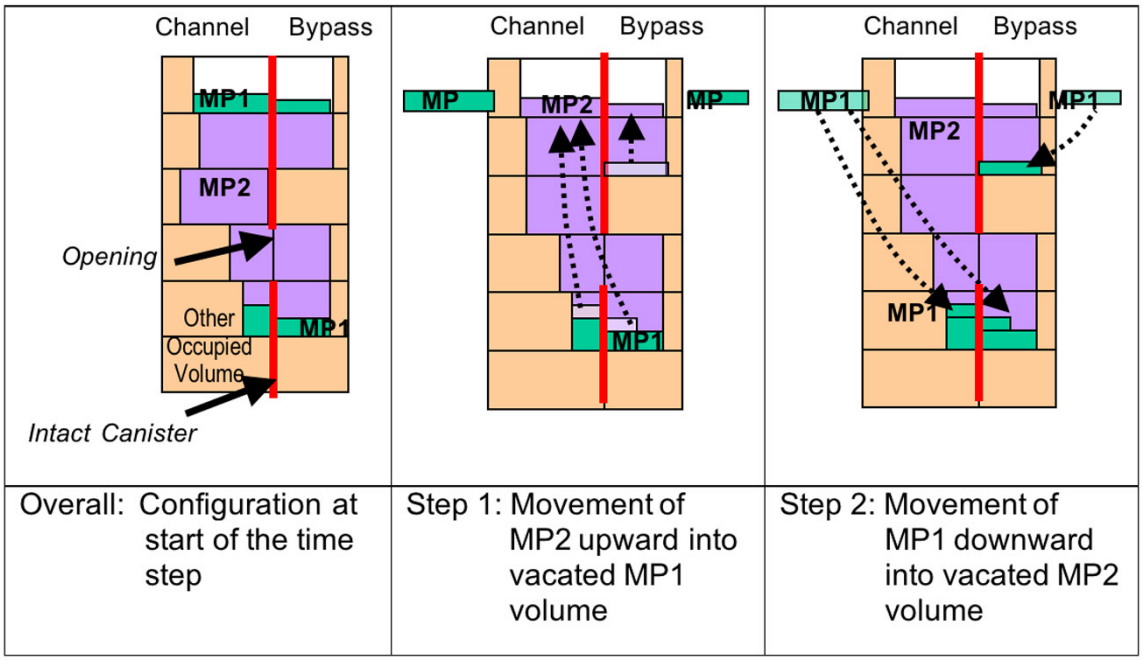


Figure 3.7 Algorithm for displacement of MP1 and MP2 components

Note that these models do not take into account heat transfer or interactions as material sinks into a molten pool and material relocation is limited by a blockage or by the distance it can travel in a timestep, assuming that it is falling at velocity VFALL as provided on the COR_LP input record.

3.3.2 Contiguous Molten Pools

As already discussed, contiguous volumes containing molten pool components constitute coherent molten pools that are assumed to be uniformly mixed by convection, so as to have uniform material composition, radionuclide composition, and temperature. Two distinct molten pools are allowed in the lower plenum (oxide and metallic), and potentially four molten pools can be modeled in the upper core (oxide and metallic in channel, and oxide and metallic in bypass volume). A search is made in the core and lower plenum to find the largest contiguous molten pools (by volume), which are then modeled as convecting molten pools. This requirement for contiguity ensures that isolated cells containing molten materials are not mixed with the convecting pools. These convecting molten pools transfer heat to the lower head (lower plenum pools), fluids (water or steam), and substrate material. In addition, there is transfer of heat and radionuclides that occurs between stratified molten pools. New models have been added to predict heat transfer coefficients to the substrate supporting the molten pool, heat transfer between pools, and heat transfer to surroundings. Note that isolated volumes of molten pool material are not part of these contiguous molten pools and are not included in the convective mix. They

have distinct temperatures and composition and transfer heat as discussed in previous sections.

3.3.3 Partitioning of Radionuclides

In MELCOR 1.8.5 all fission products are hosted on and relocate with UO_2 material. The capability to transfer RN masses to other materials, perhaps in the metallic phase (Zr, Fe, and AG-IN-CD control rod poison), was added to MELCOR 1.8.6 where partitioning factors are implemented as control functions. The partitioning factors are defined as the ratio of the mass concentration of the radionuclide class in the metallic pool to the mass concentration of the radionuclide class in the oxide pool; in other words,

$$f_{partition,i} = \frac{\frac{Mass\ of\ RN_i in\ pool_2}{Mass\ of\ pool_2}}{\frac{Mass\ of\ RN_i in\ pool_1}{Mass\ of\ pool_1}} \tag{3-33}$$

A partition factor is defined for each RN class so that RN classes can be treated independently. The default is to retain all RN masses in the oxide phase. Logic has also been added to redistribute fission products between metallic materials as materials may be depleted (oxidation, etc). Physical models for partitioning fission products between condensed phases [49] can be modeled and tested in MELCOR via these user-defined control functions.

3.4 Displacement of Fluids in CVH

When core materials relocate from one core cell to another by any of the mechanisms discussed in the previous sections, they cease displacing fluid in the old location and commence displacing it in the new one. Candled material (conglomerate debris) is treated as occupying space in the same region, channel, or bypass, as the component that supports it. As already implied, each core component (fuel rods, control elements, canisters, and particulate debris) is treated as occupying space in an associated CVH control volume. However, the spatial nodalizations used in COR and CVH are largely independent, and may be quite different. The two representations are maintained independently throughout a MELCOR calculation.

In order to treat the displacement of fluid in CVH, each control volume is considered to have virtual volume in addition to the current fluid volume. The virtual volume includes the total volume of all core components within the volume. Part or all of it becomes available to CVH fluids when these core components relocate. The COR package handles relocation by directing CVH to free virtual volume in the original location and occupying it in the new location. The vertical distribution of the virtual volume is defined only within the resolution provided by the Volume/Altitude (V/A) table for the CVH volume. (See the CVH/FL Packages Reference Manual for more details about V/A tables and virtual volume.) The most detailed agreement is obtained if the elevations in the CVH V/A table

match those in the core nodalization. Checks included as part of MELGEN input processing generate warning messages if the CVH V/A tables in CVH input do not correspond to all axial limits of core cells in COR input.

Further checks are included as part of MELGEN input processing to ensure that the CVH and COR representations of the distribution of fluid volume are compatible. If there is an initial inconsistency, an error message is generated and processing terminated without generation of a restart file. (An option is available to override these checks if the user is determined to continue with inconsistent data.) Specifically, the tests require that no fluid volume in COR may exceed those in CVH, thus ensuring that core debris cannot overfill the CVH volume. The requirement enforced is on the total fluid volume in all core cells (or fractions of core cells) associated with each range of elevations in the V/A tables for each CVH control volume. Separate checks are performed for channel and bypass regions.

One subtle point must be dealt with to maintain consistency between the representations as a simulation progresses. Although each component is assumed to displace fluid in either the channel or the bypass but not in both, canisters have two sides that interface with different volumes and that may oxidize independently. By convention, canisters are assumed to occupy the channel (there would be no essential difference if they were assumed to occupy the bypass). If the interior of a canister is filled, any further oxidation of its inner surface is precluded, but steam and/or oxygen present outside (in the bypass volume) can continue to oxidize its outer surface. Because the volume of oxide produced is greater than the volume of metal consumed, this produces a volume of oxide that cannot be accommodated in the channel but that must be put somewhere.

The solution devised for this conundrum involves borrowing the necessary volume from the bypass. Thus, if there is more material associated with channel components than can be accommodated there, the excess is treated as reducing the fluid volume in the bypass. In a sense, canisters can occupy bypass volume when necessary. Borrowing of channel volume by bypass components is also allowed, in the interest of symmetry, but should be necessary only in cases involving round-off. The borrowing is, of course, limited to the actual fluid volume available.

A single call at the completion of the advancement in the COR package communicates the net changes in occupied volumes as calculated within the COR package to the CVH package. They are converted to the nodalization used by CVH, for later use in advancing the hydrodynamic equations. For each portion (channel or bypass) of each core cell, the quantity actually communicated is the negative of the change in fluid volume, rather than the sum of changes in occupied volumes. This insulates CVH from the details of volume borrowed within the core representation.

As a simulation advances in time, the COR package repeatedly rechecks the internal consistency of its representation of volumes and warns of any discrepancies that may develop. The treatment of errors is controlled by elements of sensitivity coefficient array, 1504. If the borrowed volume in any core cell exceeds a limit set by C1504(2), a warning

message is issued. (Issuance of the message is terminated after 100 such messages in any execution.) Checking of total volume occupied in a core cell is only a test on the logical consistency of the coding. If the total occupied volume exceeds that available by more than a limit set by C1504(1), an error message is issued (also terminated after 100 such messages). If it exceeds that limit by a factor of 100 times, the calculation is terminated.

If the representations of volumes within CVH and COR are initially consistent, the one in CVH should remain consistent while that in COR does. However, even if they are initially consistent, the CVH and COR representations remain independent. Therefore, it is still possible that they diverge as a result of accumulated round-off. If the divergence is great enough, the COR package may attempt to relocate debris to regions where there is no volume in CVH to accommodate it. If this occurs, a warning message is issued, but the calculation can continue.

4. Control Rod Silver Release Model

The silver release model describes release and drainage from control rods containing silver indium cadmium alloy following failure of the stainless steel cladding on the control rod. Released control alloy is allowed to candle via the COR candling models and can transfer material to RN classes via a vaporization model. Reheated conglomerate containing control alloy can also release to RN by vaporization. Once in RN, the alloy is treated by the RN vapor/aerosol transport and deposition models and by the RN model for vaporization/condensation on heat structures.

4.1 Description of Control Rods and Failure Scenarios

Control rods containing silver-indium-cadmium alloy are generally used in PWRs. The alloy composition initially consists of 80% silver, 15% indium, and 5% cadmium, by weight. The control rod is constructed something like a fuel rod, with the control alloy clad with a stainless-steel sheath filled with helium. The control rod is inserted in a guide tube made of stainless steel or Zircaloy.

There are two somewhat different failure scenarios for control rods that depend on system pressure versus the internal pressure of the control rod. The internal pressure of the control rod is due mostly to the high vapor pressure of the cadmium together with contributions from the other components and the helium fill gas.

At low system pressure, the control rod can bulge as the stainless-steel sheath approaches its melting point and loses strength. If the guide tube is stainless steel, the control rod fails at around 1720 K. If the guide tube is Zircaloy, the bulged sheath can contact it and form a eutectic, resulting in failure at the somewhat lower temperature of 1470 K. On failure at low system pressure, the molten alloy is ejected under pressure, resulting in some initial aerosol formation.

At high system pressure, the control rod fails when the stainless-steel sheath loses strength at 1720 K. The molten alloy is not under a high-pressure differential and flows down the control rod.

4.2 Model Implementation in MELCOR

The silver release model is based on the VAPOR model [50, 51]. The VAPOR model describes the release, drainage, and vaporization of control rod alloy under high system pressure conditions. Consideration of low-pressure phenomena in MELCOR was not attempted for this first model due to lack of data and internal MELCOR architecture constraints. This is further discussed in Section 4.3. The model consists of several main functions: control rod failure, internal drainage of the control rod to the break, flow down the control rod, and vaporization from the flowing film. These parts correspond roughly to the VAPOR model. Vaporization from reheated conglomerate is also allowed.

Control rod failure occurs at the stainless steel melting temperature, using the present MELCOR failure criterion. Upon failure, as much as the total silver-indium-cadmium mass in the failed COR node is available to be transferred to conglomerate in that timestep. The transfer rate through the failure is calculated using a quasi-steady solution to Bernoulli's equation, using the available head of molten alloy inside the control rod [50]. The release velocity is given as

$$V_r = \left[\frac{2gh}{1 + K_c - (A_2/A_1)^2}\right]^{1/2} \tag{4-1}$$

where

V_r = Release velocity (m/s),
g = gravitational constant (m^2/s),
h = head of molten alloy (m),
A_2 = total area of break in ring (m^2),
A_1 = cross-sectional area inside control rods in a ring (m^2), and
K_c = form loss coefficient at break.

The equation for the head can be solved analytically as

$$h(t) = \left(h_0 - \frac{\gamma t}{2}\right)^2 \tag{4-2}$$

where

h_0 = initial head at t = 0 (m) and

t = time since initial break (s)

and

$$\gamma = \left[\frac{2g(A_2/A_1)^2}{1 + K_c - (A_2/A_1)^2}\right]^{1/2} \tag{4-3}$$

The form loss coefficient is

$$K_c = \begin{cases} 0.32\left(1 - \dfrac{A_2}{A_1}\right) & if \quad A_2 < 0.4712A_1 \\ \dfrac{A_2}{A_1} & if \quad A_2 > 0.4712A_1 \end{cases} \tag{4-4}$$

This approach is valid because, for small breaks where $A_2 << A_1$, the gravitational head is fairly constant, whereas for large breaks where $A_2 \sim A_1$, release is so rapid (seconds) that a more detailed analysis is not necessary. Given the release velocity, the mass release rate for a COR node is

$$\dot{m}_r = \rho_{AIC} A_2 V_r \tag{4-5}$$

where

$\dot{m}_r$ = mass release rate for node (kg/s)

ρ_{AIC} = density of alloy (kg/m^3)

Release of more material than is in the failed COR node during one timestep is not allowed. Possible drainage of alloy from nodes above is calculated at the beginning of each timestep by moving molten alloy down, filling each node starting from the break successively. The head for each node is also calculated at the beginning of each timestep by summing the amount of molten alloy above each node in each ring.

Once released to the conglomerate field, the molten alloy is allowed to drain using the existing COR candling model. While it is draining, alloy can vaporize and be released to the RN classes via a vaporization model.

4.2.1 Vaporization Model

The vaporization model is a simplification of the actual vaporization, which would in principal allow the individual alloy components to vaporize at rates according to their

individual vapor pressures as modified by Raoult's law (in reality, the vaporization is nonideal). The simplification is necessary because MELCOR only considers one control poison mass, not a mass with three components. Accordingly, vaporization of alloy mass is assumed to occur at the rate for the component with the lowest vapor pressure (silver), and the composition of the remaining unvaporized alloy is not changed. In reality, cadmium would vaporize quickest, followed by indium, and then silver, and the composition of the molten alloy would change.

The vaporization rate from a flowing film or from conglomerate into the gas field is calculated using a mass transfer analogy to heat transfer as

$$\dot{m}_{vap} = h_m A_c M_{Ag} \left(\frac{P_v(T_c)}{RT_c} - \frac{P_{Ag}(T_b)}{RT_b} \right) \tag{4-6}$$

where

h_m = mass transfer coefficient (m/s),

A_c = molten alloy surface area (m^2),

M_{Ag} = molecular weight of silver (kg),

R = gas constant,

T_c = temperature of molten alloy (K),

T_b = bulk gas temperature (K),

P_v = vapor pressure of silver (Pa),

P_{Ag} = partial pressure of silver vapor in the bulk gas (Pa), and

$\dot{m}_{vap}$ = vaporization rate of molten alloy (kg/s).

This model assumes that vaporization is controlled by mass diffusion in the bulk gas. The effect of the heat of vaporization on cooling the molten alloy is ignored. Condensation of vapor on the molten alloy is not allowed; once the vapor is transferred to the RN classes, the usual RN vaporization/condensation models for heat structures apply. The total mass transfer rate is apportioned among the three control rod component classes in RN according to their original mass fractions in the control rod alloy.

4.3 Model Limitations

There are several model limitations, as implemented in MELCOR. The limitation on the realism possible in the vaporization rate has already been noted, as well as not being able to change the composition of the molten alloy. Another limitation is that a pressure differential is not calculated for use in calculating the break velocity—only the gravitational head is used. This is because of the lack of geometry information in MELCOR—the internal control rod pressure and the rate at which it decreases following a break depend

on the internal free volume of the control rod. The internal volume is not input to MELCOR—in fact, only the alloy mass, node heights, and total cross-sectional area of rods per ring are available. It was therefore decided not to include the effect of internal rod pressure in the release model.

Another limitation is that only the total cross-sectional area of rods per ring is input, which is the cross-sectional area including the guide tubes. The internal cross-sectional area of the control rod and number of rods per ring are not available. Looking back at the release equations, only the area ratio is necessary for calculating the release velocity; this ratio is input to the model. The break area per ring, appearing in the equation for mass release rate, can be gotten from the cross-sectional area using this ratio. Because the cross-sectional area in the code is the total area including the guide tubes, the release rate is somewhat in error unless the input area ratio is corrected to account for the difference between the inside flow area of the control rods and the total area.

5. Structure Support Model

5.1 Model for SH and FM components

In MELCOR 2.x, what was available as other structure component (OS) in previous versions is now treated as FM (only permitted for a PWR). There are no mechanical models for the SH and FM components. Therefore, only simple parametric models are available for failures. The SH component has the property that it can fail, allowing debris to relocate into the peripheral bypass volume between the shroud and the core support barrel. Debris in the bypass volume can then be relocated downward, supported by the FM component until the FM component itself fails. The FM component fails when the temperature of the FM reaches the failure temperature defined for that axial level by TSFAIL on input record COR_SS. Failure also occurs if an optional logical control function defined on input record COR_CFF becomes TRUE. Upon failure, all components supported by the FM are converted to particulate debris (PD), which, with any PD previously supported by the FM, are allowed to fall through to lower cells. The FM component itself remains in place until it melts.

5.2 Models for SS

The supporting structure component, SS, in any core cell may be treated as representing an edge-supported plate, a grid-supported plate, a BWR core plate, or BWR control rod guide tubes. The model used is determined by user input on records COR_SS where these four models are associated with the keywords “PLATE,” “PLATEG,” “PLATEB,” and “COLUMN,” respectively

There are differences in the ability of each form of SS to support other intact components and particulate debris, and in the resulting loads on and stresses in the structure. Failure of the structure may be based on the calculated stresses. Parametric models equivalent to those for SS are also available. The consequences of failure (in terms of which

components collapse) also differ for the various models. Subsections 5.2.1–5.2.4 describe the four models. Subsection 5.2.5 describes the failure models.

5.2.1 The PLATEG Model

The PLATEG model represents a plate that is supported by an underlying array or grid of beams, which may be formed as an integral part of the plate. In general, the beams have enough strength that their failure is not an issue, and the interest is in failure of the web between them. PLATEG is not dependent on support from SS in any other core cell. After failure, the plate element remains in place until it melts.

Until it fails, PLATEG in each cell supports itself, intact components, and debris above it and is loaded by that total weight. When failure occurs in any ring, only the capability to support PD and intact components in cells above is removed. Thus, everything resting on the plate falls, but the plate remains in place until it melts.

For small deflections of solid plates, the stress is related to the bending moment per unit length, M, by

$$\sigma_e = 6\frac{M}{h^2} \tag{5-1}$$

where *h* is the thickness of the plate.

Only numerical solutions are available for most cases involving uniform loading of plates with underlying supports. If the support involves a rectangular grid of stiff beams of negligible width, the maximum bending moment for use in Equation (5-1) occurs at the point of support at the midpoint of the longer edge, and is given by [52]

$$M = K\ q\ x^2 \tag{5-2}$$

where *q* is the load per unit area, *x* is the short dimension of the supporting grid, and *K* is a function of the aspect ratio of the supporting grid.

The stress in the plate in any ring is calculated as

$$\sigma_{e,ring,PLATEG} = 6\ \frac{K\ q\ x^2}{h^2} = 6\ K\frac{x^2}{A_{ring}}\frac{1}{h^2}W_{ring} \tag{5-3}$$

Here W_{ring} is the load carried in that ring and A_{ring} is the ring area. By default, the PLATEG model uses a value of *K* corresponding to a square supporting grid of beams (or an eggcrate plate) and Poisson's ratio $\nu = 0.3$ appropriate to stainless steel, for which

$$K_{square} = 0.0513 \tag{5-4}$$

The value for K_{square} may be changed through user input record, COR_SS.

5.2.2 The PLATE Model

The PLATE model represents a simple edge-supported plate that may span more than one ring of the core. It initially supports itself and intact components above it and is loaded (as a whole) by its own weight and that of the other supported components, including particulate debris. Inner rings of the plate are allowed to fail before the outer ones, leaving the outer portion of the core still supported by the annular remains of the plate. If the failure mechanism is stress based, the local stress is calculated as a function of the total load, the position in the plate, and the fraction of the plate that has not yet failed.

When failure occurs in any ring, support is removed for the SS representing the portion of the PLATE in that ring and any surviving inner rings, as is support for intact components and PD in cells above these. Thus, a failed section of the plate and everything resting on it is converted to PD and be allowed to fall, taking with it any unfailed inner rings of the plate together with everything resting on them. The outermost ring of the plate is treated as self-supporting until it fails.

As with the PLATEG model, the stress is related to the bending moment per unit length through Equation (5-1). For uniform loading of a round plate of constant thickness, the bending moments vary radially, and the tangential moment is always greater than the radial moment. The value of the tangential bending moment per unit length, denoted as $M_0(r)$, is [53]

$$M_0(r) = K_{M,0}(r)\ q\ a^2 \tag{5-5}$$

$$K_{M,0}(r) = \frac{3+\nu}{16}\left[1 - \frac{1+3\nu}{3+\nu}\left(\frac{r}{a}\right)^2\right] \equiv K_0\left[1 - K_1\left(\frac{r}{a}\right)^2\right] \tag{5-6}$$

where q is the load per unit area, a is the radius of the plate, and ν is Poisson's ratio. In Equation (5-6), $K_0 = 0.125$ and $K_1 = \nu$.

Any variation in loading across the plate is neglected, and q is considered to be the total load on the plate divided by its total area. Equations (5-5) and (5-6) capture the variation of the bending moment, and therefore, the stress from the center to the outside of an intact plate.

If an inner ring of the plate fails before the outer ones, it leaves the outer portion of the core still supported by the annular remains of the plate. Although the resulting configuration is surely messy, one can expect certain qualitative changes in the stress pattern. The decrease in the total load on the plate tends to decrease stresses, while the loss of the stiffness of the central portion tends to increase them. The dominant effect of the formation of a central hole in the plate by failure of inner rings is a stress concentration

that tends to accelerate the failure of the innermost surviving ring. The magnitude of the effect decreases as the hole grows to include a substantial fraction of the original plate.

The model implemented in the MELCOR COR package uses a very simple expression to capture these effects, in the form of

$$M_0(r;r_0) = K_M(r;r_0)\ q\ a^2 \tag{5-7}$$

$$K_M(r;r_0) = K_{M,0}(r)\ \left[1-\left(\frac{r_0}{a}\right)^2\right]\ \left[1+\left(\frac{r_0}{r}\right)^2\right] \tag{5-8}$$

Here r_0 is the size of the hole, the first factor in Equation (5-8) reflects the reduction in load, while the second factor reflects the stress-concentrating effects of the hole. Note that this equation can be considered to be the general form. It is exact in the absence of a hole (r_0 = 0), where

$$K_M(r;0) = K_{M,0}(r) \tag{5-9}$$

and Equations (5-7) and (5-8) become equivalent to Equation (5-5).

The approximation given by Equation (5-8) has been compared [54] to the exact solution for a uniformly loaded annular plate with a free inner boundary and simple edge support at the outer boundary [55]. The simplified form agrees quite well with the exact solution—rather better, in fact, than the exact model—corresponding to the expected geometry of a degraded core.

Under the assumption of continued uniform loading of the surviving portion of the plate, the total load on the plate may be written as

$$W_{total} = \pi(a^2 - r_0^2)\ q \tag{5-10}$$

W_{total} is evaluated as the total load on the entire plate or annulus, summed over all core rings in which an unfailed portion of the plate is present. For a given ring, the stress is greatest at its inner edge; for the innermost ring, at the center of the plate. Therefore,

$$\sigma_{e,ring\ 1} = 6K_0\frac{1}{\pi}\ \frac{1}{h^2}\quad W_{total} \tag{5-11}$$

$$\sigma_{e,ring\ i>1} = 6K_0\frac{1}{\pi}\ \frac{1}{h^2}\left[1-K_1\left(\frac{r_{i-1}}{a}\right)^2\right]\quad\left[1+\left(\frac{r_0}{r_{i-1}}\right)^2\right]W_{total} \tag{5-12}$$

Here r_i is the outer radius of ring i, and the coefficients K_0 and K_1 are defined by Equation (5-6). The values used by default correspond to a Poisson's ratio, $\nu = 0.3$, for

which K_0 = 0.206 and K_1 = 0.576. These coefficients can be changed through user input record, COR_SS.

5.2.3 The PLATEB Model

For a BWR, the primary support of the core is the control rod guide tubes (CRGTs), functioning as columns. The core plate is supported by beams and is loaded only by its own weight and that of debris on it. Although it does not bear the weight of the fuel and canisters, the presence of the plate is required for the CRGTs to support them.

SS, representing PLATEB, is not dependent on support from SS in adjacent radial rings, or in any other core cell. When the plate fails in any ring, it loses the ability to support PD, which then falls, but the plate remains in place until it melts.

Stresses in the plate for the PLATEB model are calculated using Equation (5-3) in the case of beam support without cross beams, neglecting the fact that supporting beams span more than one ring of the core. The value of *x* in this equation is the spacing between the beams. The differences from the PLATEG model are that the loading is limited to the weight of the plate and any PD resting on it and that the default value of *K* is taken as

$$K_{beam} = 0.0833 \tag{5-13}$$

This corresponds to the limit of the grid result cited above for an infinite aspect ratio of the grid and a Poisson's ratio of $\nu = 0.3$. The default may be changed by user input record, COR_SS.

5.2.4 The COLUMN Model

SS, representing an unfailed COLUMN in a core cell, directly supports SS modeled as COLUMN in the level immediately above. Failure of SS, representing COLUMN in one core cell, implies failure of contiguous COLUMN elements higher in the same radial ring, resulting in their collapse to PD.

In MELCOR 1.8.5, the lowest element of a COLUMN was always treated as being self-supporting; it would not collapse until it itself failed. MELCOR 1.8.6 and 2.1 allows transfer of loads from the lower end of a column to a plate (PLATE or PLATEG) below it. Only the lowest element of a column *not* resting on a plate (including an element in the lowest axial level of the core) is treated as self-supporting. As part of the change, a distinct SS-type name, ENDCOL, is now assigned to those elements of COLUMN that are self-supporting. The change is performed automatically as part of input processing for those elements that are self-supporting under the interpretation above. The name is also accepted as input on COR_SS input records and can be used to define self-support for interior elements of what would otherwise be considered to be a continuous column.

If there is SS modeled as PLATEB in the level above an unfailed COLUMN, the COLUMN indirectly supports (and is further loaded by) intact fuel assemblies, canisters, and control blades in and above that level, but not the plate or any PD. The internal coding logic treats the PLATEB as if it supported the intact components in the levels above (without being loaded by them) by transferring the load to the COLUMN in the cell below. This support is dependent on the existence of the COLUMN. If it fails (or is initially absent), the fuel assemblies and control blades supported by PLATEB immediately collapse to PD.

For thick columns in compression, the relationship between stress and load is simply

$$\sigma_e = \frac{W}{A_c} \tag{5-14}$$

where W is the load (including the column itself and the indirectly supported fuel assemblies, canisters, and control blades) and A_c is the cross-sectional area. If there are N identical circular columns in a ring of the core nodalization, each with inner radius r_i and outer radius r_o, the stress is evaluated as

$$\sigma_{e,ring,COLUMN} = \frac{1}{N\pi\left(r_o^2 - r_i^2\right)} \frac{M_{column,0}}{M_{column}} W_{ring} \tag{5-15}$$

The factor of the ratio of the original column mass to its current mass is included to account for any reduction in the load-bearing area of the column by oxidation or melting.

5.2.5 User Defined SS Types

MELCOR 1.8.6 includes a new, limited capability for the user to define additional types of SS with properties specified by input. Although only parametric failure models are allowed, a LOGICAL control function can be written to represent a stress-based failure mechanism. This feature is also available in MELCOR 2. For more details, see the COR Package Users' Guide.

5.3 SS Failure Models

Several mechanisms for failure of structures are included in the modeling of supporting structures using the SS core component. These include equivalences to the failure temperature and control function models used for the PWR FM component. There are also mechanical models that consider the stresses in SS, as calculated from the models in the preceding subsections.

The stress-based failure models include the failure of plates and columns by yielding and the failure of columns by buckling. These are both catastrophic failure models. In addition, structures can fail over time by creep at stresses below the yield stress. This possibility is represented using a Larson-Miller creep-rupture model, which is closely related to the

(default) zero-dimensional form of the model for failure of the lower head described in Section 6.2.

5.3.1 Failure by Yielding

Unless a parametric model has been specified, failure occurs if the stress in a structural element exceeds the yield stress. For this analysis, the stress is calculated using equations in Section 5.2 for the loading model specified on the relevant COR_SS input record. The temperature-dependent yield stress is represented by the following equation, which has a form like that used for the lower head:

$$\sigma_Y(T) = 260.\times 10^6 \left\{ \left[1 + \left(\frac{T}{800.} \right)^{3.} \right]^{-1} - \left[1 + \left(\frac{1700.}{800.} \right)^{3.} \right]^{-1} \right\} \tag{5-16}$$

The constants 260. x 10^6, 1700., 800., and 3., which were chosen to approximate the data for 304 stainless steel in the *Nuclear Systems Materials Handbook* [56], have been implemented as sensitivity coefficient array, C1606.

5.3.2 Failure by Buckling

Columns buckle if the load exceeds the value given by [57]

$$W_{buckl} = \pi^2 E \frac{I}{\ell^2} \tag{5-17}$$

where *I* is the moment of inertia, *E* is the elastic modulus, and ℓ is the length of the column. For a circular column with outer and inner radii r_o and r_i, respectively, the moment of inertia is

$$I = \frac{\pi}{4}\left(r_o^4 - r_i^4\right) = \frac{\pi}{4}\left(r_o^2 - r_i^2\right)\left(r_o^2 + r_i^2\right) \tag{5-18}$$

Comparison with Equation (5-15) shows that for *N* identical columns in a ring, buckling occurs if the stress exceedsa

$$\sigma_{buckling} < \pi^2 \frac{\left(r_o^2 + r_i^2\right)}{4N\ell^2} \quad E \tag{5-19}$$

The elastic modulus is represented by the following equation, which has a form like that for the lower head:

$$E(T) = 370.\times 10^9 \left\{ \left[1 + \left(\frac{T}{1650.} \right)^{3.} \right]^{-1} - \left[1 + \left(\frac{1700.}{1650.} \right)^{3.} \right]^{-1} \right\} \tag{5-20}$$

The constants 370. x 10^9, 1700., 1650., and 3. were chosen to approximate the data for 304 stainless steel in the *Nuclear Systems Materials Handbook* [58]. They are implemented as sensitivity coefficient array, C1605.

5.3.3 Failure by Creep

The Larson-Miller creep-rupture failure model [59] gives the time to rupture, t_R, in seconds as

$$t_R = 10^{\left(\frac{P_{LM}}{T} - 16.44\right)} \tag{5-21}$$

where the temperature, T, is in K. The Larson-Miller parameter, P_{LM}, for stainless steel can be fit as a function of the effective stress, σ_e, in Pa, as

$$P_{LM} = 81000 - 7500\, log_{10}(\sigma_e) \tag{5-22}$$

from ASME data [60]. The three constants in Equations (5-21) and (5-22) have been implemented as sensitivity coefficient array, C1604.

Because stress and temperature are not constant, a fractional lifetime rule is applied, and failure is assumed to occur when

$$\int \frac{dt}{t_R(t)} \approx \sum \frac{\Delta t_i}{t_R(t_i)} = 1 \tag{5-23}$$

For the multiring geometry of MELCOR, both the loading and the temperature histories—and therefore the Larson-Miller parameter—vary from cell to cell, and SS in different cells is allowed to fail independently. It is therefore necessary to integrate Equation (5-23) separately for each core cell that contains SS subject to a stress-based failure model.

6. Lower Head Model

The lower head nodalization framework was described in Section 1.1.2; the illustration in Figure 1.9 is repeated here as Figure 6.1 for convenience and with more detail depicting the lower head heat transfer logic. The lower head model physics described in this section is divided into four parts: heat transfer among the model elements, melting of the lower head, determination of failure at some penetration or gross failure in a segment (when penetrations are absent), and ejection of debris into the reactor cavity. Where phenomena associated with lower head failure is very poorly understood, such as penetration failure, the models are very simple and parametric, allowing the user significant flexibility in controlling lower head behavior.

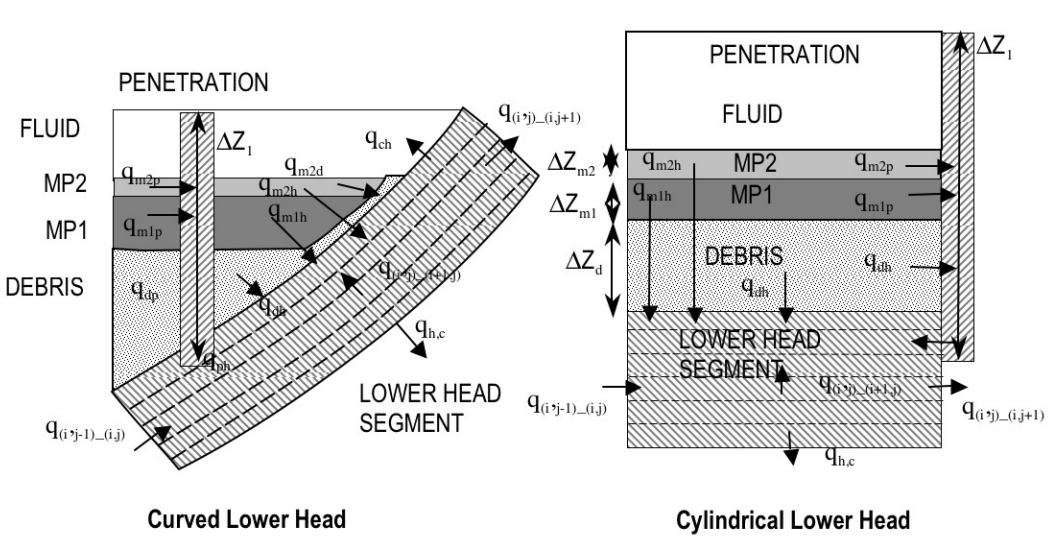


Figure 6.1 Lower head nodalization

For MELCOR 1.8.6 and 2.1, heat transfer is calculated for each lower head segment (rather than ring), for which multiple segments may intersect with a single-core cell. The outer boundary condition for each segment is derived from heat transfer to an outer cavity control volume, which may vary and is specified for each segment. Also, penetrations are specified for each segment and vessel, or penetration failure is determined at the segment level.

6.1 Heat Transfer

In MELCOR 1.8.5, heat transfer to the lower head and its penetrations (e.g., instrumentation tubes, control rod guide tubes, or drain plugs) are considered to be heat transfer from particulate debris to the lower head; heat transfer from particulate debris to penetrations; conduction from the penetrations to the lower head; and convective heat transfer from the penetration, debris, and lower head surfaces. In MELCOR 1.8.6 and 2.1, we also consider heat transfer from the molten pools, either directly in contact with the lower head or conducting through a crust that is calculated from the Stefan calculation. Heat transfer areas are calculated based on debris and molten pool depths as well as from input geometry (lower head radius specified on record COR_VP) and ring radii (specified on the COR_RP records) and account for the curved geometry of cells in contact with a hemispherical lower head. The Stefan model calculates the thickness of the crust that has refrozen on the surface of the lower, and the temperature is obtained from the particulate debris temperature for the cell. Lower head and penetration masses are calculated from composition and nodalization input specified on record COR_LH or specified directly on record COR_PEN. Heat transfer coefficients for debris and molten pool components (specified on record COR_LHF) may be

determined by parametric values or control function values, or, in the case of molten pools, may be calculated directly by the molten pool heat transfer models.

The heat transfer rate from the debris in the bottommost axial level to the lower head is given by

$$q_{d,h} = h_{d,h} A_h \left(T_d - T_{h,s} \right) \quad (6\text{-}1)$$

where

$q_{d,h}$ = heat transfer rate between debris and lower head (W),
$h_{d,h}$ = debris-lower head heat transfer coefficient (W/m^2-K),
A_h = lower head surface area (m^2),
T_d = debris temperature (K), and
$T_{h,s}$ = lower head inner surface temperature (K).

The heat transfer rate from the debris in the bottommost axial level to a penetration is similarly given by

$$q_{d,p} = h_{d,p} \frac{\Delta z_d}{\Delta z_1} A_p \left(T_d - T_p \right) \quad (6\text{-}2)$$

where

$q_{d,p}$ = heat transfer rate between debris and penetration (W),
$h_{d,p}$ = debris-penetration heat transfer coefficient (W/m^2-K),
A_p = penetration area (m^2),
Δz_d = debris height in the bottom axial level (m),
Δz_1 = bottom axial level height (m), and
T_p = penetration temperature (K).

The penetration area is based on the height of the bottom axial level, Δz_1, and the multiplier $(\Delta z_d - \Delta z_1)$ accounts for the partial covering of the penetration area by the debris of height Δz_d.

The heat transfer rate from the lower molten pool, MP1, in the axial level in contact with the lower head to a penetration is similarly given by

$$q_{m1,p} = h_{m1,p} \frac{\Delta z_{M1}}{\Delta z_1} A_p \left(T_{MP1} - T_p \right) \quad (6\text{-}3)$$

where

$q_{m1,p}$ = heat transfer rate between debris and penetration (W),

$h_{m1,p}$ = debris-penetration heat transfer coefficient (W/m^2-K), and

Δz_{M1} = height of lower molten pool in the bottom axial level (m).

The heat transfer rate from the upper molten pool, MP2, in the axial level in contact with the lower head to a penetration is similarly given by

$$q_{m2,p} = h_{m2,p} \frac{\Delta z_{M2}}{\Delta z_1} A_p \left(T_{MP2} - T_p \right) \quad (6\text{-}4)$$

where

$q_{m2,p}$ = heat transfer rate between debris and penetration (W),

$h_{m2,p}$ = debris-penetration heat transfer coefficient, and (W/m^2-K),

Δz_{M2} = height of upper molten pool in the bottom axial level (m).

The Stefan model calculates the transient heat conduction from the molten pool fields to debris ($q_{m1,d}$, $q_{m2,d}$) and from the molten pool fields to the lower head cells ($q_{m1,h}$, $q_{m2,h}$). In the case of the molten pools that are directly in contact with the lower head (no refrozen crust on the head surface), the heat loss from the molten pool is partitioned among the lower head nodes. In the case of a refrozen crust that exists between the molten pool and the lower head, a portion of the molten pool heat loss is transferred to the particulate debris field. In addition, the volume (thickness and area) of the calculated crust limits the inward radial relocation of solid conglomerate in order to retain sufficient heat capacity within the cell to model the refrozen crust (Section 3.2.5).

The heat transfer rate from the penetration to the top lower head node is based on the conduction area between the penetration and lower head specified by the user:

$$q_{p,h} = k_p A_{p,h} \frac{T_p - T_{h,s}}{\Delta z_1} \quad (6\text{-}5)$$

where

$q_{p,h}$ = heat transfer rate between penetration and lower head (W),

k_p = penetration thermal conductivity (W/m-K), and

$A_{p,h}$ = conduction area between penetration and head (m^2).

The conduction area $A_{p,h}$ should be chosen to appropriately model the two-dimensional nature of the heat conduction; note that conduction to only the top lower head node is modeled.

In MELCOR 1.8.6 and 2.1, a lateral conduction calculation is first performed and the heat transfer to or from each node used to determine a heat source that is then implemented in the implicit through-wall heat transfer calculation. Even though both the lateral and the through-wall calculations are implicit, the two calculations are essentially independent resulting in a 'semi-implicit' conduction calculation.

Transverse and normal bounding surface areas are calculated for each node in the vessel wall, as are transverse and through-wall conduction path lengths and vessel volumes. First, the material properties and conduction path lengths are used to construct the implicit matrix system of equations, which forms the discrete representation of the heat conduction equations in the transverse direction (along a layer of the vessel wall). The equations for heat transfer in the transverse direction are solved to obtain a new vector of temperatures. These temperatures are then used to calculate the net heat transfer rate due to lateral conduction into each mesh. Finally, a one-dimensional, through-wall heat conduction calculation is performed using the net heat transfer from the transverse calculation as an internal heat source.

Conduction heat transfer rates within the lower head are given by

Transverse:

$$q_{(i,j)\rightarrow(i,j+1)} = \frac{1}{\frac{\tau_j}{k_{i,j}} + \frac{\tau_{j+1}}{k_{i,j+1}}} A_{transverse,j} \left(T_{h,(i,j)} - T_{h,(i,j+1)}\right) \quad (6\text{-}6)$$

Through-wall:

$$q_{(i,j)\rightarrow(i+1,j)} = k_i \; FAC \; A_h \; \frac{T_{h,(i,j)} - T_{h,(i+1,j)}}{\Delta z_i} \quad (6\text{-}7)$$

where

$q_{(i,j)\rightarrow(i+1,j)}$ = heat transfer rate from node (*i,j)* to node (*i*+1,j),

$k_{i,j}$ = thermal conductivity of node (*i,j)*,

$T_{h,(i,j)}$ = temperature of lower head node (*i,j)*,

Δz_i = width of mesh layer *i*, and

$\tau_{i,j}$ = transverse path length from center of node (i,j) to boundary,

and FAC is a factor to enhance conduction through material that exceeds the melting point, given by Equation (2-32), in Section 2.2. The use of a planar finite-difference

equation to model heat transfer in hemispherical geometry is an adequate approximation because the thickness of the lower head is much smaller than the radius.

Convection heat transfer rates from the penetrations, debris, and inner surface of the lower head to the fluids in the lower-plenum control volume ICVHC (specified on record COR_RBV), $q_{p,v}$, $q_{d,v}$, and $q_{h,v}$ respectively, are modeled by the methods described in Section 2.3.

In MELCOR 1.8.6, the outer boundary of the lower head can transfer heat to multiple volumes that make up the reactor cavity. Heat transfer from the outer boundary of each lower head segment to the reactor cavity control volumes specified on the COR_LH records is partitioned between the atmosphere and the pool in each control volume, based upon the pool fraction of the surface area of the lower head in each segment as follows:

$$q_{h,c} = h_{ATM}(1 - F_{PL})A_h(T_{h,1} - T_{ATM}) + h_{rlx,PL}F_{PL}A_h(T_{h,1} - T_{SAT}) \quad (6\text{-}8)$$

where

- h_{ATM} = heat transfer coefficient from lower head to reactor cavity atmosphere (W/m^2-K),
- $h_{rlx,PL}$ = relaxed heat transfer coefficient from lower head to reactor cavity pool (W/m^2-K),
- F_{PL} = pool fraction of surface area A_h,
- T_{ATM} = temperature of reactor cavity atmosphere obtained from CVH (K),
- T_{SAT} = saturation temperature of reactor cavity pool obtained from CVH (K), and
- $T_{h,1}$ = lower head outer surface temperature at the beginning of the timestep (K).

The first term on the righthand side of Equation (6-8) accounts for heat transfer to the reactor cavity atmosphere, while the second term accounts for heat transfer to the reactor cavity pool. The pool fraction, F_{PL}, is simply the fraction of the area that is immersed in the pool, based upon the depth of the pool obtained from the CVH database at the beginning of each timestep. The heat transfer coefficient to the reactor cavity atmosphere, h_{ATM}, is implemented as sensitivity coefficient C1246(1), with a default value of 10 W/m^2-K. The unrelaxed heat transfer coefficient to the reactor cavity pool, h_{PL}, is calculated using a simple downward-facing saturated pool boiling model. Relaxation of h_{PL} is implemented exactly as discussed in Section 2.3. Heat transfer to the pool before boiling is currently ignored, as is subcooling of the pool; it is calculated only when the temperature of the outer surface of the lower head exceeds the saturation temperature in the reactor cavity. Hence, the second term on the righthand side of Equation (6-8) cannot be negative.

The downward-facing saturated pool boiling model treats three heat transfer regimes:

(1). fully-developed nucleate boiling with no dependence on the orientation of the boiling surface;

(2). transition boiling between the fully developed and film boiling regimes, in which the heat flux is obtained by logarithmic interpolation between the critical heat flux and the minimum heat flux, based upon the temperature difference between the surface and saturation; and

(3). stable film boiling, which depends upon the orientation of the boiling surface.

The boundaries between the heat transfer regimes are determined by a correlation for the critical heat flux, which separates fully developed and transition boiling, and a correlation for the minimum-stable-film-boiling heat flux, which separates transition and stable film boiling. Although heat transfer in the nucleate boiling regime is assumed to be independent of the orientation of the surface, the critical heat flux, which determines its upper limit, is dependent on surface orientation and is given by [61]

$$q_{CHF}(\theta) = (0.034 + 0.0037\theta^{0.656})\ \rho_v^{1/2} h_{lv}[g\ \sigma(\rho_l - \rho_v)]^{1/4} \quad (6\text{-}9)$$

where

θ = inclination angle of the surface in degrees ($\theta = 0^\circ$ for a downward-facing surface);

ρ_l, ρ_v = densities of water and steam, respectively (kg/m^3);

g = acceleration of gravity (m/s^2);

σ = interfacial surface tension between steam and water (N/m); and

h_{lv} = latent heat of vaporization of water (J/kg),

and the constants 0.034, 0.0037 and 0.656 have been implemented in sensitivity coefficient array C1245 in the HS package. Similarly, the minimum-stable-film-boiling heat flux, which separates transition boiling from stable film boiling, is given as a function of θ as [61]

$$q_{MIN}(\theta) = (4.8 \cdot 10^{-4} + 8.2 \cdot 10^{-4}\theta^{0.407})\ \rho_v^{1/2} h_{lv}[g\sigma\ (\rho_l - \rho_v)]^{1/4} \quad (6\text{-}10)$$

where the constants 4.8 x 10^{-4}, 8.2 x 10^{-4}, and 0.407 have also been implemented in sensitivity coefficient array C1245 in HS package.

In the nucleate boiling regime, the heat flux, as a function of the difference between the surface temperature and the saturation temperature, $\Delta T \equiv T_{SRF} - T_{SAT}$, is given by

$$q_{NB}(\Delta T) = h_{NB}\,\Delta T \tag{6-11}$$

where h_{NB} is given as a function of ΔT (and pressure) by Equation (2-109). In the stable film boiling regime, the heat flux as a function of ΔT is given by

$$q_{FLM}(\Delta T) = h_{FLM}\,\Delta T \tag{6-12}$$

and the user has two options for determining the h_{FLM} as a function of ΔT. The default option gives the heat transfer coefficient as [62]

$$h_{FLM}(\Delta T) = 0.142\,k_v\left[\frac{h_{lv}\,\rho_v\,g(\rho_l-\rho_v)}{\mu_v k_v\,\Delta T}\right]^{1/3}(\sin\theta)^{0.3333333} \tag{6-13}$$

where the constants 0.142 and 0.3333333 have been implemented in sensitivity coefficients array C1245 in HS package. The other option, which is invoked when the user changes the value of sensitivity coefficient C1245(7) to 1.0, gives the heat transfer coefficient as [63]

$$h_{FLM}(\Delta T) = \left(0.055 + 0.016\,\theta^{0.5}\right)k_v\left[\frac{h_{lv}\,\rho_v\,g(\rho_l-\rho_v)}{\mu_v k_v\,\Delta T}\right]^{1/3} \tag{6-14}$$

where the constants 0.055, 0.016, and 0.5 have been implemented in sensitivity coefficient array C1245 in HS package.

Equations (6-9) to (6-14) give values of heat fluxes and heat transfer coefficients for particular values of θ (except for Equation (6-11), which is independent of θ). To obtain an average value, f_i, of function f(θ), which is appropriate for segment *i* in the lower head model, f(θ) is averaged over the wetted surface area of segment *i* as follows:

$$f_i = \frac{\int_{\theta_{i-1}}^{\theta_u} f(\theta)\,\sin\theta\,d\theta}{\cos\theta_{i-1} - \cos\theta_u} \tag{6-15}$$

$$\theta_u = min[\theta_i, max(\theta_{i-1}, \theta_{PL})] \tag{6-16}$$

where θ_{PL} is the angle from the bottom of the lower head to the pool surface, $\theta_0 = 0°$ and $\theta_{NRAD} = 90°$. This averaging results in positive values for all quantities in Equations (6-9) to (6-14), even though the heat transfer coefficient in Equation (6-13) is zero at $\theta = 0°$. Note, that for MELCOR 1.8.5 and earlier versions, specification of very small values of θ_i, (i.e., defining a very small innermost segment) was discouraged because the lower head

model did not include azimuthal conduction, which tended to limit the formation of local hot spots where boiling heat removal is low (at the very bottom of the lower head). Because MELCOR 1.8.6 and 2.1 does model azimuthal heat conduction in the vessel, this problem is less likely.

If the heat flux from Equation (6-11) is set equal to the average heat flux from Equation (6-9) for any segment and is solved for ΔT, the result is equal to ΔT_{CHF}, the temperature difference at the critical heat flux for that segment. If the average heat flux from Equation (6-12) is set equal to the average heat flux from Equation (6-10) and is solved for ΔT, the result is equal to ΔT_{MIN}, the temperature difference at the minimum-stable-film-boiling heat flux for that segment. Because the actual value of ΔT is known from the database at the beginning of each COR timestep, it can be compared to ΔT_{CHF} and ΔT_{MIN} to determine the appropriate heat transfer regime. If the value is less than ΔT_{CHF}, then fully developed nucleate boiling occurs and h_{PL} is given by h_{NB} from Equation (2-232). If ΔT is greater than ΔT_{MIN}, then stable film boiling occurs and h_{PL} is given by Equation (6-13) or (6-14), as specified by the user. If the value of ΔT lies between ΔT_{CHF} and ΔT_{MIN}, then h_{PL} is equal to the transition boiling heat transfer coefficient, which is found by logarithmic interpolation as follows:

$$h_{TRN} = \frac{q_{MIN}}{\Delta T}\left(\frac{\Delta T}{\Delta T_{MIN}}\right)^{\left[\frac{log(q_{CHF}/q_{MIN})}{log(\Delta T_{CHF}/\Delta T_{MIN})}\right]} \tag{6-17}$$

The net energy transfer for each of the model elements is given by the following equations:

$$C_{p,p}\left(T_p^n - T_p^o\right) = \left(q_{d,p} - q_{p,h} - q_{p,v}\right)\Delta t \tag{6-18}$$

$$C_{p,d}\left(T_d^n - T_d^o\right) = \left(q_s - q_{d,p} - q_{d,h} - q_{d,v} - q_{d,d}\right)\Delta t \tag{6-19}$$

$$C_{p,h,n}\left(T_{h,n}^n - T_{h,n}^o\right) = \left(q_{n-1,n} + q_{d,h} + q_{p,h} - q_{h,v}\right)\Delta t \tag{6-20}$$

$$C_{p,h,i}\left(T_{h,i}^n - T_{h,i}^o\right) = \left(q_{i-1,i} - q_{i,i+1}\right)\Delta t \tag{6-21}$$

$$C_{p,h,1}\left(T_{h,1}^n - T_{h,1}^o\right) = \left(-q_{d,c} - q_{1,2}\right)\Delta t \tag{6-22}$$

where many of these variables were defined in Section 6.1 and

$C_{p,j}$ = heat capacity of model element j times the mass of element j, ($M_j c_{p,j}$),

q_s = debris heat source from oxidation and decay heat (W),

$q_{d,d}$ = debris cell-to-cell heat transfer rate (W), and

Δt = COR package timestep (s),

and superscripts *o* and *n* refer to old-time and new-time temperatures, respectively. All temperatures in Equations (6-1) through (6-6) are considered to be new-time temperatures, and Equations (6-8) through (6-11) are solved implicitly for new-time temperatures by matrix inversion.

6.2 Lower Head Melting

When internal nodes of the lower head reach the melting temperature for carbon steel, that material is available to relocate with other molten debris in the lower head. In addition, heat conduction for those nodes is changed as the conduction path is reduced, resulting in an increase in the heat flux through the vessel wall. Furthermore, melted nodes are no longer able to support the mechanical load on the vessel.

As the enthalpy of a node reaches the melt enthalpy, the fraction of the node mass that has melted is calculated for each material node (using temperatures from associated temperature nodes) as follows:

$$f_{i,left}^{n} = \max\left(f_{i,left}^{n-1}, \left(ELH_{i-1} - E_{mlt}\right)/E_{LHF}\right)$$

$$f_{i,right}^{n} = \max\left(f_{i,right}^{n-1}, \left(ELH_{i} - E_{mlt}\right)/E_{LHF}\right)$$

Where

I is an index on the material node and the fraction of melted material is calculated using temperatures from the associated temperature nodes (material nodes shown as solid colors and temperature nodes shown in dashed rectangles in Figure 6.2).

n is an index on time. The fraction melted is non-decreasing (freezing does not restore the lower head node but creates a crust material modeled as conglomerate particulate debris) and the fraction melted is initialized at 0.0.

ELH_i is the enthalpy of node i of the lower head.

E_{mlt} is the enthalpy at the solidus.

E_{LHF} is the latent heat of fusion for the lower head material.

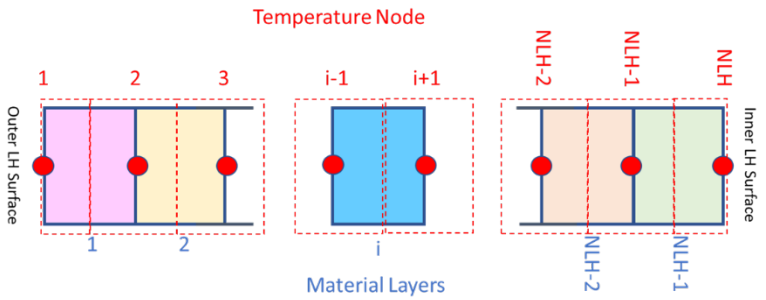


Figure 6.2 Through-wall nodalization diagram.

The fraction of the mass node that is melted is then given by $f_{j,i}^{n} = 0.5 * \left(f_{i,left}^{n} + f_{i,right}^{n}\right)$ which is multiplied by the initial node mass ($mass_{LH,i}$) to determine the melted mass that is sourced into the COR package with a specific enthalpy of $E_{mlt} + E_{LHF}$. In addition, the density of the receiving material (currently only stainless steel can receive the melt) is used to increase the virtual volume of the COR cell that is in contact with the inner node, $\Delta V = f_{j,i}^{n} \cdot mass_{LH,i}/\rho_{SS}$.

The through-wall conduction equation (6-2) is modified by adjusting the conduction path length, Δz_i and the reduced heat capacity of the melting nodes and the melt enthalpy are also incorporated into equations (6-18) to (6-22) to account for this loss of mass in the lower head structure.

As a node becomes completely melted, the boundary node shifts inwards and all inner boundary heat terms are applied at this new boundary node. These melted nodes are also considered in the mechanical model described in the following section.

6.3 Failure

Failure of the lower head occurs if any of four criteria is met:

(1). the temperature of a penetration (or the temperature of the innermost node of the lower head) reaches a failure temperature (TPFAIL) specified by the user on record COR_LHF,

(2). a failure logical control function (specified by the user on record COR_RP) is found to be true (for example, such a control function might refer to a table of differential failure pressures as a function of lower head temperature),

(3). overpressure from the falling-debris quench model occurs (see Section 2.3.7),and the lower head is allowed to fail from overpressure, with a

default failure criterion of 20 MPa that may be changed on input record COR_LP, or

(4). creep-rupture failure of a lower head segment occurs, in response to mechanical loading under conditions of material weakening at elevated temperatures.

The creep-rupture failure model uses the temperature profile through the lower head to calculate creep based on a Larson-Miller parameter and a life-fraction rule whenever the effective differential pressure across the lower head exceeds a user-specified minimum value (implemented as sensitivity coefficient C1600(3) with a default value of 1000 Pa). The effective differential pressure is the sum of the actual differential pressure between the lower plenum and the reactor cavity and the pressure caused by the weight of any debris resting on the lower head. The lower limit on the effective pressure differential was imposed to bypass the model and save computational resources when the threat of creep-rupture is minuscule.

The model is applied to the load-bearing mesh layers in the lower head, which include all NLH-1 mesh layers by default. However, by entering a positive value for NINSLH on input record COR_LH, the user defines the outer NINSLH layers to consist of non-load-bearing insulation. An optional one-dimensional mechanical model that calculates the thermal and plastic strain in each load-bearing mesh layer may be invoked by setting the value of sensitivity coefficient C1600(1) equal to 1.0. By default, however, a zero-dimensional model based on the mass-averaged temperatures in the load-bearing mesh layers is used with the effective membrane stress induced by the effective differential pressure to calculate a single Larson-Miller parameter for each radial ring.

The Larson-Miller creep-rupture failure model [59] gives the time to rupture, t_R, in seconds, as

$$t_R = 10^{\left(\frac{P_{LM}}{T} - 7.042\right)} \tag{6-23}$$

where P_{LM} is the Larson-Miller parameter given by

$$P_{LM} = 4.812x10^4 - 4.725x10^3 \quad log_{10} \quad \sigma_e \tag{6-24}$$

where σ_e is the effective stress in Pa, and the constants 4.812 x 10^4, –4.725 x 10^3, and 7.042 (Equation (6-23)), which are appropriate for SA533B1 vessel steel [64], have been implemented in sensitivity coefficient array C1601. The life-fraction rule gives the cumulative damage, expressed as plastic strain, $\varepsilon_{pl}(t)$, as

$$\varepsilon_{pl}(t + \Delta t) = \varepsilon_{pl}(t) + 0.18 \frac{\Delta t}{t_R} \quad (6\text{-}25)$$

where the constant 0.18, which has been implemented as sensitivity coefficient C1601(4), implies that failure (defined as when the integrated value of $\Delta t / t_R$ reaches unity) occurs when the strain reaches 18% [63].

For the zero-dimensional default option, the effective stress is given by

$$\sigma_e = \frac{(\Delta P + \rho_d g \Delta z_d) R_i^2}{R_o^2 - R_i^2} \quad (6\text{-}26)$$

where ΔP is the pressure difference across the lower head, ρ_d and Δz_d are the density and depth of the debris resting on the lower head, and R_i and R_o are the inner vessel radius and outer radius of load-bearing vessel steel, respectively. Substitution of σ_e from Equation (6-26) into Equation (6-24) yields a value of P_{LM} for each ring. Substitution of the temperature, mass-averaged over all the load-bearing mesh layers in each lower head ring, and the value of P_{LM} into Equation (6-23) yields t_R (the predicted time lapse until failure for a specimen subjected to the current temperature and stress). And, finally, substitution of t_R into Equation (6-25) yields the accumulated plastic strain at each timestep. Failure is declared when $\varepsilon_{pl}(t)$ reaches failure strain, given by sensitivity coefficient C1601(4), with a default of 0.18, and the mechanical calculation in that ring ceases.

The optional one-dimensional mechanical model predicts the stress-strain distribution through the lower head and treats stress redistribution from both thermal strain and material property degradation. The elastic modulus as a function of temperature is given by [63]

$$E(T) = 2.0x10^{11}\left[\frac{1}{1+\left(\frac{T}{900.}\right)^{6.}} - \frac{1}{1+\left(\frac{1800.}{900.}\right)^{6.}}\right] \quad (6\text{-}27)$$

where the constants 2.0×10^{11}, 1800., 900., and 6., which are appropriate for reactor vessel steel, have been implemented as sensitivity coefficient array C1602. The yield stress as a function of temperature is given by [63]

$$\sigma_Y(T) = 4.0x10^8 \left[\frac{1}{1+\left(\frac{T}{900.}\right)^{6.}} - \frac{1}{1+\left(\frac{1800.}{900.}\right)^{6.}} \right] \quad (6\text{-}28)$$

where the constants 4.0 x 10^8, 1800., 900. and 6., which are appropriate for reactor vessel steel, have been implemented as sensitivity coefficient array C1603.

The one-dimensional model requires that the stress distribution integrated over the vessel thickness be equal to the imposed load:

$$[\Delta P + \rho_d g \Delta z_d] R_0^2 = \sum_i^{N_{NY}} \sigma_i \left(R_i^2 - R_{i-1}^2 \right) + \sum_j^{N_Y} \sigma_Y(T_j) \left(R_j^2 - R_{j-1}^2 \right) \quad (6\text{-}29)$$

where the first sum on the righthand side is over all layers that have not yielded, N_{NY}, or melted and the second sum is over all layers that have yielded, N_Y, but have not melted.

The stress, σ_i in layers that have not yielded is given by

$$\sigma_i = E(T_i) \left[\varepsilon_{tot} - (\varepsilon_{pl,i} + \varepsilon_{th,i}) \right] \quad (6\text{-}30)$$

where $E(T_i)$ is the value of the elastic modulus at the average temperature in mesh layer *i*, which is equal to the average of the node temperatures on the two boundaries, ε_{tot} is the total strain across the lower head for that particular segment, which is the same for mesh layers in that segment, and $\varepsilon_{pl,i}$ and $\varepsilon_{th,i}$ are the plastic and thermal strains, respectively, in mesh layer *i*. The thermal strain is given by

$$\varepsilon_{th,i} = 1.0 \times 10^{-5} (T_i - T_{ref}) \quad (6\text{-}31)$$

where the constant 1.0 x 10^{-5} is the linear thermal expansivity, which has been implemented as sensitivity coefficient C1600(2), and T_{ref} is the reference temperature, which is equal to the initial temperature specified by the user for that ring of the lower head. Equations (6-29) and (6-30) are solved implicitly and iteratively for ε_{tot}, σ_i, and $\varepsilon_{pl,i}$ ($\varepsilon_{th,i}$ is known because the temperature profile is known) using Equations (6-23) – (6-25) to update the plastic strain profile with the latest stress profile after each iteration. Failure is declared when ε_{tot} reaches 18% (the use of ε_{tot} rather than ε_{pl} makes little

difference because the elastic and thermal strains are insignificant compared to the plastic strain when ε_{tot} becomes large).

Whenever any failure condition is satisfied, an opening with an initial diameter defined by the user on record COR_PEN or with an initial diameter of 0.1 m is established. If there are no penetrations (this gives a relatively rapid ejection of debris without numerical difficulties), the COR package control function argument COR-ABRCH (see Section 4 of the COR Package Users' Guide) is set to the initial failure flow area calculated from this diameter. COR-ABRCH can then be used to open a valve in the flow path from the lower plenum control volume to the reactor cavity control volume. COR-ABRCH may be increased by additional penetration failures (up to three per radial ring) or by ablation of the failure openings, as described in the next section.

6.3.1 User Flexibility in Modeling

As noted in preceding subsections, the coefficients in the equations that relate stress to load for the various models can be modified through user input. This capability can be used to model variations in the form of the structure. For example, if a plate is supported on a square grid of columns of radius *c* and spacing *x*, the maximum bending moment per unit length is at the support. The value is given by Equation (5-2) with a modulus [65]

$$K_{support} = \frac{(1+\nu)\ [ln(x/c) - 0.811]}{4\pi} \tag{6-32}$$

A value can be computed from this equation (for example, for $\nu = 0.3$ and (*x/c*) = 5, $K_{support}$ = 0.0826) and used in the PLATEG model to represent a column-supported plate.

MELPROG [66] used this expression for a column-supported plate and accounted for the effects of holes in the plate by dividing *K* by a ligament efficiency, ε. In the case of four holes per cell of radius *b* and spacing *d*, this is given by

$$\varepsilon = 1 - b/d \tag{6-33}$$

In fact, the results of a full structural analysis (outside of MELCOR) of a more complicated structure could be used to calculate an effective *K* for use in one of the plate models. Thus, the relatively simple models could be used to represent quite complicated support structures, should a user so desire.

6.4 Debris Ejection

After a lower head failure has occurred, the mass of each material in the bottom axial level that is available for ejection (but not necessarily ejected) is calculated. Two simple options exist. In the default option (IDEJ = 0 on record COR_TST), the masses of each material available for ejection are the total debris and molten pool material masses,

regardless of whether or how much they are molten. Note, however, that this option has been observed to lead to ejection of much more solid debris with the melt than is realistic.

In the second option (IDEJ = 1 on MELCOR record COR_TST), the masses of steel, Zircaloy, and UO2 available for ejection are simply the masses of these materials that are molten. The masses of steel oxide and control poison materials available for ejection are the masses of each of these materials multiplied by the stainless-steel (the group lead) melt fraction, based on an assumption of proportional mixing; this is the stainless steel ejection group. Similarly, the mass of ZrO2 available for ejection is the ZrO2 mass multiplied by the Zircaloy melt fraction. Additionally, the mass of solid UO2 available for ejection is the Zircaloy melt fraction multiplied by the mass of UO2 that could be relocated with the Zircaloy as calculated by the secondary material transport model (see Section 3.1). These materials form the Zircaloy ejection group. User-defined COR materials can be assigned to a particular ejection group or a user-defined ejection group. A summary of the various ejection groups is provided in Table 6-1. A fourth ejection group for materials involved in the eutectics model has not yet been developed. Materials in an "EXEMPT" ejection group cannot be ejected but a more appropriate ejection group can be assigned using the MP_BHVR card.

Table 6-1 Ejection Groups for MELCOR Database Materials

COR Material Index	Material	Ejection Group
8	ZIRCALOY	EJ-ZIRCALOY
9	ZIRCONIUM-DIOXIDE	EJ-ZIRCALOY
10	URANIUM-DIOXIDE	EJ-ZIRCALOY
11	STAINLESS-STEEL	EJ-STAINLESS-STEEL
12	STAINLESS-STEEL-OXIDE	EJ-STAINLESS-STEEL
13	BORON-CARBIDE	EJ-STAINLESS-STEEL
14	SIC	EJ-STAINLESS-STEEL
15	EUTECTIC	EJ-EUTECTIC
16	URANIUM-METAL	EXEMPT
17	GRAPHITE	EXEMPT
29	URANIUM-HEXAFLUORIDE	EXEMPT
30	ALUMINUM	EXEMPT
31	ALUMINUM-OXIDE	EXEMPT
32	CADMIUM	EXEMPT
33	STAINLESS-STEEL-304	EJ-STAINLESS-STEEL
34	LITHIUM-ALUMINUM	EXEMPT
35	URANIUM-ALUMINUM	EXEMPT

COR Material Index	Material	Ejection Group
36	CARBON-STEEL	EJ-STAINLESS-STEEL
37	B4C-INT	EJ-STAINLESS-STEEL
38	ZRO2-INT	EJ-ZIRCALOY
39	UO2-INT	EXEMPT
All other indices	All other materials	EXEMPT

Materials in blue are “Ejection Group Leads”.

Regardless of which of the options described above is chosen, other constraints have been imposed on the mass to be ejected at vessel failure. A total molten mass of at least 5000 kg C1610(2) or a melt fraction (C1610(1)) of 0.1 (total molten mass divided by total debris mass) is necessary before debris ejection can begin to avoid calculational difficulties with the core-concrete interactions modeling. Also, whenever the bottom lower head node exceeds the penetration failure temperature, TPFAIL, gross failure of the lower head in that ring is assumed, and all debris in the bottom cell is discharged linearly over a 1 s timestep, regardless of the failure opening diameter. However, no mass associated with either the lower head hemisphere or the penetrations is added to the core/lower plenum debris.

After the total mass of all materials available for ejection has been determined, the fraction of this mass ejected during a single COR package subcycle is determined from hydrodynamic considerations. The velocity of material being ejected is calculated from the pressure difference between the lower-plenum control volume and the reactor cavity control volume, the gravitational head from the debris layer itself, and a user-specified flow discharge coefficient input on record COR_LHF, using the Bernoulli equation:

$$v_{ej} = C_d \left(\frac{2\,\Delta P}{\rho_m} + 2\ g\ \Delta z_d \right)^{1/2} \qquad (6\text{-}34)$$

where

v_{ej} = velocity of ejected material,

C_d = flow discharge coefficient,

ΔP = pressure difference between lower plenum control volume and reactor cavity control volume,

ρ_m = density of material being ejected,

g = gravitational acceleration, and

Δz_d = debris and molten pool height.

If the expression in parentheses in Equation (6-34) is negative, the ejection velocity is set to zero.

The maximum mass of all materials that can be ejected during a single COR timestep is

$$M_{ej} = \rho_m \; A_f \; v_{ej} \; \Delta t \tag{6-35}$$

where

M_{ej} = maximum mass ejected,

A_f = penetration failure area, and

Δt = timestep.

The fraction of the total mass available for ejection that actually is ejected during the subcycle is simply M_{ej} divided by the total mass available to be ejected, up to a maximum value of 1.0. This fraction is applied to the mass of each material available for ejection.

Mass and energy that are ejected from the COR package via the foregoing model are transferred to the Transfer Processes (TP) package. That package is a generalized interface utility for mass and energy transfers of core materials between packages and within the radionuclide (RN) package, and performs various bookkeeping functions related to different equation-of-state and mass-species representations between packages. The CAV, fuel dispersal interactions (FDI), and RN packages may all call the TP package to transfer core materials into their domain. The “IN” Transfer Process number that specifies the TP package input for transferring masses and energies from the COR package must be specified on record COR_TP.

Ablation of the failure opening is modeled by calculating the heat transfer to the lower head by flowing molten debris. A simplified implementation of the ablation model by Pilch and Tarbell [67] is used, which gives the heat transfer coefficient for the flowing molten debris as the maximum of a tube correlation and a flat plate correlation:

$$h_{abl,tube} = 0.023 \; K_p \; v_{ej}^{0.8} \; / \; D_f^{0.2} \tag{6-36}$$

$$h_{abl,plate} = 0.0292 \; K_p \; v_{ej}^{0.8} \; / \; \Delta z_h^{0.2} \tag{6-37}$$

where

h_{abl} = ablation heat transfer coefficient,

K_p = $k(\rho/\mu)^{0.8}\ Pr^{1/3}$ (using average property values from [38]),

D_f = failure diameter, and

Δz_h = lower head thickness.

The ablation rate is then calculated as

$$\frac{dD_f}{dt} = \frac{2\ h_{abl}\ (T_d - T_{m,s})}{\rho_s\ [c_{p,s}\ (T_{m,s} - T_{h,avg}) + h_{f,s}]} \quad (6\text{-}38)$$

where ρ_s, $c_{p,s}$, $h_{f,s}$, and $T_{m,s}$ are the density, heat capacity, latent heat of fusion, and melting temperature of the (lower head) steel, respectively, and T_d and $T_{h,avg}$ are the debris and average lower head temperatures, respectively. The diameter of the penetration failure is updated explicitly with time using Equation (6-38). The value of the control function argument COR-ABRCH is then redefined to reflect the new failure-opening diameter.

7. Reactor Point Kinetics Model

A point kinetics model was added to allow MELCOR calculation of accident sequences without SCRAM. The model includes: (a) An active neutron source for zero-power reactor startup; and (b) doppler and fuel and graphite temperature reactivity feedbacks. The point kinetics model uses an extremely efficient exponential matrix technique with a discretization error on the order of $(\Delta t)^3$, in which the exponential matrix is approximated using the 7th order-accurate Padé(3,3) function. The present model is capable of modeling high reactivity insertion cases, i.e. $\rho/\overline{\beta}$ > \$1.0. Other desirable characteristics of the present model include unconditional stability, and the freedom to use a time step size that is less restrictive than that used by the MELCOR thermal-hydraulics package. The values of the default temperature feedback coefficients are obtained by a least-squared fit of the preliminary neutronics calculations performed by INL for the NGNP prismatic reactor [68].

The developed 6 delayed group kinetics model was successfully benchmarked using the Inhour solution for step reactivity insertions (both positive and negative). The model was also tested using a HTGR core model at steady state with a \$0.50 step reactivity insertion. Results were consistent with those expected from an analytic model.

7.1 Model Equations

The equations solved are the 6 delayed group point kinetics equations [69]

$$\frac{dn}{dt} = \frac{\rho - \beta}{\Lambda} n + \sum_{i=1}^{6} \lambda_i C_i + S_0 \quad (7\text{-}1)$$

$$\frac{dC_i}{dt} = \frac{\beta_i}{\Lambda} n - \lambda_i C_i \quad (7\text{-}2)$$

where

n = prompt neutron power [W]

ρ = reactivity

β = total delayed neutron fraction

Λ = prompt neutron generation time [s]

λ_i = decay constant of i-th precursor group [1/s]

C_i = power of i-th delayed neutron precursor group [W]

β_i = fraction of i-th delayed neutron group

S_0 = initial neutron source [W/s]

The default decay constants, delayed neutron fractions, and neutron generation time are those for a U^{235} thermal spectrum reactor [70]. These are implemented as C1405, Point Kinetics Nuclear Data.

The reactivity can be expressed as

$$\rho = \rho_{ext} + \rho_D + \rho_f + \rho_G \quad (7\text{-}3)$$

where

ρ_{ext} = external reactivity insertion

ρ_D = Doppler feedback

ρ_f = fuel thermal density feedback

ρ_G = graphite thermal density feedback

The external reactivity can be input by the user as either a table or control function.

The feedback reactivities are represented as functions of the average fuel and moderator temperatures referenced to the average temperatures at reactor steady-state (i.e., where $\rho = 0$). The fuel Doppler feedback is represented as a logarithmic function of temperature

$$\rho_D = \chi_D \, ln(\bar{T}_f / \bar{T}_f^0) \tag{7-4}$$

where

χ_D = Doppler coefficient [1/s]

$\bar{T}_f$ = average fuel temperature [K]

$\bar{T}_f^0$ = average fuel temperature at steady state reference [K]

The fuel density feedback reactivity is represented as a 2nd order polynomial in the average fuel temperature, and the graphite thermal feedback reactivity as a 4th order polynomial in the average graphite temperature.

7.2 Solution Method

The solution method is an exponential matrix method using Pade approximations, based on work by Porsching.[71] This method is unconditionally stable for timesteps much larger than those typically used in a MELCOR problem. For details, see Reference [72].

8. Heat Pipe Modeling

Heat pipes (HPs) are self-contained devices that use the large latent heat associated with phase transition together with the very high heat transfer rates associated with boiling and condensation to enable remarkably efficient heat transfer. Several recent reactor development efforts have included HPs for heat removal from the core, effectively replacing the functionality of the primary cooling loop that typically exists in a more traditional reactor design.

To enable modeling such reactors, a generalized approach has been developed and implemented for modeling HPs in MELCOR. Internally, the approach defines software interfaces to MELCOR packages that are independent of the details of the HP internal model so that models of different fidelity and applicability can be written and made available as needs arise. Use of a HP model replaces the conventional convective heat transfer between the fuel and coolant channel with the energy transfer from the fuel to the evaporative region of the HP. Heat rejection from the HP model at the condensation interface is then transferred to the CVH package.

Two relatively simple but functional HP models that meet the interface requirements have been written and used to test the approach and the associated interfaces with other MELCOR models and packages. Model “1” treats the working fluid region as an extremely high thermal conductivity material. Model “2” treats the liquid and vapor phases in the working fluid region (currently either sodium or potassium) as being in thermodynamic equilibrium and applies analytical expressions or user-input tabular data to capture the

effects of the capillary, boiling, and sonic limitations. Both current models apply steady-state approximations that limit their applicability to accurately capture detailed transient behavior. However, the interface defined accommodates the development and implementation into MELCOR of more detailed transient heat pipe models as future needs arise. The heat pipe modeling interfaces with the other COR package models to treat heat transfer between HP regions (lateral "heat paths"), HP degradation and failure (conversion to melt or rubble material), RN release and transport, etc.

8.1 Geometric Assumptions

Figure 8.1 illustrates key aspects and regions of a generic heat pipe and is useful as a reference for the descriptions that follow.

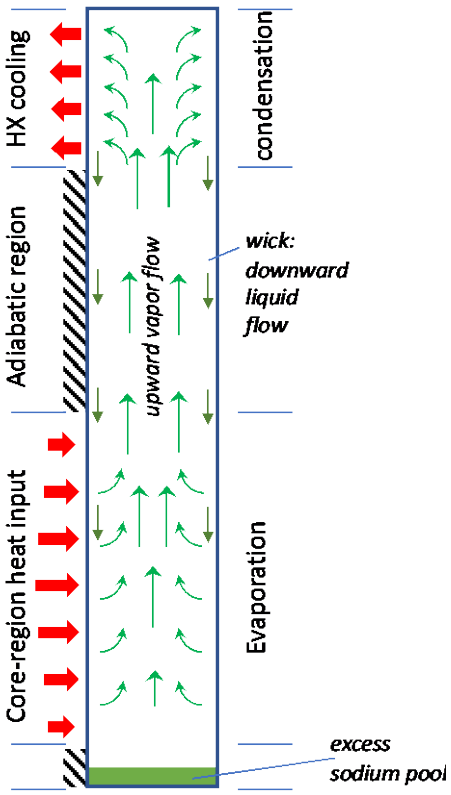


Figure 8.1 Illustration of a generic heat pipe

In MELCOR 2, the basic geometry of a heat pipe is assumed to be a circular cylinder characterized by a relatively small set of quantities. Required input quantities include

R_O outside radius of heat pipe wall (m),

R_I inside radius of heat pipe wall (m),

D_{wick} thickness (or depth) of the wick (m), and

ϕ_{wick} porosity of the wick (-).

Also required are the individual lengths of the three sections in a generic heat pipe, (1) the evaporator region, (2) the adiabatic region, and (3) the condenser region. The lengths of each of these sections are implicitly defined by the COR package cells that these regions are associated with. A new "COR_HP" input card (see MELCOR User's Manual) is where the association between COR cells and HP regions is defined. Also defined on this card is the number of actual heat pipes being represented by this single instantiation. This number acts as a multiplier for computing other important geometric quantities, such as volume, surface and cross-sectional areas, etc. that are derived and computed based on these primary geometric variables. Figure 8.2 illustrates the required geometry and its association with COR cells and CVH volumes.

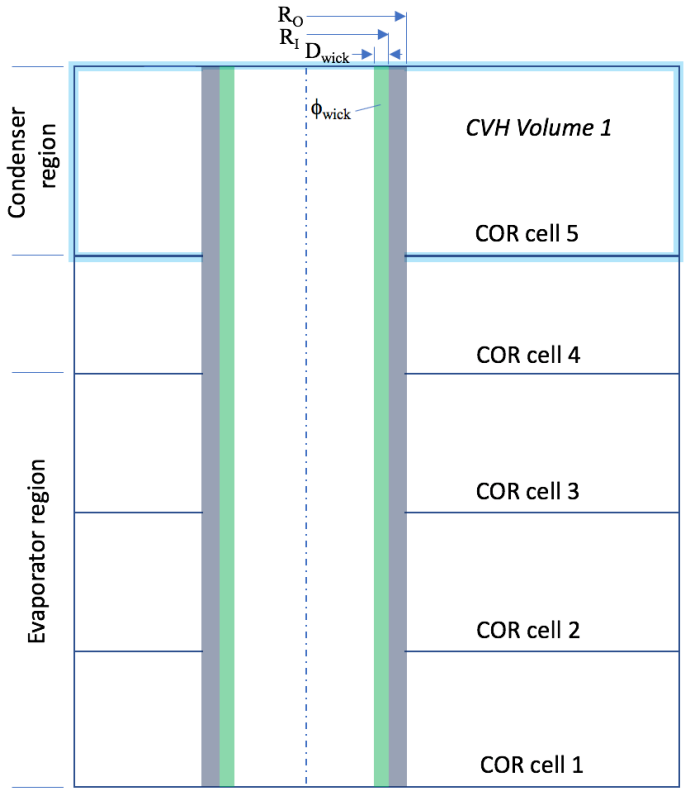


Figure 8.2 HP geometry and association with COR cells and CVH volumes

The HP modeling also requires that intact fuel (COR component "IFU") be present in all COR cells associated with the evaporator region. In Figure 8.2 these would be cells 1, 2 and 3. We currently assume that the fuel surrounds the heat pipe and is in direct contact with the outer surface of the heat pipe wall. This assumption is subject to change in the

future as other HP-based reactor designs may not conform to this. The current approach also assumes that any “cladding” (COR component "ICL") present in an evaporator region cell exists on the outside of the fuel (note that the modeling of the HP wall itself is contained within the separate HP internal model).

In any cell “*i*”, the heat transfer rate between the fuel and the surrounding metal clad is computed as described in Section 2.2.8 (Fuel Cladding Gap Heat Transfer). Applying this model, despite the geometric differences, is considered adequate for present purposes due to the minimal role that the cladding plays in the heat transfer from the fuel in the presence of a heat pipe.

The current modeling assumes that the coolant in the CVH volume associated with the condenser region COR cells is in direct contact with the outer surface of the heat pipe wall. In the adiabatic region, the model assumes no heat transfer from/to the HP outer wall.

8.2 Energy exchange with CVH and COR

When HP modeling is active, the COR package skips the normal models for convective heat transfer between COR components and the CVH coolant. The HP model replaces convective heat transfer to the coolant as the mechanism by which energy is transferred away from the fuel in the core region of the reactor.

The HP internal model interacts with other models in MELCOR through defined subroutine and data structure interfaces and details are model specific. Specifically,

- The COR package takes the current HP wall temperature and surface area and calls a designated HP internal model subroutine to compute a heat transfer rate from the fuel to the HP. This heat transfer rate is then passed back into the HP model as a boundary condition.

- The HP Model takes a current coolant temperature and solves for the heat transfer from the condenser region. This is passed back to the CVH control volume as an energy source into the coolant for a given time step.

- Various geometric (e.g. volumes) and connectivity data (e.g. what COR cells and what CVH volumes the HP interfaces with) must be available.

8.3 Modeling Requirements of the HP internal model

The HP internal model interfaces with the COR and CVH packages by calculating the energy exchange rates at the heat pipe outer wall with its surroundings. For cell “i” in the evaporator region, Q_i^n, is computed using beginning-of-COR-package time step values, and thus becomes an explicit boundary condition for the HP internal model equations. It takes the form:

$$Q_i^n = h_{eff,i} A_{hpw,i} (T_{fu,i}^n - T_{hpw,i}^n) \tag{8-1}$$

where the temperature of the fuel in COR cell i, $T_{fu,i}^n$, is provided by the COR package and the HP internal model provides

$h_{eff,i}$ an "effective" heat transfer coefficient in cell i,

$A_{hpw,i}$ surface area of the HP wall in cell i in contact with the fuel, and

$T_{hpw,i}^n$ wall temperature of the HP wall in cell i at beginning of time step.

The “effective” heat transfer coefficient is currently user-specified on input card COR_HPM1 or COR_HPM2 as H_FU (see MELCOR User’s Manual). The super script "n" denotes beginning of time step values, with "n+1" denoting end of timestep values.

In the condenser region, the temperature of the coolant in the CVH volume seeing the HP wall, $T_{cvh,i}^n$, is assumed constant during the timestep, and is used in the convective heat transfer boundary condition for the equations in that region. The condition takes the form:

$$Q_i^{n+1} = h_{eff,i} A_{hpw,i} (T_{cvh,i}^n - T_{hpw,i}^{n+1}) \tag{8-2}$$

where the CVH coolant temperature is provided by the CVH package and the HP internal model provides

$h_{eff,i}$ an "effective" heat transfer coefficient in cell i,

$A_{hpw,i}$ surface area of the HP wall in cell "i" in contact with the coolant,

and the HP internal model integrates its equation set over a time step to compute the end-of-step values for

$T_{hpw,i}^{n+1}$ wall temperature of the HP wall in cell "i" at end of time step, and

Q_i^{n+1} heat transfer rate at HP wall to coolant in cell "i" at end of time step.

Note that because the CVH coolant temperature is assumed constant in this equation, that this numerical approximation imposes a time-step stability constraint that must be respected to avoid numerical oscillations in the CVH coolant.

The details of how the HP wall temperatures and the effective heat transfer coefficients are calculated are specific to the HP internal model. A wide range of different choices with differing levels of fidelity and computational cost are possible. At present, two functional HP internal models that meet the interface requirements have been written to test and demonstrate the approach and the associated interfaces with other MELCOR models and packages.

8.4 Two Simple HP internal models for MELCOR-2

A wide range of transient heat pipe models have been developed and described in the literature. Brocheny [73] finds it useful to classify these models into three categories of increasingly higher levels of detail and complexity:

(1). Simplified generic models, which are mostly analytical in nature.

(2). Geometrically-based numerical models that consider selected heat pipe regions (e.g. vapor, wick, and wall regions) but typically without accounting for fluid flow dynamics. and

(3). Functional detailed numerical models that capture in various degrees of complexity and dimensionality the detailed physics (liquid and vapor fluid flow, liquid-vapor interface physics, etc.).

Here we describe two models of the second type that meet the MELCOR 2 interface requirements. Model “1” treats the working fluid region as an extremely high thermal conductivity material. Model “2” treats the liquid and vapor phases in the working fluid region (currently either sodium or potassium) as being in thermodynamic equilibrium and applies analytical expressions or user-input tabular data to capture the effects of the capillary, boiling, and sonic limitations. Both current models apply steady-state approximations that limit their applicability to accurately capture detailed transient behavior.

8.4.1 Geometric Regions

Figure 8.3 illustrates the four heat pipe regions represented in both models:

(1). the heat pipe wall in the evaporator region,

(2). the heat pipe wall in the adiabatic region,

(3). the heat pipe wall in the condenser region, and

(4). An inner region modeled either as a working fluid (model 2) or a high conductivity (model 1) region.

The three heat pipe wall regions are discretized as control volumes whose axial heights are aligned with COR package axial cells.

The inner region (illustrated here for model 2) is modeled as a single control volume with a bulk average temperature of $\hat{T}_{\mathrm{wf}}$. (or $\hat{T}_{\mathrm{hcm}}$ for Model 1). Although the wick region is not explicitly modeled, its depth and porosity can be used to provide estimates for certain operating conditions (e.g. axial pressure drop, liquid and vapor velocities).

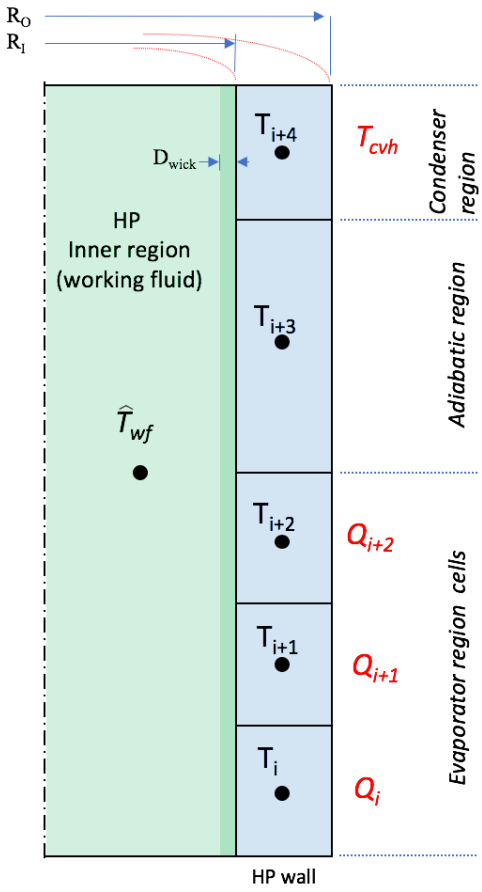


Figure 8.3 Heat Pipe regions and example discretization

In the equations describing Model 1 and Model 2, the following index references are used to refer to axial locations in the discretization.

IABE	index i corresponding to the bottom of the evaporator region
IATE	index i corresponding to the top of the evaporator region
IABC	index i corresponding to the bottom of the condenser region
IATC	index i corresponding to the top of the condenser region

8.4.2 HP Model Equations

The equations for two relatively simple geometrically-based numerical models are described in this section. Because the same geometric conceptualization is used in both models the equations for the HP wall region are identical. Model 1 treats the inner region as an extremely high conductivity material. In contrast, Model 2 uses a simplified thermodynamic model of a working fluid in the inner region (it assumes the working fluid is always in thermodynamic equilibrium) to qualitatively capture additional physical

behaviors associated with heat pipe devices. Model 2 provides more fidelity in representing fluid temperatures and pressures and would be required in predicting failures based on internal temperature and pressure.

8.4.2.1 Heat Pipe Wall Regions

An equation for conservation of thermal energy of the following form is written for each of the discretized "control volume" regions in the HP wall.

$$M_i C \frac{\partial T_i}{\partial t} = \sum_{f=1}^{4} A_{i,f} \dot{q}_{i,f} \tag{8-3}$$

where

M_i	mass of control volume i, kg
C	specific heat of wall material, J/(kg K)
T_i	temperature of control volume i, K
$A_{i,f}$	surface area of face "f" in control volume i, m^2
$q_{i,f}$	heat flux across face “f” in control volume i, J/(sec m^2)
f	denotes one of four faces to the control volume: inner, outer, bottom, top

Surface areas for each face are computed from the geometry of the heat pipe (inner and outer radius, axial heights of COR cells). Mass is computed from the heat pipe geometry and the specified density of the wall material. Specific heat is known based on a specified value for the wall material (currently assumed to be stainless steel). The heat flux across each of the four faces is computed as follows.

Inner face:	$\dot{q}_{i,f} = h_{i,f}(\hat{T}_{wf} - T_i)$	
Outer face:	$\dot{q}_{i,f} = h_{i,f}(T_{fuel} - T_i)$	for evaporator region cells
	$\dot{q}_{i,f} = h_{i,f}(T_{cvh} - T_i)$	for condenser region cells
	$\dot{q}_{i,f} = 0$	for adiabatic region cells
Bottom face:	$\dot{q}_{i,f} = \boldsymbol{k}\frac{(T_{i-1} - T_i)}{(Z_i - Z_{i-1})}$	for $i \neq 1$,
	$\dot{q}_{i,f} = 0$	for $i = 1$,
Top face:	$\dot{q}_{i,f} = \boldsymbol{k}\frac{(T_{i+1} - T_i)}{(Z_{i+1} - Z_i)}$	for $i \neq IATC$

$$\dot{q}_{i,f} = 0 \qquad \text{for } i = IATC,$$

where

$h_{i,f}$	an "effective" heat transfer coefficient for face *f* of cell i, J/(sec m^2 K)
T_{fuel}	temperature of the fuel component in cell i, K
T_{cvh}	temperature of the CVH coolant associated with cell i, K
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)

The material properties of the wall material (i.e. density, thermal conductivity, and specific heat) are currently taken to be those of stainless steel.

8.4.2.2 Model 1 Inner Region: High Thermal Conductivity Material

Model 1 treats the inner region as consisting of an extremely high thermal conductivity. The value of the thermal conductivity $\boldsymbol{k_{hcm}}$, can be specified by the user and has a default value of 1.e5 W/(m K). Values for the density and specific heat in this region are set to 1.0 kg/m^3 and 1000 J/(kg K), respectively. Note that these two values have negligible impact on a steady solution.

Although the model formulation is based on steady-state approximations, the code must evolve the solution in time to reach steady conditions. Under the above assumptions the working-fluid energy time rate of change in the control volume can be expressed as

$$M_{hcm} C_{hcm} \frac{\partial T_{hcm}}{\partial t} = - \sum_{i=IABE}^{i=IATC} A_{i,1} \dot{q}_{i,1} \qquad (8\text{-}4)$$

where

M_{hcm}	total mass of the high-conductivity-material, kg
C_{hcm}	effective specific heat of the high-conductivity-material, J/(kg K)
T_{hcm}	temperature of the high-conductivity-material, K
$A_{i,1}$	surface area of face "f=1" (inner face) in control volume i, m^2
$\dot{q}_{i,1}$	heat flux across face "f=1" (inner face) in control volume i, J/(sec m^2)

8.4.2.3 Model 1 Effective Heat Transfer

Models for effective heat transfer coefficients are needed for

- fuel to wall in the evaporator section
- CVH coolant to wall in the condenser section

- high-conductivity material to wall in the evaporator section, and
- high-conductivity material to wall in the condenser section.

In Model 1, these are calculated as follows.

Fuel to wall in evaporator: $h_{i,2}$, $IABE \leq i \leq IATE$

$$h_{i,2} = [dr_{wall}/2\boldsymbol{k} + 1/h_{FU}]^{-1} \quad (8\text{-}5)$$

where

dr_{wall}	thickness of the HP wall, m
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)
h_{FU}	a constant user specified value, J/(sec m^2 K)

The factor 2 results from the cell-centered temperature in the HP wall leading to a conduction path equal to half the wall thickness.

CVH coolant to wall in condenser: $h_{i,2}$, $IABC \leq i \leq IATC$

$$h_{i,2} = [dr_{wall}/2\boldsymbol{k} + 1/h_{CVH}]^{-1} \quad (8\text{-}6)$$

where

dr_{wall}	thickness of the HP wall, m
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)
h_{CVH}	a constant user specified value, J/(sec m^2 K)

High-Conductivity-Material to wall in the evaporator section: $h_{i,1}$, $IABE \leq i \leq IATE$

$$h_{i,1} = 2/[dr_{wall}/\boldsymbol{k} + dr_{wick}/\boldsymbol{k_{hcm}}] \quad (8\text{-}7)$$

where

dr_{wall}	thickness of the HP wall, m
dr_{wick}	thickness of the HP wick, m
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)
$\boldsymbol{k_{hcm}}$	thermal conductivity of the high-conductivity-material, J/(sec m K)

High-Conductivity-Material to wall in the condenser section: $h_{i,1}$, $IABC \leq i \leq IATC$

$$h_{i,1} = 2/[dr_{wall}/\boldsymbol{k} + dr_{wick}/\boldsymbol{k_{hcm}}] \quad (8\text{-}8)$$

where

dr_{wall}	thickness of the HP wall, m
dr_{wick}	thickness of the HP wick, m
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)
$\boldsymbol{k_{hcm}}$	thermal conductivity of the high-conductivity-material, J/(sec m K)

8.4.2.4 Model 1 Temporal discretization and time-step integration

The energy equations for HP wall control volumes and the high-conductivity-material control volume are advanced in time using the following time discretization. In these equations the old and new time-step values are denoted by the "n" and "n+1" superscripts respectively. To highlight this, the time "n" temperatures used in the equations are written in red. Note that the second term on the RHS of the evaporator equation (enclosed in brackets) is a precalculated (i.e. explicit) heat source computed using the previous time-step values. Also, the CVH coolant temperature T_{cvh}^{n} is a boundary condition temperature at time step n. In isolation this equation set does not have a stability constraint on the time step. However, the coupling of these equations to the CVH and COR package introduces a timestep stability limit that must be respected.

Evaporator

$$\frac{M_i C}{dt}\left(T_i^{n+1} - T_i^{n}\right) = A_{i,1}h_{i,1}\left(\hat{T}_{wf}^{n+1} - T_i^{n+1}\right) + \left[A_{i,2}h_{i,2}\left(T_{fuel}^{n} - T_i^{n}\right)\right] + A_{i,3}h_{i,3}\left(T_{i-1}^{n+1} - T_i^{n+1}\right) + A_{i,4}h_{i,4}\left(T_{i+1}^{n+1} - T_i^{n+1}\right) \quad (8\text{-}9)$$

Adiabatic

$$\frac{M_i C}{dt}\left(T_i^{n+1} - T_i^{n}\right) = A_{i,1}h_{i,1}\left(\hat{T}_{wf}^{n+1} - T_i^{n+1}\right) + A_{i,3}h_{i,3}\left(T_{i-1}^{n+1} - T_i^{n+1}\right) + A_{i,4}h_{i,4}\left(T_{i+1}^{n+1} - T_i^{n+1}\right) \quad (8\text{-}10)$$

Condenser

$$\frac{M_i C}{dt}\left(T_i^{n+1} - T_i^{n}\right) = A_{i,1}h_{i,1}\left(\hat{T}_{wf}^{n+1} - T_i^{n+1}\right) + A_{i,2}h_{i,2}\left(T_{cvh}^{n} - T_i^{n+1}\right) + A_{i,3}h_{i,3}\left(T_{i-1}^{n+1} - T_i^{n+1}\right) + A_{i,4}h_{i,4}\left(T_{i+1}^{n+1} - T_i^{n+1}\right) \quad (8\text{-}11)$$

High Conductivity Material

$$\frac{M_{hcm}C_{hcm}}{dt}\left(T_{hcm}^{n+1}-T_{hcm}^{n}\right)=-\sum_{i=IABE}^{i=IATC}A_{i,1}h_{i,1}\left(\hat{T}_{hcm}^{n+1}-T_{i}^{n+1}\right) \quad (8\text{-}12)$$

These equations are cast into matrix form and solved directly using Gaussian elimination. The residuals are checked to confirm that the solution is valid. The energy in the working fluid at the new time step is then updated as follows:

$$E_{hcm}^{n+1}=E_{hcm}^{n}+M_{hcm}C_{hcm}\left(T_{hcm}^{n+1}-T_{hcm}^{n}\right). \quad (8\text{-}13)$$

8.4.2.5 Model 2 Inner Region: Working Fluid

The key modeling assumption in Model 2 is that the working fluid is in thermodynamic equilibrium. The equilibrium state is calculated based on knowing

V_{wf}	volume of the working-fluid control volume (m^2),
M_{wf}	mass of the working fluid in the control volume (kg), and
H_{wf}	enthalpy of the working fluid in the control volume (J),

where both the volume and mass of the working fluid are constant. Knowing the total enthalpy, the values of bulk equilibrium temperature $\hat{T}_{wf}$, pressure $\hat{P}_{wf}$, and the vapor/liquid mass fractions can be calculated using an equation of state.

There are two working fluids currently supported: sodium and potassium. For use in modeling the heat pipe working fluid, equations for the following thermodynamic properties have been coded as found in References [74], [75], [76], and [77].

- vapor pressure over saturated liquid
- enthalpy of liquid along the saturation curve
- enthalpy of vaporization
- enthalpy of vapor above the saturated liquid
- temperature derivative of the pressure along the saturation curve
- density of saturated liquid
- density of vapor above the saturated liquid
- specific volume of saturated liquid.
- specific volume of sodium vapor above the saturated
- ratio of specific heats Cp/Cv of the vapor
- thermal conductivity of liquid
- thermal conductivity of vapor
- dynamic viscosity of liquid
- dynamic viscosity of vapor

Until failure, the working fluid in each heat pipe has a fixed mass and volume. Thus, its equilibrium thermodynamic state only depends on changes in the energy that occur by heat transfer across its walls. Applying the user-specified geometry and mass, which are defined by input together with thermodynamic properties of the working fluid, a look-up table is generated during the initialization stage of the calculation for each unique heat pipe type. This table provides values for key properties (pressure, temperature, quality, specific heat) as a function of specific enthalpy to be used in the calculations.

8.4.2.6 Model 2 Effective Heat Transfer

Models for effective heat transfer coefficients are needed for

- fuel to wall in the evaporator section
- CVH coolant to wall in the condenser section
- working fluid to wall in the evaporator section, and
- working fluid to wall in the condenser section.

In Model 2, these are calculated the same as in Model 1.

Fuel to wall in evaporator: $h_{i,2}\ ,\ IABE \le i \le IATE$

$$h_{i,2} = [dr_{wall}/2\boldsymbol{k} + 1/h_{FU}]^{-1} \tag{8-14}$$

where

dr_{wall}	thickness of the HP wall, m
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)
h_{FU}	a constant user specified heat transfer coefficient, J/(sec m^2 K)

CVH coolant to wall in condenser: $h_{i,2}\ ,\ IABC \le i \le IATC$

$$h_{i,2} = [dr_{wall}/2\boldsymbol{k} + 1/h_{CVH}]^{-1} \tag{8-15}$$

where

dr_{wall}	thickness of the HP wall, m
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)
h_{CVH}	a constant user specified heat transfer coefficient, J/(sec m^2 K)

Working-fluid to wall in the evaporator section: $h_{i,1}\ ,\ IABE \le i \le IATE$

$$h_{i,1} = \left[dr_{wall}/2\boldsymbol{k} + dr_{wick}/2\boldsymbol{k_{wf,liq}}\right]^{-1} + \boldsymbol{2k_{wf,vap}}/r_{wall} + \tag{8-16}$$

where

dr_{wall}	thickness of the HP wall, m
$\boldsymbol{dr_{wick}}$	thickness of the HP wick, m
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)
$\boldsymbol{k_{wf,liq}}$	thermal conductivity of the working fluid liquid phase, J/(sec m K)

Working fluid to wall in the condenser section: $h_{i,1}$, $IABC \le i \le IATC$

$$h_{i,1} = \left[dr_{wall}/2\boldsymbol{k} + dr_{wick}/2\boldsymbol{k_{wf}}\right]^{-1} + \boldsymbol{k_{wf,vap}}/r_{wall} \tag{8-17}$$

where

dr_{wall}	thickness of the HP wall, m
dr_{wick}	thickness of the HP wick, m
$\boldsymbol{k}$	thermal conductivity of the HP wall material, J/(sec m K)
$\boldsymbol{k_{wf,liq}}$	thermal conductivity of the working fluid liquid phase, J/(sec m K)

8.4.2.7 Model 2 Temporal discretization and time-step integration

The energy equations for HP wall control volumes and the working fluid control volume are advanced in time using the following time discretization. In these equations the old and new time-step values are denoted by the "n" and "n+1" superscripts respectively. To highlight this, the time "n" temperatures used in the equations are written in red. Note that the second term on the RHS of the evaporator equation (enclosed in brackets) is a precalculated (i.e. explicit) heat source computed using the previous time-step values. Also, the CVH coolant temperature T_{cvh}^{n} is a boundary condition temperature at time step n. In isolation this equation set does not have a stability constraint on the time step. However, the coupling of these equations to the CVH and COR package introduces a time-step stability limit that must be respected.

Evaporator

$$\frac{M_i C}{dt}\left(T_i^{n+1} - T_i^{n}\right) = A_{i,1}h_{i,1}\left(\hat{T}_{wf}^{n+1} - T_i^{n+1}\right) + \left[A_{i,2}h_{i,2}\left(T_{fuel}^{n} - T_i^{n}\right)\right] + A_{i,3}h_{i,3}\left(T_{i-1}^{n+1} - T_i^{n+1}\right) + A_{i,4}h_{i,4}\left(T_{i+1}^{n+1} - T_i^{n+1}\right) \tag{8-18}$$

Adiabatic

$$\frac{M_i C}{dt}\left(T_i^{n+1} - T_i^{n}\right) = A_{i,1}h_{i,1}\left(\hat{T}_{wf}^{n+1} - T_i^{n+1}\right) + A_{i,3}h_{i,3}\left(T_{i-1}^{n+1} - T_i^{n+1}\right) + A_{i,4}h_{i,4}\left(T_{i+1}^{n+1} - T_i^{n+1}\right) \tag{8-19}$$

Condenser

$$\frac{M_i C}{dt}\left(T_i^{n+1} - T_i^n\right) = A_{i,1}h_{i,1}\left(\hat{T}_{wf}^{n+1} - T_i^{n+1}\right) + A_{i,2}h_{i,2}\left(T_{cvh}^n - T_i^{n+1}\right) + A_{i,3}h_{i,3}\left(T_{i-1}^{n+1} - T_i^{n+1}\right) + A_{i,4}h_{i,4}\left(T_{i+1}^{n+1} - T_i^{n+1}\right) \quad (8\text{-}20)$$

Working fluid

$$\frac{M_{wf}C_{wf}}{dt}\left(T_{wf}^{n+1} - T_{wf}^n\right) = -\sum_{i=IABE}^{i=IATC} A_{i,1}h_{i,1}\left(\hat{T}_{wf}^{n+1} - T_i^{n+1}\right) \quad (8\text{-}21)$$

These equations are cast into matrix form and solved directly using Gaussian elimination. The residuals are checked to confirm that the solution is valid. The enthalpy in the working fluid at the new time step is then updated as follows:

$$H_{wf}^{n+1} = H_{wf}^n + M_{wf}C_{wf}\left(T_{wf}^{n+1} - T_{wf}^n\right). \quad (8\text{-}22)$$

8.5 Heat Pipe Energy Transfer Limits

There are a variety of steady-state operational limits that can constrain the heat rejection capacity of a heat pipe under operating conditions. Details of these limits are well described in the HP literature and will not be reviewed here. Currently the following three operational limits can be modeled in HP Model 2 (Model 1 does not provide for operations limits).

Sonic limit: This is associated with choked flow of vapor through the central core.

Capillary flow limit: This is associated with the liquid flow rate through the wick at maximum capillary pressure difference.

Boiling limits: As heat flux increases, the onset of both nucleate and film boiling related issues can disrupt heat transfer. Film boiling in particular can lead to a sudden drop in heat transfer efficiency.

Each of these limits depends on HP-specific details (geometry, materials, type of wick, working fluid etc.) and will vary in magnitude based on operating conditions. These limits can be described by power vs operating temperature curves, and can be measured experimentally and/or estimated using analytical or numerical models.

Sonic, capillary and boiling limits can be specified in the MELCOR input deck through user-specified tabular functions. Also, for the sonic limit only, an analytical expression suggested by Faghri ([70], see Eq. 3-65) is applied if a tabular function was not provided by the user. If the user does not specify tabular functions for the capillary or boiling limits,

then they are simply not applied. Each of the three operational limits affects the equations solved by MELCOR in different ways.

If the boiling limit is exceeded at a particular axial level in the evaporator region, then the heat transfer coefficient between the working fluid and the heat pipe wall is set to a near-zero value of 1.e-4 W/m^2-K. This severe reduction in the heat transfer coefficient will lead to a rapid rise in the local heat pipe wall temperature.

If the capillary limit is exceeded, then the overall heat transfer rate in the evaporator region is restricted based on the availability of liquid phase working fluid calculated from the capillary limit.

If the sonic limit is exceeded, then the boundary condition between the HP wall and the HX coolant is changed, and the modified equation set is solved over again. In the modified equations the condenser region boundary condition is given a specified value equal to the value computed from the sonic limit.

In a HP reactor system, there also is another potentially important limit associated with the capacity of the reactor cooling system to accept heat from the HP condenser region. We refer to this as the "Condenser HX limit", and it can be an important factor when modeling accidents where the heat exchanger operation and/or effectiveness has been compromised.

Under steady-state conditions, we can express the heat transfer rate in the heat exchanger as an energy balance in the following form.

$$Q_x = h_{x,ef} A_x (T_{wf} - T_{x,o}) = \dot{m}_{fx}\, Cp_f (T_{x,o} - T_{x,in}) \tag{8-23}$$

where

Q_x	heat transfer rate in the heat exchanger
$h_{x,ef}$	an effective heat transfer coefficient in the heat exchanger (HP to coolant)
A_x	surface area of condenser wall in heat exchanger
$\dot{m}_{fx}$	mass flow rate of coolant into the heat exchanger
Cp_f	specific heat of the coolant in the heat exchanger
T_{wf}	temperature of the HP working fluid
$T_{x,o}$	outflow temperature of the coolant from the heat exchanger
$T_{x,in}$	inflow temperature of the coolant into the heat exchanger

We can solve for the outflow temperature of the coolant as

$$T_{x,o} = {(T_{wf} + \beta T_{x,in})}/{(1 + \beta)} \quad (8\text{-}24)$$

where

$$\beta = \dot{m}_{fx}\, Cp_f \,/\, h_{x,ef} A_x$$

Therefore, at steady state, the heat transfer rate in the heat exchanger must satisfy the following.

$$Q_x = \dot{m}_{fx}\, Cp_f \left(\left[{(T_{wf} + \beta T_{x,in})}/{(1 + \beta)} \right] - T_{x,in} \right) \quad (8\text{-}25)$$

A theoretical bounding value can be found by taking the limit as $h_{x,ef} \rightarrow$ infinity, yielding $T_{x,o} = T_{wf}$, and

$$Q_{x,limit} = \dot{m}_{fx}\, Cp_f \left(T_{wf} - T_{x,in} \right) \quad (8\text{-}26)$$

This expresses a reactor-system-based bounding limit that is defined by the inlet temperature, mass flow rate and thermal capacitance of the heat exchanger coolant.

8.6 Numerical Stability Considerations

The HP model equations are cast in fully implicit numerical form and are unconditionally stable in isolation. However, the coupling of these equations to the CVH and COR package is currently explicit in nature. The energy transfer from the fuel into the HP wall is calculated using beginning of time-step (superscript n) heat pipe wall temperatures, and acts as an explicit source term to the HP model. In the condenser region, the temperature of the CVH coolant is at the time "n" state. Thus, rejection of heat into the CVH coolant is pre-calculated and passed as an explicit source to the CVH package.

Because these inter-package energy transfers are explicit, unphysical numerical oscillations would potentially occur if the MELCOR system time steps were too large. Due to the very large effective thermal capacitance of the heat pipe (from working fluid heat of vaporization), the coupling at the fuel-wall boundary does not impose a numerical stability problem for time-step values normally used in MELCOR. However, this is not true for the coupling to the CVH package, where serious numerical problems would occur for many conditions of interest in accident analysis.

The method currently used to address the stability problem is based on restricting the explicit heat transfer to a maximum value based on an estimate of the physically realizable energy absorption capacity of the receiving control volume fluid. This limit is constantly monitored in the calculation. If exceeded, then the boundary condition between the HP wall and the HX coolant is changed in the equations to be a fixed value equal to the estimated limit, and the equation set is solved over again (like what is done for the sonic

limit condition). Although the method conserves energy exactly, the temporal (i.e. time integration related) error introduced is potentially large. For MELCOR purposes, this "local" temporal numerical error is acceptable because the estimated heat transfer coefficients are themselves very approximate, being based on user-specified or steady-state correlations whose applicability to short time-scale transients is already subject to large errors.

8.7 How HP Failure is Modeled

For MELCOR heat pipe "failure" means the complete and non-recoverable loss of ability to transfer heat. From an accident analysis perspective, this is a critical event. Several types of failure modes might by hypothesized to potentially occur in a HP reactor. Examples include:

- HP wall or end-cap failure due to time-at-temperature if the HP is subjected to high operating temperatures and associated pressures, such as might occur in a complete loss of heat sink (e.g., the heat exchanger fails)

- Local melt-through of the HP wall due to a sudden influx of heat

- HP wall or micro-imperfections in end-cap welds or wall materials after being subjected to time at operating temperatures and pressures.

Note that the failure of a single HP, or even a group of HPs, does not necessarily lead to other failures, because HP operation can adjust automatically.

For HP Model 1, failure only occurs at a user specified failure time, and thus it has no associated physical modeling. For HP Model 2, several failure conditions are considered. The first condition is temperature based. If any axial section of a HP wall exceeds a user specified temperature (typically assigned a value at or near the melting temperature of the HP wall material), then the HP fails. The second option is based on a user specified control function. MELCOR allows the user to specify a logical control function that, when TRUE, triggers failure of the heat pipe. For example, a Larson-Miller type approach based on cumulative time at temperature and pressure might be used.

When a representative HP fails, the HP modeling described in this section is deactivated, and the HP wall material is transferred to a standard COR package "CN" component with corresponding mass, energy, and surface area. All subsequent modeling of this material component (heat transfer, melting, relocation, etc.) behave as described elsewhere in MELCOR documentation. It is also useful (but not required) to have the working fluid volume within a collection of HPs be associated with a CVH package control volume (CV) and a MELCOR flow path (FP) defined which connects this CV to other CVs modeling the geometric domain outside and surrounding the HPs. Before failure the CV is essentially inactive from a CVH package standpoint. However, a control function can be directed to

open the FP upon HP failure, and thereby allow CVH material, and any associated fission products that have been modeled by the user, to be released into the CVH system network.

9. Specialized High Temperature Gas Cooled Reactor Models

In addition to convection, conduction, and point kinetics models (See Sections 2.2.6, 2.2.9, 2.3.5, and 7), a suite of physics models and code capabilities for fuel performance and failure was added for HTGRs. Diffusive transport of fission product species – with allowance for vapor diffusion at gas gaps and/or coolant boundaries – can be modeled during normal and accident conditions. The fuel performance/failure modeling/capability approach is flexible and can simultaneously model a diverse population of TRISO particles in an HTGR core including, e.g., intact, defective, and failed. TRISO particles may transition between intact and failed subject to fuel failure models.

To perform fission product transport/release calculations, the user defines:

- “Models” that each represents a kind of TRISO particle (e.g. intact, defective, failed, or some other cohort of the overall TRISO particle population)
- A single model representing the fuel element matrix
- One or more tracked fission product species that correlate to radionuclide classes per DCH, RN1, and COR user input
- Global parameters describing the HTGR (PBR or PMR) core
- A one-dimensional spherical or cylindrical finite volume solution grid for each defined model (comprised of zones consisting of equally-spaced nodes)
- Diffusion coefficients (Arrhenius temperature relations)
- Miscellaneous data (e.g. TRISO failure and/or population information, sorption isotherms, partition coefficients)

MELCOR computes – for each tracked fission product species - concentration within and release from each model. The TRISO particle population models release radionuclides to the matrix model, and the matrix model releases to coolant. The matrix/coolant interface constitutes the hand-off of radionuclide inventory from COR to CVH/RN1. The CVH and RN1 packages perform radionuclide transport modeling thereafter.

9.1 Radionuclide Transport and Release

The HTGR fission product release model employs a one-dimensional finite volume diffusion equation solver - in steady-state or transient mode and in spherical or cylindrical

coordinates – to ascertain radionuclide distributions (i.e. molar concentration profiles) within fuel and radionuclide release from fuel. For purposes of the HTGR fission product release model, fuel is entirely represented by user-defined models such that the calculated concentration profiles imply the distribution of radionuclides within the fuel (FU) and matrix (MX) COR components. The concentration profiles also imply diffusional release from fuel to matrix and subsequently to coolant, or equivalently from the FU component to the MX component and subsequently to coolant. The solution approach for a general transient/accident scenario consists of three sequential stages:

- Steady-state diffusion
- Steady-state transport
- Transient diffusion/transport

The steady-state diffusion stage predicts end-of-burnup (start of transient) radionuclide distribution and release. Accordingly, the steady-state diffusion stage plays a role in predicting transient/accident initial conditions. This steady-state diffusion computation occurs within a timestep at user-specified problem time and seeks to simultaneously satisfy the set of conditions imposed by DCH, RN1, and COR input.

The steady-state transport calculation occurs over a span of problem time between the steady-state diffusion calculation and the start of the transient diffusion/transport calculation. The steady-state transport stage rounds out the characterization of transient/accident initial conditions. The purpose of the steady-state transport stage is to determine steady-state trends in CVH, HS, and RN1 radionuclide distributions (in CVH control volumes, on HS deposition surfaces, and distributed across applicable RN1 forms and sections) given the steady releases from COR components. The steady release rates from COR components are held constant (without decrementing COR component radionuclide inventory) while CVH, HS, and RN1 march through problem time towards an end-state where CV radionuclide accumulation and HS surface deposition rates are approximately constant. The steady trends in the steady state transport calculation are then scaled to obtain proper end-of-burnup CVH, HS, and RN1 inventories.

The transient diffusion/transport stage begins with conditions predicted from the steady-state diffusion and transport stages. A transient form of the diffusion equation is solved to predict changes in radionuclide distribution and release (COR FU and MX components). COR component radionuclide inventories can decrease at this point as fission has presumably ceased while diffusional release is ongoing. CVH, HS, RN1, and all other physics packages are active simultaneously to predict the evolution of coolant and deposition surface radionuclide inventory. Within the HTGR fission product release model, TRISO population transfers are allowed according to user input and – if enabled – an analytic release model is applied in context of the “failed” TRISO model. Figure 9.1 illustrates the generic three-stage solution approach including inputs/outputs for each stage.

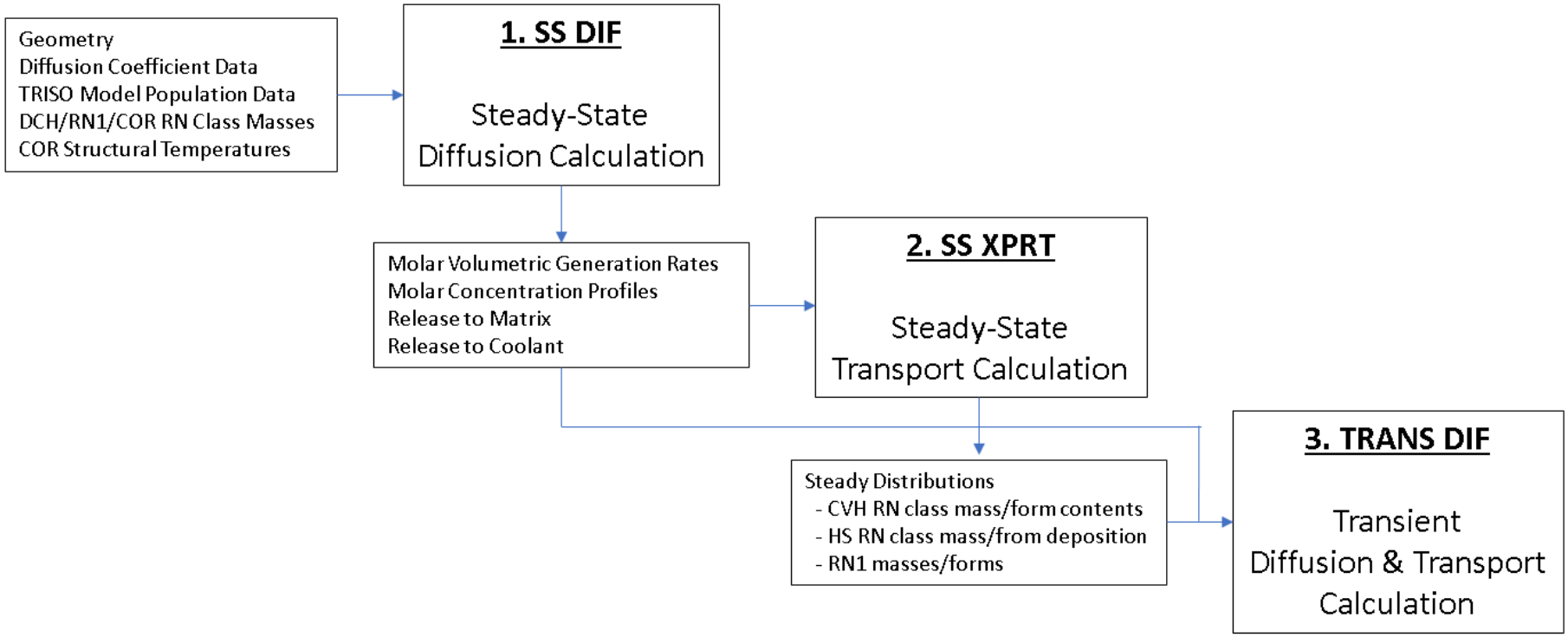


Figure 9.1 Generalized computational grid for diffusion equation solution

9.1.1 Tracked Fission Product Species Diffusion

The one-dimensional finite diffusion solver is integral to HTGR fission product release calculations in MELCOR. A general description follows, and further mathematical detail is included in Appendix A as referenced.

9.1.1.1 Diffusion Equation

The one-dimensional diffusion equation - accounting for fission product species decay and for fission product species generation - is:

$$\chi \frac{\partial C}{\partial t} = \frac{1}{r^n} \frac{\partial}{\partial r} \left(r^n D \frac{\partial C}{\partial r} \right) - \lambda C + \beta \tag{9-1}$$

Where:

C	=	Concentration of fission product (nuclide) [kmol/m^3]
D	=	Diffusion coefficient [m^2/s]
r	=	Radial coordinate [m]
λ	=	Fission product (nuclide) decay constant [1/s]
β	=	Fission product (nuclide) generation/source term [kmol/m^3/s]
n	=	1 for cylindrical coordinates, 2 for spherical coordinates
χ	=	0 for steady-state, 1 for transient

9.1.1.2 Diffusion Coefficients

Diffusion coefficients are temperature-dependent and assume an Arrhenius form:

$$D(T) = D_0 e^{-Q/RT} \tag{9-2}$$

Where:

D_0	=	Coefficient [m^2/s]
Q	=	Activation energy [J/mol]
R	=	Gas constant = 8.314 [J/mol/K]
T	=	Temperature [K]

9.1.1.3 Computational Domain

The diffusion equation is solved for each tracked fission product species on a one-dimensional grid for each user-defined “model”. Models represent the various types or conditions of TRISO particles in the core inventory. They are also used to represent the fuel matrix portion of a fuel element. A single HTGR simulation may, for example, contain a model for intact TRISO fuel particles, for intact TRISO fuel particles with failed SiC layer, for failed TRISO particles, and for the fuel element matrix. Each model has its own computational grid consisting of material zones that are themselves comprised of some number of equally-spaced computational nodes. Solving diffusion equations for each node in each zone of each given model for all tracked fission product species yields an estimate of radionuclide distribution within fuel. This assumes a mapping between tracked fission product species and DCH/RN1 radionuclide classes according to DCH/RN1 input. Figure 9.2 depicts a generalized computational grid consisting of N total nodes across M total zones for a given tracked fission product species i on a given model.

As Appendix A.1 outlines, a system of equations can be formulated for each tracked fission product species on each model. The system can be solved with an iterative method (successive over-relaxation). Each nodal equation has the form:

$$a_i C_{i-1}^n + b_i C_i^n + c_i C_{i+1}^n = d_i \tag{9-3}$$

This form is amenable to the matrix form input to the iterative method.

In general, the values of coefficients a_i, b_i, c_i, d_i can be ascertained by comparison with the solution form of the difference equation above. Appendix A.1, A.2, and A.3 elaborate on the various possible prescriptions for these coefficients.

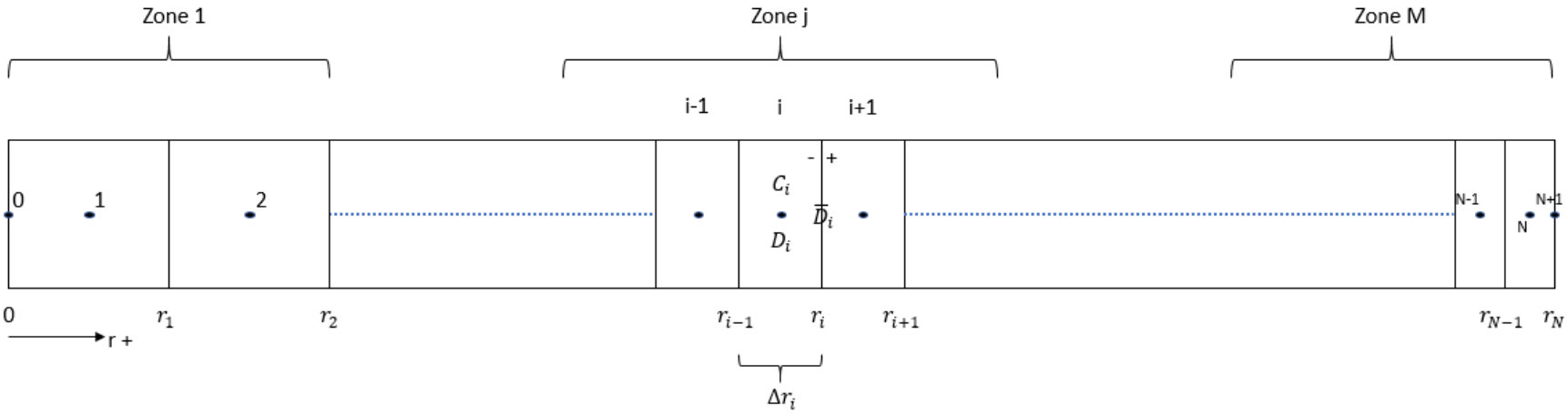


Figure 9.2 Generalized computational grid for diffusion equation solution

9.1.1.4 Boundary Conditions

A symmetry (zero slope or zero flux) condition is imposed at the interior boundary of each model. For the exterior boundary of a given model a specified zero concentration condition is imposed. These assumptions pertain to each tracked fission product species for each model. For each boundary condition, the boundary node coefficient a_i, b_i, c_i, d_i modifications are outlined in mathematical detail in Appendix A.1.3.

9.1.1.5 Sorption Isotherms

The sorption isotherm model describes the transition in fission product concentration occurring at/near a solid/gas interface. Fission products that diffuse through a solid to a solid/gas interface must – as a prerequisite to further release - evaporate to gas and then either (1) be removed by advection, or (2) diffuse through gas and sorb/deposit on other surfaces before continuing to diffuse. The sorption isotherm empirical model describes the non-linear, temperature-dependent relationship between sorbate fission product concentration (in the sorbent, or solid-phase) and sorbate fission product gas-phase vapor pressure assuming (1) spontaneous adjustment to equilibrium between these quantities, and (2) continued existence of equilibrium between these two quantities. This empirical model is useful in the context of in-core fission product transport calculations because it can describe:

- Fission product behavior around the small gap between the fuel compact outer surface and the graphite webbing inner surface (prismatic HTGR using fuel compacts within graphite hex blocks)
- Fission product behavior at a surface/coolant boundary (prismatic HTGR in coolant holes, pebble bed HTGR at pebble outer surface)

The sorption isotherm model can be leveraged to fit within a finite volume diffusion calculation scheme.

9.1.1.5.1 Sorption Isotherm Empirical Model

Vapor pressure and solid-phase concentration per unit mass (typical units of Pa and μmol/g, respectively) are related by empirical sorption isotherm equations. Generally:

$$P = P(\boldsymbol{C}, T) \tag{9-4}$$

More specifically, the total vapor pressure of a sorbate is the sum of two vapor pressure components corresponding to two different regimes or regions: Henrian/Langmuir and Freundlich. Within each component, the logarithm of vapor pressure is correlated to the logarithm of sorbate concentration via the empirical model formulation:

$$P = P_F + P_H \tag{9-5}$$

$$\ln(P_F) = \left(A + {}^{B}/_{T}\right) + \left(D + {}^{E}/_{T}\right)\ln(\boldsymbol{C}) \quad (9\text{-}6)$$

$$\ln(P_H) = \left(A + {}^{B}/_{T}\right) + \left(D - 1 + {}^{E}/_{T}\right)\ln(\boldsymbol{C}_{tr}) + \ln(\boldsymbol{C}) \quad (9\text{-}7)$$

Where:

P_F	=	Freundlich pressure [Pa]
P_H	=	Henrian pressure [Pa]
A,B,D,E	=	Empirical constants
T	=	Temperature [K] indicative of sorbate (e.g. a solid surface temperature)
$\boldsymbol{C}$	=	Mass concentration of sorbate [µmol/g]
$\boldsymbol{C}_{tr}$	=	Transition (Henrian to Freundlich) mass concentration [µmol/g]

The transition concentration is sometimes modeled as a constant and sometimes modeled empirically as:

$$\ln(\boldsymbol{C}_{tr}) = F + {}^{G}/_{T} \quad (9\text{-}8)$$

Where:

F,G	=	Empirical constants

Note also that the volumetric concentration is related to the mass concentration via:

$$\boldsymbol{C} = {}^{c}/_{\rho} \quad (9\text{-}9)$$

Where:

c	=	Volumetric concentration [µmol/cm^3]
ρ	=	Mass density [g/cm^3]

The convention chosen here is that c represents volumetric concentration while $\boldsymbol{C}$ represents mass concentration. Note also that – under an ideal gas assumption – the volumetric concentration of a sorbate can be uniquely determined as a function of vapor pressure:

$$c = \frac{P}{RT} \quad (9\text{-}10)$$

Where:

P = Vapor pressure of sorbate [Pa]

R = Universal gas constant, e.g. 8.314 [Pa*m^3/mol/K]

T = Temperature [K] indicative of sorbate (e.g. a solid surface temperature)

Note the importance of dimensional consistency between R, P, and the desired concentration units when employing the ideal gas equation. The final form of the pressure/concentration relationship is:

$$P(\boldsymbol{C},T) = e^{X(T)}\left[\boldsymbol{C}_{tr}^{Y(T)-1}\boldsymbol{C} + C^{Y(T)}\right] \quad (9\text{-}11)$$

Where:

$X(T)$ = $A + {}^{B}/_{T}$

$Y(T)$ = $D + {}^{E}/_{T}$

The above relationships may be deployed alongside other assumptions to treat – in the context of a finite volume diffusion solution - situations where fission product species transport is not by pure diffusion through solid graphite:

- A gas gap exists in a fuel element (e.g., the helium gap between a fuel compact and the graphite hex block webbing in a PMR)
- A graphite surface interfaces to coolant (e.g., the graphite hex block coolant hole surfaces in a PMR or the pebble outer surface in a PBR)

When such an interface exists, it may be defined with respect to the sorption isotherm empirical model and the typical nodal difference equations for neighboring nodes can be rewritten accordingly.

9.1.1.5.2 Gas Gap Interface Treatment in Diffusion Solution

A gas gap interface model is meant to treat fission product transport across a small gas gap between two solid surfaces. This occurs, for example, in a PMR where a fuel compact outer surface interfaces with the graphite webbing of its host graphite hexagonal block across a small helium gap. Separate sorption isotherm empirical models are applied for each surface, and the diffusion equations of each node for which the gas gap interface makes up a boundary are modified. The key physical assumption is vapor pressure equilibrium on both sides of the gas gap. Thus, for a given fission product species, the solid sorbate concentration at the inner surface generates a vapor pressure that is equal to the vapor pressure generated by the solid sorbate concentration at the outer surface. Since the sorption isotherm empirical model describes these equivalent vapor pressures, a nonlinear relationship between the sorbate concentrations at each surface may be derived. This condition is then effectively imposed as a nonlinear boundary condition that

requires linearization and iteration to solve in the context of the diffusion solution. Note this strategy resembles that of the TRAFIC code, which is meant to model release of metallic fission products from an HTGR core [78]. The key difference in MELCOR is that a finite volume solution of the diffusion problem is used as opposed to alternative methods (e.g., of TRAFIC). Thus, the specific near-gap nodal difference equations must be derived.

In the MELCOR diffusion solution, a gas gap interface may exist in a model between two zones such that the nodes on either side of the interface belong to separate, neighboring zones. The intent is that the gas gap interface would be used in a PMR type reactor in the matrix model between a fuel-bearing inner zone (that gets fission product source terms from all fuel models) representing the fuel compact and an unfueled outer zone representing graphite webbing. A full complement of sorption isotherm data must be given for both surfaces. The iterative solution strategy and the specialized nodal difference equations are explained in Appendix A.

On user input record COR_DIFFM8 when inputting sorption isotherm data for gas gap interface condition, the inner surface uses constants a0, a1, a2, a3, ctra1, and ctra2 to represent sorption isotherm model constants A, B, D, E, F, G (previous general sorption isotherm model description). The outer surface uses constants b0, b1, b2, b3, ctrb1, and ctrb2 to represent sorption isotherm model constants A, B, D, E, F, G (previous general sorption isotherm model description).

9.1.1.5.3 Coolant Boundary Treatment in Diffusion Solution

A coolant boundary model is meant to treat fission product transport to a cooling gas from a surface accounting for boundary layer effects (kinetic resistance to advection of the sorbate). This occurs, for example, in a PMR where graphite webbing interfaces to helium in coolant holes or in a PBR where the unfueled pebble matrix outer surface interfaces to helium. A sorption isotherm empirical model is applied for the cooled surface, and the diffusion equation of the node at the surface is modified. The key physical assumption is again vapor pressure equilibrium, but now between the pressures implied by (1) the sorbate concentration at the surface, and (2) the concentration in the boundary layer near the surface. Thus, for a given fission product species, the sorbate concentration at the cooled surface generates a vapor pressure that is equal to the vapor pressure in the boundary layer implied from an ideal gas assumption. Since the sorption isotherm empirical model describes the vapor pressure at the cooled surface, a nonlinear relationship is implied between the sorbate concentration at the surface and the concentration in the near-surface boundary layer. This condition is then effectively imposed as a nonlinear boundary condition, which requires linearization and iteration to solve in the context of the diffusion solution. Note this strategy resembles that of the TRAFIC code that is meant to model release of metallic fission products from an HTGR core. The key difference in MELCOR is that a finite volume solution of the diffusion problem is used as opposed to alternative methods (e.g. of TRAFIC). Thus, the specific nodal difference equation (near the cooled surface) must be derived.

In the MELCOR diffusion solution, a coolant boundary interface may exist in a model at the end of the outermost zone (periphery of the given model) such that the outermost node on the outermost zone has a modified nodal equation. The intent is that the coolant boundary interface would be used in (1) a PMR type reactor in the matrix model in the unfueled outer zone representing graphite webbing, or (2) a PBR type reactor in the matrix model in the unfueled outer zone representing the outer region of a pebble. A full complement of sorption isotherm data must be given for the cooled surface. The iterative solution strategy and the specialized nodal difference equations are explained in Appendix A.

On user input record COR_DIFFM8 when inputting sorption isotherm data for gas gap interface condition, the inner surface uses constants a0, a1, a2, a3, ctra1, and ctra2 to represent sorption isotherm model constants A, B, D, E, F, G (previous general sorption isotherm model description)

9.1.1.6 Partition Coefficient Model

In the present context of fission product species diffusion, a partition coefficient describes the “jump” in concentration seen at/across an interface – usually a solid/gas interface. It is a less-complicated alternative to the sorption isotherm empirical model for capturing fission product species evaporation/transport across a gas gap or to coolant. The formulation consists of a prescribed jump condition on fission product species concentration at a given interface. The specialized nodal difference equations are explained in 0.

9.1.2 Steady-State Diffusion Solution Methodology

The steady-state diffusion stage is the first and most complicated step of an HTGR radionuclide release calculation in MELCOR. It is meant to predict pre-transient fuel radionuclide distributions and release rates. The distributions and release rates follow from dimensional molar concentration profiles generated for each fuel model and for each tracked fission product species. The concentration profiles are computed from a two-loop iterative method that employs:

- an inner loop bisection algorithm on molar volumetric generation rates, and
- a two-step outer loop that approximates the fuel matrix radionuclide inventory

The intended outcome of the two-tiered iterative solution method is a set of concentration profiles that agree with the radionuclide inventory targets implied by DCH/RN1 user input.

9.1.2.1 Goal of the Steady-State Diffusion Solution

The aim of the steady-state stage of the diffusion solution is to generate molar concentration profiles according to a steady-state diffusion treatment in each COR cell and in each TRISO and matrix model for each tracked fission product species as depicted in Figure 9.3. These concentration profiles are assumed to reflect end-of-burnup

conditions in the TRISO particles and in the fuel elements. They are computed without a priori knowledge of the end-of-burnup matrix and coolant inventories. Steady concentration profiles should reflect the combined effects of:

- Generation due to fission
- Loss due to radioactive decay
- Diffusion (and its temperature dependence)
- Leakage/release (model-wise transfers)
    - Transfer between models (e.g. intact TRISO diffusional release to matrix)
    - Transfer from matrix model to coolant (implied by boundary diffusion flux)
    - Recoil and contamination if applicable

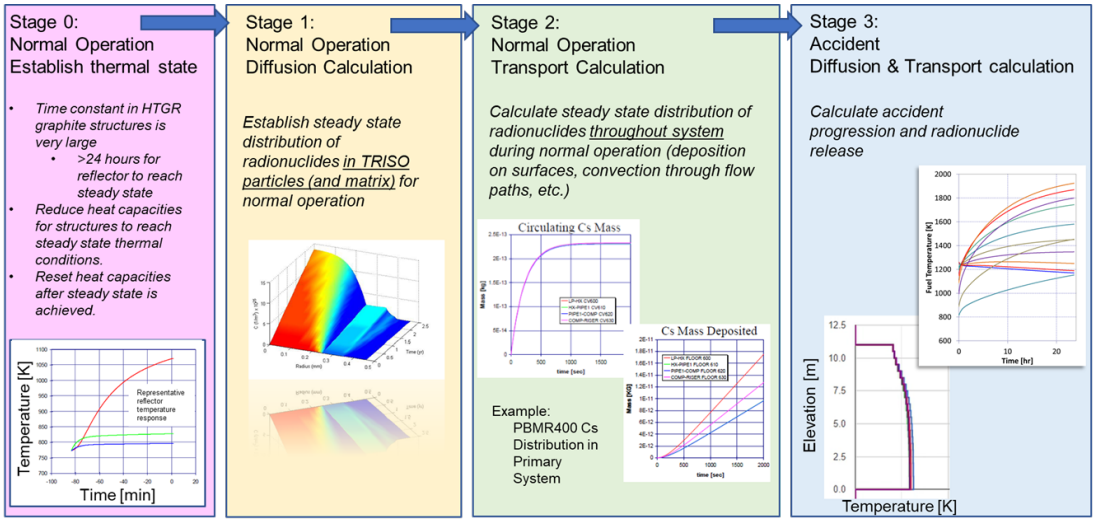


Figure 9.3 Steady-state through transient accident analysis methodology

Known quantities when computing steady concentration profiles include:

- Model noding/geometry
- Model material diffusivities (with temperature dependency)
- Temperatures (COR package plus energy/temperature models)
- TRISO model population fractions
- Decay constants
- DCH/RN1/COR inputs
    - DCH/RN1 radionuclide class configurations
    - DCH total masses (by radionuclide class)
    - RN1 distribution to COR components across all COR cells (by class)
- Model boundary conditions (mostly user-supplied)

Unknown quantities other than molar concentration include:

- Volumetric molar generation rates for tracked fission product species
- Matrix radionuclide inventories
- Release/transfer rates between models and between matrix and coolant

The molar volumetric generation rate is the β term from the general diffusion equation and is presumably nonzero in regions where either:

- fission occurs, or
- recoil deposits fission fragments, or
- contamination (tramp Uranium) deposits fission fragments, or
- TRISO particle release (leakage) occurs to the matrix

Values of β are not known but could be resolved via an iterative bisection method that targets the RN1 class inventories known (by COR cell) from DCH/RN1/COR input.

9.1.2.2 Two-Loop Bisection Algorithm

The steady-state diffusion calculation entails the difficulty that molar volumetric generation rates for source terms of the diffusion equation are not directly known a priori. Diffusion coefficient data, geometry, boundary conditions, burnup time, TRISO model population fractions, COR structural temperatures, and end-of-burnup mass inventories are known for each radionuclide class by COR cell, but this information does not readily translate into a set of volumetric generation rates for use in the steady state equations.

However, if volumetric generation rates were known or assumed, concentration profiles could be computed from steady-state diffusion equations. Those concentration profiles could then be aggregated over all fuel models and all fuel volumes and compared to known end-of-burnup inventories on a COR cell-by-cell basis. Based on any mismatch, improved guesses at molar volumetric generation rate can be made and the process can be repeated to convergence in a bisection algorithm. This suggests that an iterative bisection approach could allow for computation of a self-consistent set of molar volumetric generation rates. Self-consistent here refers to the condition that the final set of molar volumetric generation rates leads to concentration profiles that match user-specified end-of-burnup radionuclide inventories.

One outstanding difficulty remains in that there is no known end-of-burnup matrix inventory because such information is not a requirement of user input. Nevertheless, the matrix model necessarily comprises some nonzero portion of COR cell end-of-burnup radionuclide inventory if indeed any radionuclide mass escapes the TRISO particle population or if any matrix contamination is allowed. Some method for predicting fuel matrix radionuclide inventory is required unless the non-conservative assumptions are made that:

- no TRISO release occurs by diffusion under irradiation, and
- no recoil or contamination effects facilitate release

User input implies COR cell radionuclide class mass inventory and gives TRISO model population fractions for the steady-state diffusion calculation. Total COR cell mass inventory (for a given radionuclide class) must equal the sum of model-wise inventories, i.e. the sum of all inventories across TRISO models and the matrix model, because these

together represent the totality of the fuel. Model-wise inventories set targets for the bisection algorithm, but again there is no fixed target for the matrix model inventory. In MELCOR, a two-step process is used to fix a matrix model inventory target as illustrated in Figure 9.4. A two-step outer iteration wraps the inner bisection on molar volumetric generation rate. In the first outer iteration, the inner loop approximates all TRISO model concentration profiles and uses them to surmise TRISO model release. This generates an effective molar volumetric source term for the diffusion solution and therefore a nonzero molar inventory in the matrix model. The matrix inventory is counted and brought to bear on all TRISO model inventory targets for the second outer iteration.

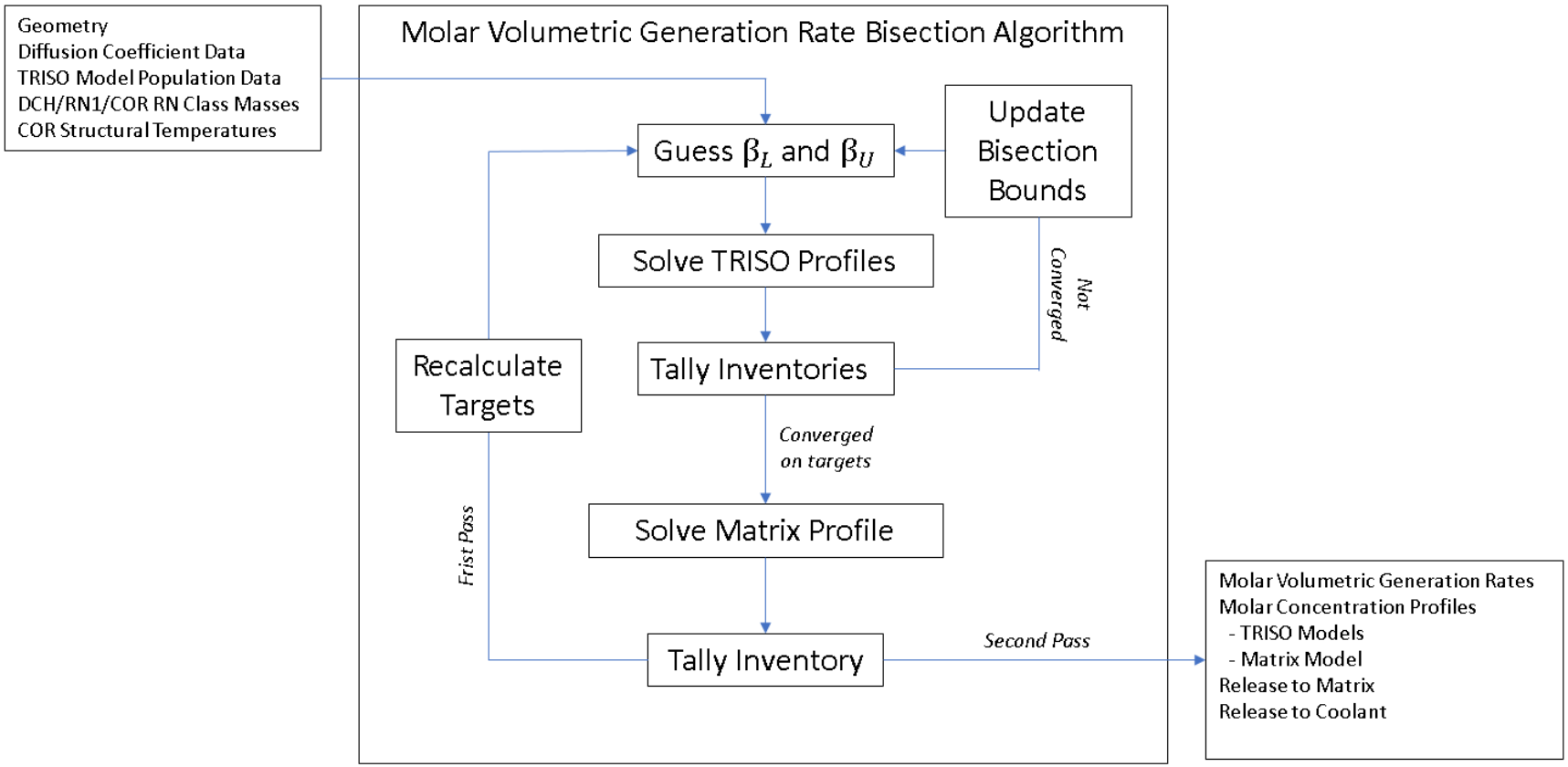


Figure 9.4 Generalized computational grid for diffusion equation solution

9.1.2.3 Fission Product Generation in TRISO Models

TRISO models are configured by the user as COR input. Each TRISO model facilitates radionuclide release calculations for some cohort of the overall TRISO population. The kernel zone in a TRISO model is permitted a nonzero molar volumetric generation rate (source term) from fission exclusively. The buffer zone in a TRISO model is permitted a nonzero molar volumetric generation rate (source term) from fission product recoil exclusively. Other zones defined in a TRISO model generally have no molar volumetric source terms and receive molar inventory exclusively by diffusion from some other zone. The molar volumetric generation rates due to fission are identically the bisection variables referenced in the description of the steady-state diffusion solution algorithm. These quantities are implied by user input and are calculated iteratively. These quantities also link to fission product recoil calculations within TRISO models.

Fission product recoil is a phenomenon whereby fission fragments produced near the periphery of the TRISO fuel kernel escape, by virtue of their kinetic energy, to the surrounding buffer (assuming an "intact" TRISO model where the buffer is present). Thus, the buffer zone radially adjacent to (surrounding or outside of) a kernel zone can receive a fraction of the molar volumetric generation that otherwise would have belonged to the kernel zone. The molar volumetric generation rate due to recoil in a buffer zone is computed from the molar volumetric generation rate in a kernel zone according to a user-defined recoil fraction and a kernel-to-buffer volume fraction:

$$\beta_{m,b,j} = \left[\frac{\beta_{m,k,j}}{1 - f_{rec}}\right] [f_{rec}] \left[{V_{m,k}}/{V_{m,b}}\right] \tag{9-12}$$

Where:

$\beta_{m,k,j}$ = Species *j* generation rate in kernel, model *m* [kmol/m^3_kernel/s]

$\beta_{m,b,j}$ = Species j generation rate in buffer, model *m* [kmol/m^3_buffer/s]

f_{rec} = User-defined recoil fraction [-]

$V_{m,k}$ = Kernel volume, model *m* [m^3_kernel]

$V_{m,b}$ = Buffer volume, model *m* [m^3_buffer]

Note that the kernel-to-buffer volume fraction transforms the buffer volumetric generation rate to a per unit buffer volume basis. For the "failed" TRISO model, MELCOR assumes that only a kernel zone exists, i.e. the failed TRISO particle is treated as a bare fuel kernel for purposes of diffusional release calculations. Fission product recoil is still allowed, but the destination of recoiling fission fragments is the matrix model instead of the non-existent buffer zone.

9.1.2.4 Fission Product Generation in the Matrix Model

The matrix model has nonzero molar volumetric generation rates in its kernel zone that generally consists of three components:

- Aggregated TRISO model leakage (release)
- Recoil from the “failed” TRISO model
- Contamination (fission of tramp material occurring in matrix)

The overall molar volumetric generation is the sum of these components provided that each component is translated into the proper per unit matrix volume basis.

9.1.2.4.1 TRISO Model Leakage/Release

All TRISO models are assumed to reside within the kernel zone of the matrix model. When TRISO model concentration profiles are solved, concentration gradients imply some diffusion flux (or leakage, or release) at the model boundaries. This leakage represents a source of radionuclide inventory to the fuel element matrix and is generally calculated as:

$$\beta_{m\_to\_mMatrix,k,j} = \left(\sum_{m=1}^{M}\left(R_{m,j}\right)\right) * \left(\frac{N_p}{V_{mMatrix,k}}\right) \quad (9\text{-}13)$$

Where:

$\beta_{m\_to\_mMatrix,k,j}$ = Species *j* generation rate in matrix kernel due to release from model *m* [kmol/m^3_kernel/s]

$R_{m,j}$ = Species *j* release rate from model *m* [kmol/s]

N_p = Total number of TRISO particles per single fuel element

$V_{mMatrix,k}$ = Kernel volume in matrix model, single fuel element [m^3_kernel]

The release rate from each model is computed from an approximate expression for the model boundary diffusion flux using the known TRISO model concentration profile. Note this is the case for all TRISO models – including the “failed” model – in the steady-state diffusion stage. The bare kernel of the “failed” model is subjected to the finite volume diffusion solution, a concentration profile is derived, and a boundary diffusion flux is calculated. This is not the case for the “failed” model during the transient diffusion and transport stage as a different treatment is applied.

Generally, TRISO model molar release rate is computed as:

$$R_{m,j} = \overline{D_N}\left(\frac{C_N - C_{N-1}}{\Delta r_N}\right) A_N \qquad (9\text{-}14)$$

Where:

$R_{m,j}$ = Species *j* generation rate in kernel, model *m* [kmol/s]

N = Outermost node number for TRISO model *m* finite volume grid [-]

$\overline{D_N}$ = Interface N diffusion coefficient [m^2/s]

C_i = Node *i* molar concentration [kmol/m^3]

Δr_N = Distance between nodes *N* and *N-1* [m]

A_N = Node N interfacial area, model *m* [m^2]

The molar concentrations for TRISO models are computed on a per particle basis according to molar volumetric generation rates that account for population fraction. Therefore, the molar releases need only be multiplied by the total number of TRISO particles per fuel element to ascertain the total model-wise release to a single fuel element. The matrix model diffusion is likewise solved on a per fuel element basis, so the term $\beta_{m\_to\_mMatrix,k,j}$ reflects the volumetric generation in the matrix kernel of a single fuel unit due to release from type *m* TRISO particles in that fuel unit.

9.1.2.4.2 Recoil from Failed TRISO Model

Recoil from the bare kernel of a "failed" TRISO model is sourced directly into matrix. Mathematically, this model resembles that of fission product recoil between kernel and buffer of other TRISO models. The molar volumetric generation rate due to fission is split – according to recoil fraction and relative volume fraction – between the "failed" TRISO model kernel and the matrix model kernel. Accordingly:

$$\beta_{mMatrix\_recoil,k,j} = \left[\frac{\beta_{mFail,k,j}}{1 - f_{rec}}\right][f_{rec}]\left[N_p V_{mFail,k} \Big/ V_{mMatrix,k}\right] \qquad (9\text{-}15)$$

Where:

$\beta_{mMatrix\_recoil,k,j}$ = Species *j* generation rate in kernel, model *m=mMatrix* due to recoil [kmol/m^3_kernel/s]

$\beta_{mFail,k,j}$ = Species j generation rate in buffer, model *m=mFail* [kmol/m^3_kernel/s]

$V_{mFail,k}$ = Kernel volume, model *m=mFail* [m^3_kernel]

$V_{mMatrix,k}$ = Kernel volume, model *m=mMatrix* [m^3_kernel]

9.1.2.4.3 Contamination

The user can specify a contamination fraction to account for the possibility that fuel element matrix – the carbonaceous material that hosts embedded TRISO particles – is contaminated with fissile or fissionable material that will generate fission products under irradiation and thereby deposit radionuclide inventory into the matrix directly. According to the implementation, a fraction of each TRISO model's fission generation rate is allowed to occur in the matrix (without assuming that fraction is debited from TRISO):

$$\beta_{mMatrix\_contam,k,j} = \sum_{m=1}^{M} \left(\left[\frac{\beta_{m,k,j}}{1 - f_{rec}} \right] [f_{con}] \left[{N_p V_{m,k}} \Big/ {V_{mMatrix,k}} \right] \right) \qquad (9\text{-}16)$$

Where:

$\beta_{mMatrix\_contam,k,j}$ = Species *j* generation rate in kernel, model *m=mMatrix* due to contamination [kmol/m^3_kernel/s]

$\beta_{m,k,j}$ = Species j generation rate in kernel, model *m* [kmol/m^3_kernel/s]

$V_{m,k}$ = Kernel volume, model *m* [m^3_kernel]

$V_{mMatrix,k}$ = Kernel volume, model *m=mMatrix* [m^3_kernel]

9.2 Steady-State Transport Solution Method

Once steady-state radionuclide distributions and COR component releases are known, they are held constant while CVH, HS, and RN1 march through problem time. The goal of the steady-state transport stage is a set of established trends in CVH/RN1 inventory and HS/RN1 deposition rates which can be used to surmise an end-of-burnup CVH/HS/RN1 condition.

As the calculation marches through problem time, CVH, HS, and RN1 are basically establishing steady trends in CVH inventory change and HS deposition rates. The problem time required to establish steady CVH/HS/RN1 distributions likely will not be equivalent to the burn-up time, or the time for which fuel was under steady irradiation prior to the transient/accident. To account for this difference, a scaling is applied whereby the CV contents, HS surface depositions, etc. are multiplied to account for the true time for which the system experiences steady COR component radionuclide release.

The user specifies the steady-state transport problem time duration indirectly by specifying the time at which steady-state diffusion is done and the time at which transient diffusion and transport begins. The user should verify that appropriate CVH/HS/RN1 trends exist before scaling and commencement of the transient diffusion/transport stage.

9.3 Transient Diffusion/Transport Solution Method

Once steady-state radionuclide distributions and COR component releases are known, and once properly scaled steady-state CVH/HS/RN1 distributions and inventories are known, the initial conditions of a transient/accident sequence are fully known. The fission product release model switches to a transient mode such that:

- TRISO models (except for the failed model) and the matrix model is treated with a transient form of the diffusion equation
- Fission, recoil, and contamination source terms (molar volumetric generation rates) are universally zeroed out
- Failed TRISO model is optionally treated according to an analytic release model
- User-specified TRISO model inventory dynamics are imposed (e.g. by CF)
- COR component inventories are subject to decline as release occurs
- CVH/HS/RN1 packages perform their typical radionuclide transport functions

Release from TRISO models (other than failed) to matrix can still occur according to calculated boundary diffusion fluxes, but fuel degradation/failure – i.e. release from failed TRISO particles - will play a dominant role in radionuclide release for any given accident/transient. There are built-in provisions for predicting fuel failure, but the user has flexibility with control functions in this regard.

9.4 Analytic Release Model for Failed TRISO

To compute a total release from failed TRISO particles, one must properly account for

- ongoing release from currently failed TRISO particles, plus
- release from recently failed TRISO particles (i.e. since last time-step) to include:
    - kernel release (according to modified Booth model diffusion)
    - burst release from outlying zones of the previously intact TRISO particle

Using data to include:

- the intact TRISO particle failure rate
- fractional release from failed TRISO particles up to the present time, and
- failed TRISO time history - previous failures and diffusion/temperature history

User inputs or a temperature-dependent, empirical failure curve determines the current intact TRISO particle failure rate. With respect to fractional release rate from failed TRISO particles, a short-term approximation is made to a diffusion solution on an equivalent

sphere (a so-called modified Booth approach) that accounts for both irradiation and time-dependent diffusion coefficients.

9.4.1 TRISO Fuel Failure and Inventory Dynamics

The user specifies initial TRISO model population fractions and determines how those fractions evolve over the course of an accident/transient. Control functions are powerful tools for this purpose and are the only alternatives for TRISO inventory management other than either:

- assuming all TRISO model population fractions are constant, or
- employing the built-in failure model that governs transition from intact TRISO to failed TRISO

The temperature-dependent failure model is a simple curve-fit based on German data. It correlates TRISO population failure fraction as a function of temperature according to:

$$f_{fail} = (2.28109 * 10^{-7}) e^{(0.00498*(T_{FU}-273.0))} \tag{9-17}$$

The failure fraction is applied to the currently available intact TRISO inventory to ascertain a gross inventory of failure and a failure rate.

9.4.2 Fractional Release

The fractional release of a tracked fission product species up to time t for an equivalent sphere after irradiation (no fission power, no further nuclide generation) is, according to the modified Booth approach:

$$F(t) = 6 \sum_{n=1}^{\infty} \int_0^t exp\left[-\left(\left(n^2 \pi^2 \int_0^{\mu} D' t' dt' \right) + \lambda \mu \right) \right] D'(\mu) d\mu \tag{9-18}$$

Where:

$F(t)$ = Fractional release of fission product up to time t [-]

$D'(t)$ = Reduced diffusion coefficient [m^2/s] = $D(t)/a^2$

a = Radius of equivalent sphere [m], taken as fuel kernel radius here

In the short term ($\pi^2 D' t \leq 1$) and for a constant reduced diffusion coefficient, the fractional release expression simplifies to:

$$F(t) = \frac{3D'}{\lambda}\left(e^{-\lambda t} - 1 + \sqrt{\frac{\lambda}{D'}} erf(\lambda t) \right) \quad (9\text{-}19)$$

Where:

D' = Constant reduced diffusion coefficient [m^2/s]

λ = Decay constant [1/s]

A further simplification proposed by Gelbard [79] is utilized to compute fractional release:

$$F(t) = \begin{cases} 1.0006964\left(\left(\frac{36}{\pi}\right)D't - 3D't\right) & if\ D't < 0.155 \\ 1 - \left(\frac{6}{\pi^2}\right)e^{(-\pi^2 D't)} & if\ D't \geq 0.1551 \end{cases} \quad (9\text{-}20)$$

9.4.3 Total Release

Releases from failed TRISO particles are an integrated result of particle history. Assuming a failure fraction $F_W(t)$ at time t, a failure rate $\frac{dF_W}{dt}$ can be computed. This is the effective rate at which freshly failed TRISO is appearing. Assuming the fission product release fraction from a failed TRISO particle at time t due to failure at a previous time $\tau < t$ is $F_R(t-\tau)$, then the total release fraction at time t due to all failures at times τ between 0 and t is computed according to a convolution integral:

$$F_{tot}(t) = \int_0^t \frac{dF_W(t)}{d\tau} F_R(t-\tau) d\tau \quad (9\text{-}21)$$

The calculation of total release fraction from failed TRISO can be simplified from a general convolution integral to a simpler product of the failed TRISO kernel release fraction and the failure fraction imposed on intact TRISO particles. Buffer (and outlying zone) release from just-failed TRISO particles – envisioned as burst release coincident with particle failure (SiC layer failure) – can then be added to the kernel release from failed TRISO particles.

In MELCOR, the calculation of total particle release from failed TRISO is performed via a sum over historical failed TRISO kernel failure/release using data stored at certain “save points” in problem time history. This approximate time integration of kernel release is:

$$F_{tot}(t)_j = \int_0^t \frac{dF_W(t)}{d\tau} F_{R,j}(t-\tau)d\tau \approx \sum_{nc=1}^{ncsave} \left[\left(\frac{dF_W}{dt}\right)(F_{R,j})(\Delta t)X_{k,int,j}\right]_{nc} \quad (9\text{-}22)$$

Where:

$ncsave$ = Number of save points spanning past to present problem time

$\frac{dF_W}{d\tau}$ = Failure rate, approximated from data for a given save point nc [1/s]

$F_{R,j}$ = Release rate, species j, for save point nc [-]

Δt = Timestep between save points (present minus last save point time) [s]

$X_{k,int,j}$ = Intact kernel inventory, species j [kmol]

The failure rate is computed from user input data (or with assistance from the built-in failure model) according to the approximation:

$$\frac{dF_W}{dt} \sim \frac{\Delta f_f}{\Delta t} \quad (9\text{-}23)$$

Where:

Δf_f = Change in TRISO failure fraction, from current and last save point

Δt = Change in time between current and last save point [s]

The species j release rate is computed from the modified Booth approximation as previously presented in equation (9-4) where now the average reduced diffusion coefficient, $\overline{D'}$, is given by:

$$\overline{D'} = \frac{\sum_{ns=n1}^{ncsave}\left[\left(t_0(nc) - t_0(nc-1)\right)\left(D_0 e^{-Q/T_{nc}}\right)\right]_{ns} + \left(t - t_0(ncsave)\right)D_0 e^{-Q/T_{ncsave}}}{r^2\left(t - t_0(n1)\right)} \quad (9\text{-}24)$$

Where:

$n1$ = nc , from current nc of computed kernel release summation

$t_0(n)$ = Problem time associated with save point index n [s]

D_0 = Pre-coefficient for current fission product species [m2/s]

Q = Activation energy for current fission product species [J/mol]

T_{nc} = Kernel (COR component FU) temperature, save point nc [K]

t = Current problem time [s]

r = Radius of equivalent sphere from reduced coefficient definition [m]

9.5 Energy/Temperature Models for Fuel Elements

For calculating temperature profiles in a TRISO or matrix model, assumptions include:

- Known COR structural (FU, MX component) temperatures
- One-dimensional, radial, steady conduction with (conditional) energy generation
- For TRISO models:
    - Spherical geometry only
    - Conventional TRISO layers
        - An inner fuel-bearing zone with constant heat generation
        - Outer zones without heat generation (for intact TRISO)
    - The outermost zone's outermost surface temperature (temperature of the TRISO particle "surface") is the COR FU component temperature
- For the matrix model:
    - PBR reactor type (fuel pebbles):
        - Spherical geometry
        - As shown in Figure 9.5, there are two pebble configurations:
            1. Conventional pebble
                - An innermost fuel-bearing zone (COR FU component)
                    - $r \leq R_{f,o}$ for $R_{f,o}$ the radius of the fueled region
                - An outer unfueled zone (COR MX component)
                    - $R_{f,o} \leq r \leq R_{PB}$ for R_{PB} the pebble radius
                - Heat generation in inner zone but not outer zone
                - Average of the inner zone temperature profile is TFU
                - Average of the outer zone temperature profile is TMX
                - Symmetry condition at pebble center
                - Heat flux continuity condition at the zonal interface
            2. Fueled shell pebble
                - An innermost unfueled zone (COR FUXM)
                    - $r \leq R_{f,i}$ for $R_{f,i}$ the inner radius of the fueled zone (outer radius of central unfueled zone)
                - A middle, fueled, spherical shell zone (COR FU)
                    - $R_{f,i} \leq r \leq R_{f,o}$ for $R_{f,o}$ the outer radius of the fueled zone
                - An outer unfueled zone (COR MX)
                    - $R_{f,o} \leq r \leq R_{PB}$ for R_{PB} the pebble radius

- No heat generation in either the inner or outer zones
- Heat generation in fueled shell
- Average across the inner and fueled shell zone temperature profiles – taken together - is TFU
- Average of the outer zone temperature profile is TMX
- Continuity conditions at zonal interfaces

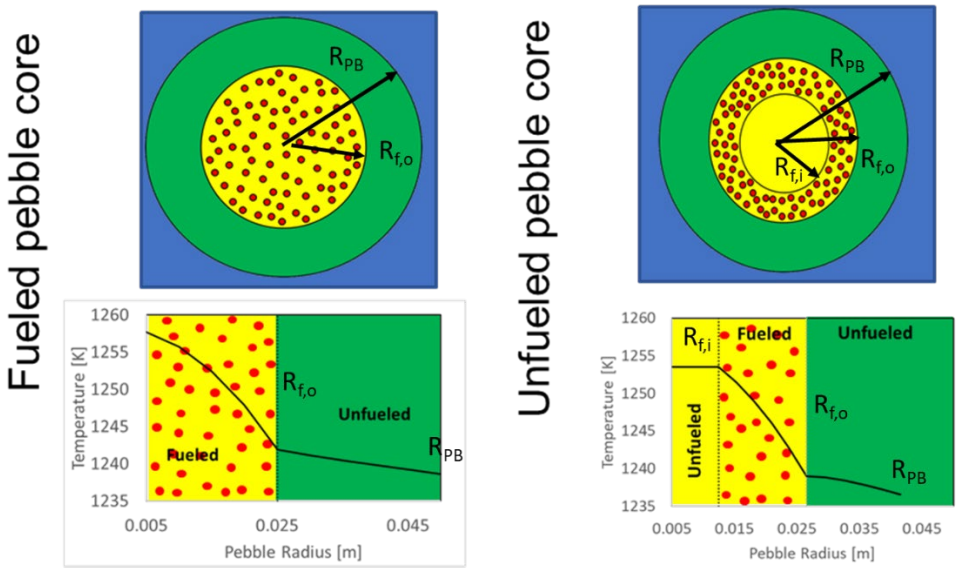


Figure 9.5 Temperature profiles in pebbles.

- PMR reactor type (fuel compacts)
  - Cylindrical geometry
  - Inner fueled zone (COR FU),
  - Outer unfueled zone (COR MX)
  - Heat generation in inner zone, no heat generation in outer zone
  - Average of the inner zone temperature profile equals TFU
  - Average of the outer zone temperature profile equals TMX
  - Symmetry condition at fuel compact center
  - Continuity conditions at zonal interfaces

The radial (one-dimensional), steady conduction equations for cylindrical and spherical geometries are:

$$k\frac{d}{dr}\left(r\frac{dT}{dr}\right)+r\dot{q}=0, \qquad cylindrical \tag{9-25}$$

$$k\frac{d}{dr}\left(r^2\frac{dT}{dr}\right)+r^2\dot{q}=0, \qquad spherical \tag{9-26}$$

Where:

r = Radial coordinate/dimension [m]

k = Thermal conductivity [W/m/K]

T = Temperature [K]

$\dot{q}$ = Constant energy generation rate [W/m^3]

In the presence of constant heat generation $(\dot{q} \neq 0)$, the general profiles are:

$$T(r)=A\,ln(r)+B-\frac{\dot{q}r^2}{4k}, \qquad cylindrical \tag{9-27}$$

$$T(r)=\frac{A}{r}+B-\frac{\dot{q}r^2}{6k}, \qquad spherical \tag{9-28}$$

In the absence of constant heat generation $(\dot{q}=0)$, the general profiles are:

$$T(r)=A\,ln(r)+B\,, \qquad cylindrical \tag{9-29}$$

$$T(r)=\frac{A}{r}+B, \qquad spherical \tag{9-30}$$

Where constants A and B are resolved by imposing boundary conditions.

For purposes of energy/temperature solutions on cylindrical and spherical geometries, averages of zone-wise temperature profiles can be computed as:

$$\bar{T}(R_i<r<R_o)=\frac{\int_0^{2\pi}\int_{R_i}^{R_o}\big(T(R_i<r<R_o)\big)r\,dr\,d\theta}{(\pi)(R_o^2-R_i^2)}, \qquad cylindrical \tag{9-31}$$

$$\bar{T}(R_i<r<R_o)=\frac{\int_0^{2\pi}\int_0^{\pi}\int_{R_i}^{R_o}\big(T(R_i<r<R_o)\big)r^2\sin\theta\,dr\,d\theta\,d\phi}{\left(\frac{4\pi}{3}\right)(R_o^3-R_i^3)}, \qquad spherical \tag{9-32}$$

Where:

R_i = An inner radial boundary, e.g. of a zone [m]

R_o = An outer radial boundary, e.g. of a zone [m]

θ = Azimuth angle coordinate [rad]

Φ = Zenith angle coordinate [rad]

Profile shapes and averages are used extensively in the derivation of temperature profiles for TRISO particles, PBR pebbles, and PMR compacts in hexagonal blocks.

9.5.1 TRISO Particle Temperature Distribution

For the generalized mathematical description of a non-failed TRISO particle temperature distribution, a sphere consisting of N layers - the first and inner-most of which has energy generation – is conceptualized as:

$$\begin{aligned} &k\frac{d}{dr}\left(r^2\frac{dT}{dr}\right)+r^2\dot{q}=0,\ \ r\le\ R_{f,o} && (n=1)\\ &\frac{d}{dr}\left(r^2\frac{dT}{dr}\right)=0,\ \ R_{f,o}\le r<\ R_{f2,o} && (n=2)\\ &\ldots\\ &\frac{d}{dr}\left(r^2\frac{dT}{dr}\right)=0, \qquad R_{f(N-1),o}\le\ r<R_{fN,o} && (n=N) \end{aligned} \tag{9-33}$$

Accordingly, the general temperature and heat flux distributions are:

$$T(r)=\begin{cases} \dfrac{C_1}{r}+C_2-\dfrac{\dot{q}r^2}{6k}, & r\le R_{f,o} && (n=1)\\ \dfrac{C_3}{r}+C_4, & R_{f,o}\le r<R_{f2,o} && (n=2)\\ & \ldots & & \\ \dfrac{C_{2n-1}}{r}+C_{2n}, & R_{f(n-1),o}\le r<R_{fn,o} && (n)\\ & \ldots & & \\ \dfrac{C_{2N-1}}{r}+C_{2N}, & R_{f(N-1),o}\le r<R_{fN,o} && (n=N) \end{cases} \tag{9-34}$$

$$q''(r) = \begin{cases} \frac{k_1 C_1}{r^2} + \frac{\dot{q} r}{3}, & r \le R_{f,o} \quad (n=1) \\ \frac{k_2 C_3}{r^2}, & R_{f,o} \le r < R_{f2,o} \quad (n=2) \\ \cdots & \\ \frac{k_n C_{2n-1}}{r^2}, & R_{f(n-1),o} \le r < R_{fn,o} \quad (n) \\ \cdots & \\ \frac{k_N C_{2N-1}}{r^2}, & R_{f(N-1),o} \le r < R_{fN,o} \quad (n=N) \end{cases} \tag{9-35}$$

By enforcing a symmetry condition, a boundary temperature condition, and interface conditions on temperature and heat flux, an algorithm can be devised whereby the general set of integration constants is solved to obtain $T(r)$. By symmetry:

$$q''(r=0) = 0 \tag{9-36}$$

The FU component temperature is assumed at the TRISO particle boundary:

$$T(r = R_N) = T_{FU} \tag{9-37}$$

The interface conditions are:

$$T\left(R_{fn,o}^{-}\right) = T\left(R_{fn,o}^{+}\right), n > 1 \tag{9-38}$$

$$q''\left(R_{fn,o}^{-}\right) = q''\left(R_{fn,o}^{+}\right), n \ge 1 \tag{9-39}$$

Application of these conditions leads to a set of constants:

$$C_1 = 0$$

$$C_2 = \left(\dot{q} R_{f1,o}^2\right)\left({}^{1}/_{3k_2} + {}^{1}/_{6k_1}\right) + C_4$$

$$C_3 = \frac{\dot{q} R_{f1,o}^3}{3k_2} \tag{9-40}$$

...

$$C_{2n-1} = C_{2n+1}\left({}^{k_{n+1}}/_{k_n}\right)$$

$$C_{2n} = \left({}^{1}/_{R_{fn,o}}\right)(C_{2n+1} - C_{2n-1}) + C_{2n+2}$$

…

$$C_{2N} = T_{FU} - {}^{C_{2N-1}}\Big/_{R_{fN,o}}$$

The form of these equations suggests the following solution algorithm:

- Immediately obtain C_1 and C_3 from the symmetry boundary condition and from knowledge of the first zone (n=1, the kernel zone) and the second zone
- Beginning with known C_3, solve for the odd-numbered constants in ascending order up to the next-to-last constant C_{2N-1} using the prescription for C_{2n-1}
- Solve for constant C_{2N}
- Beginning with known C_{2N}, solve for the even-numbered constants in descending order down to the second constant C_2 using the prescription fo C_{2n}
- Inform the temperature profile with the full complement of integration constants

9.5.2 TRISO Particle Volumetric Heat Generation Rate

Note that a volumetric heat generation rate is required to ascertain the temperature profile in a TRISO particle. An appropriate value can be calculated from COR:

- Fission and decay power in a COR/diffusion cell
- Fuel mass in a COR/diffusion cell (strictly FU primary material)
- TRISO kernel volume (single particle)
- Number of TRISO particle (kernels) within a fuel unit (e.g. a pebble or compact)
- Fuel mass per fuel unit in COR/diffusion cell

$$\dot{Q} = {}^{M_{FU}^{unit}\left(P_f + P_d\right)}\Big/_{N_p^{unit} V_k} \qquad (9\text{-}41)$$

Where:

$\dot{Q}$ = Volumetric heat generation rate in TRISO kernel [W/m^3]

M_{FU}^{unit} = Fuel mass in a fuel unit (pebble or compact) [kg]

P_f = Fission power per unit fuel mass [W/kg]

P_d = Decay power per unit fuel mass [W/kg]

V_k = Kernel volume, single TRISO particle [m^3]

N_p^{unit} = TRISO particles per fuel unit (pebble or compact) [-]

9.5.3 PBR Pebble (Conventional) Temperature Distribution

For the two zones allowed in the conventional PBR pebble matrix model, the temperature distributions can be generally written in terms of unknown constants:

$$T(r) = \begin{cases} \frac{C_1}{r} + C_2 - \frac{\dot{q}r^2}{6k_{FU}}, & r \leq R_{FU} \\ \frac{C_3}{r} + C_4, & R_{FU} < r \leq R_{pb} \end{cases} \quad (9\text{-}42)$$

Heat flux distributions can be generally written as:

$$q''(r) = \begin{cases} \frac{k_{FU}C_1}{r^2} + \frac{\dot{q}r}{3}, & r \leq R_{FU} \\ \frac{k_{MX}C_3}{r^2}, & R_{FU} \leq r \leq R_{pb} \end{cases} \quad (9\text{-}43)$$

Four conditions are required to resolve the four unknown constants of integration C_1, C_2 ,C_3, and C_4. By fixing the average value of the temperature distribution in each zone, two conditions are obtained:

$$\bar{T}(r) = \begin{cases} T_{FU} = \left(\frac{3}{4\pi R_{FU}^3}\right)\left(\int_0^{2\pi}\int_0^{\pi}\int_0^{R_{FU}} \left(\frac{C_1}{r} + C_2 - \frac{\dot{q}r^2}{6k_{FU}}\right) r^2 \sin\theta dr d\theta d\phi\right) \\ T_{MX} = \left(\frac{3}{4\pi\left(R_{pb}^3 - R_{FU}^3\right)}\right)\left(\int_0^{2\pi}\int_0^{\pi}\int_{R_{FU}}^{R_{pb}} \left(\frac{C_3}{r} + C_4\right) r^2 \sin\theta dr d\theta d\phi\right) \end{cases} \quad (9\text{-}44)$$

By enforcing heat flux continuity conditions, two other conditions are obtained and the constants of integration are found as:

$$C_1 = 0$$

$$C_2 = T_{FU} + \frac{\dot{q}{R_{FU}}^2}{10k_{FU}} \quad (9\text{-}45)$$

$$C_3 = \frac{\dot{q}{R_{FU}}^3}{3k_{MX}}$$

$$C_4 = T_{MX} - \left[\frac{R_{pb}^2 - R_{FU}^2}{2\left(R_{pb}^3 - R_{FU}^3\right)}\right]\left[\frac{\dot{q}{R_{FU}}^3}{k_{MX}}\right]$$

The constants are used to solve for the pebble surface temperature, $T_{s,pb}$ and the pebble center-line temperature, $T_{CL,FU}$. To resolve the fuel surface temperature, $T_{s,FU}$, at R_{FU}, an average is taken of $T(R_{FU})$ as computed from the inner and outer profile shapes. Then, interpolation according to radial position is used to fill out the full temperature profile at all radial points. The computational algorithm for the temperature profile proceeds as:

- Get $\dot{q}$, k_{FU}, k_{MX}, R_{FU}, and R_{pb}
- Get $T_{s,pb} = T\left(R_{pb}\right) = {C_3}/{R_{pb}} + C_4$
- Get $T_{CL,FU} = T(0) = C_2 - {\dot{q}{R_{FU}}^2}/{10k_{FU}}$
- Set $T_{s,FU} = \left(\frac{1}{2}\right)\left(T(R_{FU}^-) + T(R_{FU}^+)\right) = \left(\frac{1}{2}\right)\left(\left(C_2 - {\dot{q}{R_{FU}}^2}/{6k_{FU}}\right) + \left({C_3}/{R_{pb}} + C_4\right)\right)$

The temperature profile as a function of radial position and known zonal boundary temperatures $T_{CL,FU}$, $T_{s,FU}$, and $T_{s,pb}$ can be rewritten in a form useful for interpolation by radial position:

$$T(r) = \begin{cases} T_{CL,FU} + \left(T_{s,FU} - T_{CL,FU}\right)\left({r^2}/{R_{FU}^2}\right), & r \le R_{FU} \\ T_{s,FU} + \left(T_{s,pb} - T_{s,FU}\right)\left(\dfrac{{1}/{R_{FU}} - {1}/{r}}{{1}/{R_{FU}} - {1}/{R_{pb}}}\right), & R_{FU} < r \le R_{pb} \end{cases} \quad (9\text{-}46)$$

9.5.4 PBR Pebble (Fueled Shell) Temperature Distribution

For the three zones allowed in the fueled shell PBR pebble matrix model, the temperature distributions can be generally written in terms of six unknown constants:

$$T(r)\begin{cases}\dfrac{C_1}{r}+C_2, & 0 < r \le R_{FU,i}\\ C_3+\dfrac{C_4}{r}-\dfrac{\dot{q}r^2}{6k_{FU}}, & R_{FU,i}\le r\le R_{FU,o}\\ \dfrac{C_5}{r}+C_6, & R_{FU,o}\le r\le R_{pb}\end{cases} \tag{9-47}$$

The heat flux distributions can also be written as:

$$q''(r)\begin{cases}\dfrac{k_{FU}C_1}{r^2}, & 0 < r \le R_{FU,i}\\ \dfrac{k_{FU}C_4}{r^2}+\dfrac{\dot{q}r}{3}, & R_{FU,i}\le r\le R_{FU,o}\\ k_{MX}C_5/_{r^2}, & R_{FU,o}\le r\le R_{pb}\end{cases} \tag{9-48}$$

By fixing the average value of the temperature distribution in a certain way, two conditions on the unknown constants may be obtained. The chosen condition on fuel component temperature is that the average over the innermost unfueled zone and the fueled shell zone is set equal to T_{FU}. This is consistent with the assumption that the actual material mass of the unfueled interior zones of PBR pebbles in a COR cell is allotted to the fuel "extra material" (FUX) slot. The matrix component temperature, T_{MX}, is simply set equal to the average temperature of the outermost unfueled zone. These conditions can be written as:

$$T_{FU}=\left(\frac{3}{4\pi\left(R_{FU,o}^3\right)}\right)\left(\int_0^{2\pi}\int_0^{\pi}\int_0^{R_{FU,o}}\left(T(r)\right)r^2\sin\theta\,dr\,d\theta\,d\phi\right)$$

$$=\left(\frac{3}{4\pi\left(R_{FU,o}^3\right)}\right)\left(\int_0^{2\pi}\int_0^{\pi}\left[\int_0^{R_{FU,i}}\left(\frac{C_1}{r}+C_2\right)r^2\sin\theta\,dr\,d\theta\,d\phi+\int_{R_{FU,i}}^{R_{FU,o}}\left(C_3+\frac{C_4}{r}-\frac{\dot{q}r^2}{6k_{FU}}\right)r^2\sin\theta\,dr\,d\theta\,d\phi\right]\right) \tag{9-49}$$

$$T_{MX}=\left(\frac{3}{4\pi\left(R_{pb}^3-R_{FU,o}^3\right)}\right)\left(\int_0^{2\pi}\int_0^{\pi}\int_{R_{FU,o}}^{R_{pb}}\left(\frac{C_5}{r}+C_6\right)r^2\sin\theta\,dr\,d\theta\,d\phi\right) \tag{9-50}$$

Adding to these a symmetry heat flux condition, heat flux continuity conditions, and temperature equality conditions, equations for constants of integration can be found as:

$$C_1 = 0$$

$$C_2 = T_{FU} + \left(\frac{\dot{q}}{2k_{FU}R_{FU,o}^3}\right)\left(\frac{R_{FU,o}^5}{5} + R_{FU,o}^2 R_{FU,i}^3 - \frac{R_{FU,i}^5}{5}\right) - \frac{\dot{q}R_{FU,i}^2}{2k_{FU}}$$

$$C_3 = T_{FU} + \left(\frac{\dot{q}}{2k_{FU}R_{FU,o}^3}\right)\left(\frac{R_{FU,o}^5}{5} + R_{FU,o}^2 R_{FU,i}^3 - \frac{R_{FU,i}^5}{5}\right)$$

$$C_4 = \frac{-\dot{q}R_{FU,i}^3}{3k_{FU}} \qquad (9\text{-}51)$$

$$C_5 = \left(\frac{\dot{q}}{3k_{MX}}\right)\left(R_{FU,o}^3 - R_{FU,i}^3\right)$$

$$C_6 = T_{MX} - \frac{3\left(\frac{\dot{q}}{3k_{MX}}\right)\left(R_{FU,o}^3 - R_{FU,i}^3\right)\left(R_{pb}^2 - R_{f,o}^2\right)}{2\left(R_{pb}^3 - R_{f,o}^3\right)}$$

The constants are used to solve for the pebble surface temperature, $T_{s,pb}$, the pebble center-line temperature, $T_{CL,FU}$, and the two annular fuel shell surface temperatures, $T_{s,FU,i}$ and $T_{s,FU,o}$. Then, interpolation according to radial position is used to fill out the full temperature profile at all radial points. The computational algorithm for the temperature profile proceeds as:

- Get $\dot{q}$, k_{FU}, k_{MX}, $R_{FU,i}$, $R_{FU,o}$, and R_{pb}
- Get $T_{s,pb} = T(R_{pb}) = {C_5}/{R_{pb}} + C_6$
- Get $T_{CL,FU} = T(0) = C_2$
- Get $T_{s,FU,i} = T_{CL,FU} = C_2 = C_3 + {C_4}/{R_{FU,i}} - {\dot{q}R_{FU,i}^2}/{6k_{FU}}$
- Get $T_{s,FU,o} = C_3 + {C_4}/{R_{FU,o}} - {\dot{q}R_{FU,o}^2}/{6k_{FU}} = {C_5}/{R_{FU,o}} + C_6$

The temperature profile as a function of radial position and known zonal boundary temperatures $T_{CL,FU}$, $T_{s,FU,i}$, $T_{s,FU,o}$, and $T_{s,pb}$ can be rewritten in a form useful for interpolation by radial position:

$$T(r)\begin{cases} T_{CL,FU}\,, & 0 < r \le R_{FU,i} \\ T_{s,FU,i} + \left(T_{s,FU,o} - T_{s,FU,i}\right)\left(\dfrac{\frac{1}{R_{FU,i}} - \frac{1}{r}}{\frac{1}{R_{FU,i}} - \frac{1}{R_{FU,o}}}\right) + \left(\dfrac{\dot{q}}{6k_{FU}}\right)\left[{R_{FU,i}}^2 - r^2\right] + \\ \qquad \left(\dfrac{\dot{q}}{6k_{FU}}\right)\left[\left({R_{FU,o}}^2 - {R_{FU,i}}^2\right)\left(\dfrac{\frac{1}{R_{FU,i}} - \frac{1}{r}}{\frac{1}{R_{FU,i}} - \frac{1}{R_{FU,o}}}\right)\right], & R_{FU,i} \le r \le R_{FU,o} \\ T_{s,FU,o} + \left(T_{pb} - T_{s,FU,o}\right)\left(\dfrac{\frac{1}{R_{FU,o}} - \frac{1}{r}}{\frac{1}{R_{FU,o}} - \frac{1}{R_{pb}}}\right), & R_{FU,o} \le r \le R_{pb} \end{cases} \quad (9\text{-}52)$$

9.5.5 PMR Fuel Compact and Matrix Temperature Distribution

For the two zones allowed in the PMR compact/block matrix model, the temperature distribution can be generally written as:

$$T(r)\begin{cases} C_1\, ln(r) + C_2 - \dfrac{\dot{q}r^2}{4k_{FU}}, & r \le R_{FU} \\ C_3\, ln(r) + C_4\,, & R_{FU} \le r \le R_{MX,o} \end{cases} \quad (9\text{-}53)$$

The heat flux distribution can be written as:

$$q''(r)\begin{cases} \dfrac{-C_1 k_{FU}}{r} + {\dot{q}r}/{2}\,, & r \le R_{FU} \\ \dfrac{-C_3 k_{MX}}{r}, & R_{FU} \le r \le R_{MX,o} \end{cases} \quad (9\text{-}54)$$

Any gap between fuel and matrix is not explicitly treated when reconstructing a temperature profile for matrix in a PMR. Rather, the component temperatures T_{FU} and T_{MX} computed during COR package execution reflect the gap conductance.

Imposing average temperature conditions:

$$\bar{T}(r) = \begin{cases} T_{FU} = \left(\frac{1}{\pi R_{FU}^2}\right)\left(\int_0^{2\pi}\int_0^{R_{FU}} \left(C_1 \ln(r) + C_2 - \frac{\dot{q} r^2}{4k_{FU}}\right) r dr d\theta\right) \\ T_{MX} = \left(\frac{1}{\pi\left(R_{MX,o}^2 - R_{FU}^2\right)}\right)\left(\int_0^{2\pi}\int_{R_{FU}}^{R_{MX,o}} \left(C_3 \ln(r) + C_4\right) r dr d\theta\right) \end{cases} \quad (9\text{-}55)$$

The set of conditions that permit resolution of unknown zonal boundary temperatures $T_{s,FU}$, $T_{MX,i}$ and $T_{MX,o}$ and integration constants are:

$$C_1 = 0$$

$$C_2 = T_{FU} + \dot{q} R_{FU}^2 \Big/ 8k_{FU}$$

$$C_3 = -\dot{q} R_{FU}^2 \Big/ 2k_{MX} \quad (9\text{-}56)$$

$$C_4 = T_{MX} - \left(\frac{C_3}{2}\right)\left[\frac{2R_{FU}^2 \ln(R_{FU}) - R_{FU}^2 - R_{MX,o}^2\left(2\ln\left(R_{MX,o}\right) - 1\right)}{R_{FU}^2 - R_{MX,o}^2}\right]$$

The constants are used to solve for the compact center-line temperature, $T_{CL,FU}$, the compact fuel surface temperature, $T_{s,FU}$, and the matrix outer surface temperature $T_{MX,o}$. Then, interpolation according to radial position is used to fill out the full temperature profile at all radial points. The computational algorithm for the temperature profile proceeds as:

- Get $\dot{q}$, k_{FU}, k_{MX}, R_{FU}, $R_{MX,o}$
- Get $T_{CL,FU} = C_2$
- Get $T_{s,FU} = C_3 \ln(R_{FU}) + C_4$
- Get $T_{MX,o} = C_3 \ln\left(R_{MX,o}\right) + C_4$

The temperature profile as a function of radial position and known zonal boundary temperatures $T_{CL,FU}$, $T_{s,FU}$, and $T_{MX,o}$ can be rewritten in a form useful for interpolation by radial position:

$$T(r)\begin{cases} T_{CL,FU} + \left(T_{s,FU} - T_{CL,FU}\right)\left(r/R_{FU}\right)^2, & r \le R_{FU} \\ T_{s,FU} + \left(T_{MX,o} - T_{s,FU}\right)\left(\frac{\ln\left(R_{FU}/r\right)}{\ln\left(R_{FU}/R_{MX,o}\right)}\right), & R_{FU} \le r \le R_{MX,o} \end{cases} \quad (9\text{-}57)$$

9.5.6 Matrix Volumetric Heat Generation Rate

Note that a volumetric heat generation rate (due to fission and radioactive decay) is required to ascertain the temperature profile in the matrix of a fuel element. An appropriate value can be calculated from COR:

- Fission and decay power in a COR/diffusion cell
- Fuel mass in a COR/diffusion cell (strictly FU component material slot 1)
- Matrix "kernel" volume (i.e., volume of TRISO-bearing zone)
- Fuel mass per fuel unit in COR/diffusion cell

$$\dot{Q} = {M_{FU}^{unit}\left(P_f + P_d\right)}\Big/{N_p^{unit} V_k} \tag{9-58}$$

Where:

$\dot{Q} =$ Volumetric heat generation rate in matrix [W/m3]

$M_{FU}^{unit} =$ Fuel mass in a fuel unit (pebble or compact) [kg]

$P_f =$ Fission power per unit fuel mass [W/kg]

$P_d =$ Decay power per unit fuel mass [W/kg]

$V_k =$ Matrix "kernel" volume [m3]

9.5.7 "Thick Shell" Pebble Fuel Element in PBRs

When computing fuel-to-matrix heat transfer and matrix-to-coolant heat transfer in the COR package for the PBR reactor type, there previously were no special conductance terms to account for the effect of a sufficiently thick, unfueled pebble shell region. This stands in contrast to the COR component heat transfer computations performed for the PMR reactor type where the effects of a "thick" cylindrical region (transformed from the hexagonal geometry of prismatic graphite blocks) are computed for FU-to-MX and MX-to-coolant heat transfer. Changes for the PBR reactor type include:

1. Geometry computations that reflect the MX COR component spherical geometry (compare to the case for cylindrical PMR MX component)
    - Factors $a, f,$ and A_{MX}, but for spherical geometry:

$$a = \left[{1}/{R_{mx,i}} - {1}/{R_{mx,o}}\right]^{-1} \tag{9-59}$$

$$f = \frac{{}^{1}/_{R_{mx,i}}}{{}^{1}/_{R_{mx,i}} - {}^{1}/_{R_{mx,o}}} - \left(\frac{{}^{3}/_{2}}{{}^{1}/_{R_{mx,i}} - {}^{1}/_{R_{mx,o}}}\right)\left(\frac{R_{mx,o}^2 - R_{mx,i}^2}{R_{mx,o}^3 - R_{mx,i}^3}\right) \quad (9\text{-}60)$$

$$A_{MX}(1) = a/R_{mx,i}^2 \quad (9\text{-}61)$$

$$A_{MX}(2) = a/R_{mx,o}^2 \quad (9\text{-}62)$$

2. Fuel conductance term is modified to include ${}^{1}/_{z_0}$ where:

$$z_0 = {}^{k_{mx}a}\Big/_{fR_{mx,i}^2} = {}^{k_{mx}A_{MX}(1)}\Big/_{f} \quad (9\text{-}63)$$

3. MX component conductance term includes ${}^{1}/_{z_1}$ where:

$$z_1 = {}^{k_{mx}a}\Big/_{(1-f)R_{mx,o}^2} = {}^{k_{mx}A_{MX}(2)}\Big/_{(1-f)} \quad (9\text{-}64)$$

Note that for PBR and PMR reactor types, the FU-to-MX heat transfer conductance has a component attributable to conduction in the fuel:

$${}^{1}/_{h_f} = \begin{cases} \frac{R_{f,o}}{4k_f}, cylindrical \\ \frac{R_{f,o}}{5k_f}, spherical \end{cases} \quad (9\text{-}65)$$

These conductance terms may be derived from the assumed temperature profiles, the assumed heat flux profiles, and the definition of average temperature in each geometry:

Cylindrical:

$$T(r) = T_{f,o} + \left({}^{\dot{q}R_{f,o}^2}\Big/_{4k_f}\right)\left(1 - \left({}^{r}/_{R_{f,o}}\right)^2\right)$$
$$q''(r) = {}^{\dot{q}r}/_{2} \quad (9\text{-}66)$$

$$\bar{T} = \frac{\int_0^{2\pi}\int_0^{R_{f,o}}\left(T(r)\right)r dr d\theta}{(\pi)\left(R_{f,o}^2\right)} = T_{f,o} + \left({\dot{q}R_{f,o}^2}/{8k_f}\right)$$

$$q''\left(R_{f,o}\right) = {\dot{q}R_{f,o}}/{2} = \frac{\Delta T}{\Re A} = \frac{\bar{T} - T_{f,o}}{\Re A} = \frac{\left({\dot{q}R_{f,o}^2}/{8k_f}\right)}{\Re A} = hA\left(\bar{T} - T_{f,o}\right)$$

$$hA = \frac{1}{\Re A} = \frac{1}{\frac{R_{f,o}}{4k_f}}$$

Spherical:

$$T(r) = T_{f,o} + \left({\dot{q}R_{f,o}^2}/{6k_f}\right)\left(1 - \left({r}/{R_{f,o}}\right)^2\right)$$

$$q''(r) = {\dot{q}r}/{3}$$

$$\bar{T} = \frac{\int_0^{2\pi}\int_0^{\pi}\int_0^{R_{f,o}}\left(T(r)\right)r^2\sin\theta dr d\theta d\phi}{\left(\frac{4\pi}{3}\right)\left(R_{f,o}^3\right)} = T_{f,o} + \left({\dot{q}R_{f,o}^2}/{15k_f}\right) \qquad (9\text{-}67)$$

$$q''\left(R_{f,o}\right) = \frac{\dot{q}R_{f,o}}{3} = \frac{\Delta T}{\Re A} = \frac{\bar{T} - T_{f,o}}{\Re A} = \frac{\left(\frac{\dot{q}R_{f,o}^2}{15k_f}\right)}{\Re A} = hA(\Delta T) = \frac{(\Delta T)}{\frac{R_{f,o}}{5k_f}}$$

10. Discussion and Development Plans

In its inception, MELCOR was envisioned as a probabilistic risk analysis (PRA) tool that was to be fast running, making use of necessarily simplified physics models. In recent years, however, MELCOR has found increasing use as a best-estimate tool for severe accident analyses, and many of the physics models, including many in the COR package, have been improved considerably. Nevertheless, some of the simplified COR models remain today. In some cases, simplistic parametric models have been implemented until more advances have been made in furthering our understanding of the phenomena. In other cases, more sophisticated models are planned for implementation in the near future.

The following paragraphs are based on assessments of improvement needs for MELCOR in the area of core modeling, including deficiencies identified as part of the MELCOR Peer Review [80], and include work in progress. Suggestions from users regarding additional modification and/or upgrade of the COR package are welcomed and should be directed to the MELCOR Code Development Group.

10.1 Radiation

Radiation view factors in the COR package are defined globally, based on simple user input. Since correct characterization of many of these view factors is dependent on local geometry and nodalization; they should be definable on a local cell basis and updated internally with changing geometry. This upgrade would give the user more freedom to satisfactorily model radiative heat transfer within the core, a dominant heat transfer mechanism in reactor accidents. Improvements are planned for post-MELCOR 1.8.6 for the radiation treatment to include spatial variation of radiation view factors to account for node size effects.

10.2 Gap Cooling

While significant extensions to MELCOR's melt progression modeling have been implemented in version 1.8.6, treatment for the so-called gap cooling effect postulated as operative in preventing head failure in the TMI-2 accident is not implemented in the 1.8.6 and 2.1 releases. This is a possible area for future improvement.

10.3 Further Extensions to Molten Pool Modeling

The present models in Version 1.8.6 and 2.1 allow for a heavy ceramic molten pool and a lighter overlying molten pool. Latest research has suggested the possibility that chemical reduction of UO_2 could produce a heavy metallic layer that would drop to the bottom of a molten pool. As more becomes known about these phenomena, it may be considered for inclusion in future code releases.

10.4 Degraded Core Cooling

Potential improvements are under discussion with respect to MELCOR's falling debris heat transfer modeling and water cooling of debris beds to better model the quenching behavior predicted by these treatments. The 1-D debris bed quenching treatment may not adequately account for 2-D effects in lower head debris beds.

11. Sensitivity Coefficients

Sensitivity coefficients associated with various correlations and modeling parameters described in this reference manual are given below.

(Equation) or Chapter §	Coefficient	Value	Units
§2.5	C1001(1,1)	29.6	$kg^2(Zr)/m^4$-s
	C1001(2,1)	16820.0	K
	C1001(3,1)	87.9	$kg^2(Zr)/m^4$-s
	C1001(4,1)	16610.0	K
	C1001(5,1)	1853.0	K
	C1001(6,1)	1873.0	K
§2.5	C1001(1,2)	50.4	$kg^2(Zr)/m^4$-s
	C1001(2,2)	14630.0	K
	C1001(3,2)	0.0	$kg^2(Zr)/m^4$-s
	C1001(4,2)	0.0	K
	C1001(5,1)	10000.0	K
	C1001(6,2)	10000.0	K
§2.5	C1002(1)	2.42E09	$kg^2(steel)/m^4$-s
	C1002(2)	4.24E04	K
§2.5	C1003(1)	0.00548	$kg(Zr)-K/Pa-m^3$
	C1003(2)	0.00504	$kg(steel)-K/Pa-m^3$
§2.5	C1004(1)	1100.0	K
	C1004(2)	9900.0	K
§2.5	C1005(2)	2.E-2	-
	C1005(3)	9.E-1	-
	C1005(4)	1500.	K
§2.5	C1006(1)	1.662E5	s^{-1}
	C1006(2)	2.26472E4	K
§2.5	C1007(1..NRAD,1)	0.0	-
	C1007(1..NRAD,2)	0.0	-
§2.8.3	C1010(1,2)	1.47E14	-
	C1010(2,2)	8.01E4	K
	C1010(1,3)	1.02E15	-
	C1010(2,3)	8.14E4	K
	otherwise		
	C1010(1,J)	-1.	-
	C1010(2,J)	0.0	K
§2.8.1	C1011(1)	1400.	K
	C1011(2)	1400.	K
	C1011(3)	1520.	K
§3.1.5	C1020(1)	360.0	s

(Equation) or Chapter §	Coefficient	Value	Units
§3.2.5	C1020(2)	60.0	s
	C1020(3)	0.0	-
	C1020(4)	0.0	-
	C1020(5)	1.0	-
§1.1.1	C1021(1)	1.0	s
§2.6	C1030(1)	0.0	-
	C1030(2)	0.1	s
	C1030(3)	1.0	s
	C1030(4)	10.0	s
	C1030(5)	0.6	-
(2-80)	C1101(1)	0.8	-
	C1101(2)	0.325	-
	C1102(1)	616.4833	K
	C1102(2)	0.25617	-
	C1102(3)	0.0003474	K^{-1}
	C1102(4)	0.9999	-
	C1102(5)	0.0001	-
	C1103(1)	1000.0	K
	C1103(2)	0.9999	-
	C1103(3)	0.0	K^{-1}
	C1103(4)	0.9999	-
	C1103(5)	0.0001	-
	C1104(1)	0.325	-
	C1104(2)	3.8799999E-6	m
	C1104(3)	0.808448	-
	C1104(4)	0.001	m
	C1104(5)	0.758642	-
	C1104(6)	1500.0	K
	C1104(7)	300.0	K
	C1104(8)	0.9999	-
	C1104(9)	0.325	-
§3.1.3	C1131(1)	0.00001	m
	C1131(2)	2400.0	K
	C1131(3)	0.001	m
	C1131(4)	1700.0	K
	C1131(5)	0.00001	m
	C1131(6)	2100.0	K
§3.2	C1132(1)	2500.0	K
	C1132(2)	3100.0	K

(Equation) or Chapter §	Coefficient	Value	Units
(3-13)	C1141(1)	1.0	s
	C1141(2)	1.0	kg/m-s

Equation	Coefficient	Value	Units
(3-18) - (3-26)	C1151(I,1)	0.556	-
	C1151(I,2)	0.807	-
	C1151(I,3)	0.143	-
	C1151(I,4)	0.396	-
	C1151(I,5)	0.0	-
	C1151(2,6)	1.0	-
	C1151(6,6)	1.0	-
	C1151(9,6)	1.0	-
	otherwise		
	C1151(I,6)	0.0	-
	C1151(I,7)	100.0	1/m
(3-27)	C1152(1)	1000.	1/m
(2-100)	C1200(1)	0.5	-
	C1200(2)	0.9	-
(2-101)	C1212(1)	4.36	-
	C1212(2)	4.36	-
(2-101)	C1213(1)	0.00826	-
	C1213(2)	0.00110	-
(2-104)	C1214(1)	0.023	-
	C1214(2)	0.8	-
	C1214(3)	0.4	-
(2-105)	C1221(1)	0.18	-
	C1221(2)	0.25	-
	C1221(3)	-1./9.	-
(2-106)	C1222(1)	0.065	-
	C1222(2)	1./3.	-
	C1222(3)	-1./9.	-
(2-107)	C1231(1)	2.0	-
	C1231(2)	0.60	-
	C1231(3)	0.5	-
	C1231(4)	1./3.	-
(2-108)	C1232(1)	2.0	-
	C1232(2)	0.60	-
	C1232(3)	0.25	-
	C1232(4)	1./3.	-
(2-109)	C1241(1)	34.5	$W/m^2\text{-}K\text{-}Pa^{1/4}\text{-}K^{1.523}$

Equation	Coefficient	Value	Units
	C1241(2)	0.25	-
	C1241(3)	1.523	-
	C1241(4)	23.4	K
§2.3.6	C1241(5)	0.0	-
(2-110)	C1242(1)	1.41E07	$W/m^2\text{-}K\text{-}Pa^{1/4}\text{-}K^{-2.575}$
	C1242(2)	0.25	-
	C1242(3)	-2.575	-
(2-112)	C1244(1)	0.756	-
	C1244(2)	0.089	m
	C1244(3)	0.15	-
(6-9)	C1245(1)	0.034	-
	C1245(2)	0.0037	-
	C1245(3)	0.656	-
(6-10)	C1245(4)	4.8E-4	-
	C1245(5)	8.2E-4	-
	C1245(6)	0.407	-
§6.1	C1245(7)	0.0	-
§6.1	C1245(8)	0.142	-
	C1245(9)	0.3333333	-
§6.1	C1245(10)	0.055	-
	C1245(11)	0.016	-
	C1245(12)	0.5	-
(6-8)	C1246(1)	10.0	$W/m^2\text{-}K$
§2.2	C1250(1)	3200.	K
	C1250(2)	0.01	K^{-1}
(2-195)	C1301(1)	0.037	-
	C1301(2)	0.3	-
	C1301(3)	0.7	-
	C1301(4)	2.4384	m
	C1301(6)	7.65318E06	Pa
§2.2.3	C1260(1)	600.0	K
§2.2.2	C1260(2)	40.0	K
§2.2.3	C1260(3)	1.5E5	$W/m^2\text{-}K$
§2.2.2	C1260(4)	125.0	$W/m^2\text{-}K$
§2.2.2	C1270(1)	0.6	-
	C1270(2)	0.1	-

Equation	Coefficient	Value		Units
		BWR	PWR	
§2.7.1	C1311(1)	0.735	0.500	-

Equation	Coefficient	Value		Units
		BWR	PWR	
	C1311(2)	0.400	0.541	-
	C1311(3)	0.292	0.565	-
	C1311(4)	0.263	0.234	-
	C1311(5)	0.400	0.541	-
	C1311(6)	0.292	0.565	-
	C1311(7)	0.400	0.541	-
§2.7.1	C1312(1)	0.9	-	
	C1312(2)	1.0	-	
	C1312(3)	1.0	-	
	C1312(4)	1.0	-	
	C1312(5)	1.0	-	
	C1312(6)	1.0	-	
	C1312(7)	1.0	-	
	C1312(8)	1.0	-	
	C1312(9)	1.0	-	
		BWR	PWR	
§2.7.2	C1321(1)	0.735	0.500	-
	C1321(2)	0.400	0.541	-
	C1321(3)	0.292	0.565	-
	C1321(4)	0.263	0.234	-
	C1321(5)	0.400	0.541	-
	C1321(6)	0.292	0.565	-
	C1321(7)	0.400	0.541	-
§2.7.2	C1322(1)	0.9	-	
	C1322(2)	1.0	-	
	C1322(3)	1.0	-	
	C1322(4)	1.0	-	
	C1322(5)	1.0	-	
	C1322(6)	1.0	-	
	C1322(7)	1.0		
	C1322(8)	1.0	-	
	C1322(9)	1.0	-	
§1.1	C1401(1)	1.6	-	
	C1401(2)	0.8	-	
	C1401(3)	-1.0	-	
	C1401(4)	20.0	-	
	C1401(5)	0.5	-	
	C1401(6)	1.0	-	
§1.1.1	C1501(1)	0.5	-	
	C1501(2)	0.5	-	

Equation	Coefficient	Value		Units
		BWR	PWR	
	C1501(3)	0.5	-	
	C1501(4)	0.5	-	
	C1501(5)	0.5	-	

Equation	Coefficient	Value	Units
§1.1	C1502(1)	1.0E-6	kg
	C1502(2)	10.0	kg
§3.4	C1504(1)	10*unit round-off	-
	C1504(2)	1.E-4	-
§3.1.2	C1505(1)	1.0E-5	-
§3.1.6	C1505(2)	0.05	-
§6.2	C1600(1)	0.0	-
(6-31)	C1600(2)	1.E-5	K^{-1}
§6.2	C1600(3)	1.E3	Pa
(6-24)	C1601(1)	-4.725E3	-
	C1601(2)	4.812E4	-
(6-23)	C1601(3)	7.042	-
(6-25)	C1601(4)	0.18	-
(6-27)	C1602(1)	2.E11	Pa
	C1602(2)	1800.	K
	C1602(3)	900.	K
	C1602(4)	6.	-
(6-28)	C1603(1)	4.E8	Pa
	C1603(2)	1800.	K
	C1603(3)	900.	K
	C1603(4)	6.	-
(5-21)	C1604(1)	-7.5E3	-
	C1604(2)	8.1E4.	-
(5-22)	C1604(3)	16.44	-
(5-20	C1605(1)	370.E9	Pa
	C1605(2)	1700.	K
	C1605(3)	1650.	K
	C1605(4)	3.0	-
(5-20	C1606(1)	260.E6	Pa
	C1606(2)	1700.	K
	C1606(3)	800.	K
	C1605(4)	3.0	-

Appendix A. HTGR Fission Product Release/Transport Model

A.1 Tracked Fission Product Species Diffusion Calculation

Finite Difference Equations

To derive finite difference equations for nodal fission product species concentrations, the diffusion equation – in steady-state or transient form – must be multiplied by a differential element of volume and integrated over a node volume given the assumptions that generation rate and diffusion coefficient are constant over a node. Boundary conditions must also be appropriately applied and incorporated for boundary nodes. At node interfaces away from the inner and outer boundary, the diffusion flux is constant, and the species concentrations exhibit a proportionality (often an equality), thus:

$$D_i \frac{\partial C_i}{\partial r} \Big|_{r_i} = D_{i+1} \frac{\partial C_{i+1}}{\partial r} \Big|_{r_i} \tag{A-1}$$

$$C_i(r_i) = \gamma_i C_{i+1}(r_i) \tag{A-2}$$

Where:

C_i = Species concentration, node i [kmol/m3]

D_i = Diffusion coefficient, node i [m^2/s]

r_i = Radius, outer interface of node i [m]

γ_i = Partition coefficient, typically taken as unity [-]

The differential element of volume depends on the coordinate system chosen:

$$dV = \begin{cases} cylindrical: 2\pi r dr \\ spherical: \quad 4\pi r^2 dr \end{cases} \tag{A-3}$$

Note that in the cylindrical coordinate case, the element of volume is a volume per unit length so that the solution is carried out on a per-unit-length basis. Alternatively, a known length of cylinder may be included in the element of volume and carried through the integration and finite differencing. The integration proceeds as:

$$\int_{r_{i-1}}^{r_i} \left[\chi \frac{\partial C}{\partial t} = \frac{1}{r^n} \frac{\partial}{\partial r} \left(r^n D \frac{\partial C}{\partial r} \right) - \lambda C + \beta \right] dV \rightarrow$$

$$\Delta V_i \chi \frac{\partial C_i}{\partial t} = \left(AD \frac{\partial C}{\partial r} \right)_{r_{i-1}}^{r_i} - \lambda C_i \Delta V_i + \beta \Delta V_i \tag{A-4}$$

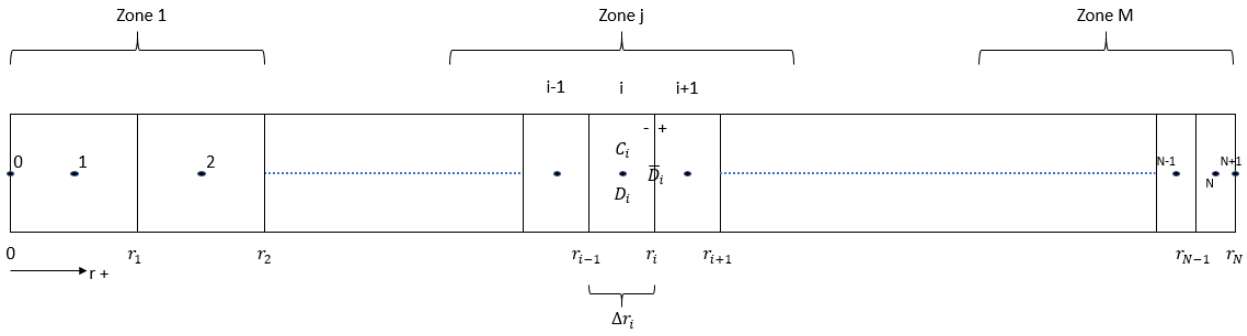


Figure A.1 Finite volume diffusion grid

The nodal equation can be written in terms of unknown interface fluxes assuming an implicit differenced form for the concentration time derivative:

$$\Delta V_i \chi \left(\frac{C_i^n - C_i^{n-1}}{\Delta t} \right) = A_i D_i \frac{\partial C}{\partial r}_{r_i} - A_{i-1} D_{i-1} \frac{\partial C}{\partial r}_{r_{i-1}} - \lambda C_i \Delta V_i + \beta \Delta V_i \tag{A-5}$$

The nodal surface flux terms – evaluated at node-to-node interfaces - must be written with respect to cell-centered concentrations and diffusion coefficients via “effective” diffusion coefficients that apply at node-to-node interfaces. Such a form may be derived by considering that the flux must be constant at the interface and that concentrations on either side of the interface ($C^+$and C^-) are equal:

$$D_i \frac{\partial C_i}{\partial r}_{r_i} = D_i \left(\frac{C^- - C_i}{\Delta r_i / 2} \right) = D_i \left(\frac{C_{i+1} - C^+}{\Delta r_{i+1} / 2} \right) \tag{A-6}$$

By solving for C^-, assuming the equality $C^- = C^+$ and substituting:

$$D_i \frac{\partial C_i}{\partial r}_{r_i} = \bar{D}_i \left(\frac{C_{i+1} - C_i}{\Delta \bar{r}_i} \right) \tag{A-7}$$

Where the effective node i diffusion coefficient, [m2/s], is:

$$\bar{D}_i = (\Delta r_i + \Delta r_{i+1}) \Big/ \left(\frac{\Delta r_i}{D_i} + \frac{\Delta r_{i+1}}{D_{i+1}} \right) \tag{A-8}$$

and an effective node i width [m] is:

$$\Delta \bar{r}_i = \frac{\Delta r_i + \Delta r_{i+1}}{2} \tag{A-9}$$

This amounts to an interface diffusion flux expression involving 1) a diffusion coefficient equal to the harmonic mean of values at nodal points between the interface, and 2) the distance between nodes i.e.:

$$\overline{D}_i = \frac{2D_iD_{i+1}}{D_iD_{i+1}} \tag{A-10}$$

Such that:

$$D_i \frac{\partial C_i}{\partial r}_{r_i} = \overline{D}_\iota \left(\frac{C_{i+1} - C_i}{\Delta r_i}\right) \tag{A-11}$$

Using the former interpretation of interface diffusion flux, the difference equation is:

$$\begin{aligned} \Delta V_i \chi \left(\frac{C_i^n - C_i^{n-1}}{\Delta t}\right) \\ = A_i \overline{D}_i \left(\frac{C_{i+1} - C_i}{\Delta \bar{r}_i}\right) - A_{i-1}\overline{D}_{i-1}\left(\frac{C_i - C_{i-1}}{\Delta \bar{r}_{i-1}}\right) - \lambda C_i \Delta V_i \\ + \beta \Delta V_i \end{aligned} \tag{A-12}$$

After rearranging for solution form, the difference equation may be rewritten as:

$$\begin{aligned} C_{i-1}^n \left(\frac{-A_{i-1}\overline{D}_{i-1}}{\Delta \bar{r}_{i-1}}\right) + C_i^n \left(\frac{A_i \overline{D}_i}{\Delta \bar{r}_i} + \frac{A_{i-1}\overline{D}_{i-1}}{\Delta \bar{r}_{i-1}} + \lambda \Delta V_i + \frac{\chi \Delta V_i}{\Delta t}\right) \\ + C_{i+1}^n \left(\frac{-A_i \overline{D}_i}{\Delta \bar{r}_i}\right) = \beta \Delta V_i + \frac{\chi \Delta V_i C_i^{n-1}}{\Delta t} \end{aligned} \tag{A-13}$$

Solution Form

A system of equations for all nodes (across all zones, for a given tracked fission product species, for a given region/model) assume a solution form:

$$a_i C_{i-1}^n + b_i C_i^n + c_i C_{i+1}^n = d_i \tag{A-14}$$

In general, the values of coefficients a_i, b_i, c_i, d_i can be ascertained by comparison with the solution form of the difference equation above. For a node on the interior of the computational domain:

$$\begin{aligned} a_i &= \frac{-A_{i-1}\overline{D}_{i-1}}{\Delta \bar{r}_{i-1}} \\ b_i &= \frac{A_i \overline{D}_i}{\Delta \bar{r}_i} + \frac{A_{i-1}\overline{D}_{i-1}}{\Delta \bar{r}_{i-1}} + \lambda \Delta V_i + \frac{\chi \Delta V_i}{\Delta t} \end{aligned} \tag{A-15}$$

$$c_i = \frac{-A_i \overline{D}_i}{\Delta \bar{r}_i}$$

$$d_i = \beta \Delta V_i + \frac{\chi \Delta V_i C_i^{n-1}}{\Delta t}$$

For boundary nodes, these coefficients are altered as discussed subsequently. Note also that equation coefficients a_{i+1} and b_i may be adjusted if partition coefficients – denoted by α – between zones in a model/region are defined by the user. If an interface between node i and node $i+1$ corresponds to a transition in zones (last node of inner zone is i and first node of outer zone is $i+1$), then:

$$\begin{aligned} b_i &= b_i + (1-\alpha)c_i \\ a_{i+1} &= a_{i+1}\alpha \end{aligned} \qquad \text{(A-16)}$$

The effective diffusion coefficient for such an interface) is also modified as:

$$\overline{D}_i = (\Delta r_i + \Delta r_{i+1}) \Big/ \left(\frac{\Delta r_i}{D_i} + \alpha \frac{\Delta r_{i+1}}{D_{i+1}}\right) \qquad \text{(A-17)}$$

Diffusion Equation Boundary Conditions

To account for boundary conditions in exterior node equations, coefficients a_i through d_i are modified accordingly. Diffusion equation boundary conditions are specified for each tracked fission product species for each model. A symmetry condition is enforced at the inner or “left hand” side of the cylindrical/spherical domain (zones of nodes) of each model for each tracked fission product species. At the exterior boundary of the cylindrical or spherical domain, a specified zero concentration boundary condition is applied. This assumption is mathematically akin to an extrapolated zero scalar flux (vacuum) boundary condition in the context of neutron diffusion theory. The claim is that the tracked species concentration is tending towards zero at the non-reentrant outer surface of a model’s geometry and not that species concentration is identically zero at the outer surface though this is the mathematically enforced boundary condition.

Inner Domain Boundary - Zero Flux or Symmetry Condition

In this case, the expression for node $i = 1$ diffusion flux at the domain boundary simply vanishes to zero. Then, the coefficients change from default interior domain values to:

$$\begin{aligned} a_1 &= 0 \\ b_1 &= \frac{A_1 \overline{D}_1}{\Delta \bar{r}_1} + \lambda \Delta V_1 + \frac{\chi \Delta V_1}{\Delta t} \end{aligned} \qquad \text{(A-18)}$$

$$c_1 = \frac{-A_1 \overline{D}_1}{\Delta \bar{r}_1}$$

$$d_1 = \beta \Delta V_1 + \frac{\chi \Delta V_1 C_1^{n-1}}{\Delta t}$$

Specified Concentration Condition

In this case, the boundary condition value for fission product species concentration is used directly in the expression for node $i = N$ diffusion flux at the domain boundary. Then, the coefficients change from default interior domain values to:

$$\begin{aligned} a_N &= \frac{-A_{N-1} \overline{D}_{N-1}}{\Delta \bar{r}_{N-1}} \\ b_N &= \frac{A_{N-1} \overline{D}_{N-1}}{\Delta \bar{r}_{N-1}} + \frac{A_N \overline{D}_N}{\Delta \bar{r}_N} + \lambda \Delta V_N + \frac{\chi \Delta V_N}{\Delta t} \\ c_N &= 0 \\ d_N &= \beta \Delta V_N + \left(\frac{A_N \overline{D}_N}{\Delta \bar{r}_N} \right) C_{N+1} + \frac{\chi \Delta V_N C_N^{n-1}}{\Delta t} \end{aligned} \quad \text{(A-19)}$$

Where:

C_{N+1} = User-specified molar concentration at "N+1" [kmol/m^3]

The zero-concentration condition is simply a special case where $C_{N+1} = 0$ so that:

$$\begin{aligned} a_N &= \frac{-A_{N-1} \overline{D}_{N-1}}{\Delta \bar{r}_{N-1}} \\ b_N &= \frac{A_{N-1} \overline{D}_{N-1}}{\Delta \bar{r}_{N-1}} + \frac{A_N \overline{D}_N}{\Delta \bar{r}_N} + \lambda \Delta V_N + \frac{\chi \Delta V_N}{\Delta t} \\ c_N &= 0 \\ d_N &= \beta \Delta V_N + \frac{\chi \Delta V_N C_N^{n-1}}{\Delta t} \end{aligned} \quad \text{(A-20)}$$

A.2 Sorption Isotherm Empirical Model

The sorption isotherm model describes the transition in (metallic) fission product concentration occurring at/near a solid/gas interface. Metallic fission products that diffuse through a solid to a solid/gas interface must – as a prerequisite to further release - evaporate to gas and then either 1) be removed by advection, or 2) diffuse through gas and sorb/deposit on other surfaces before continuing to diffuse. The sorption isotherm

empirical model describes the non-linear, temperature-dependent relationship between sorbate fission product concentration (in the sorbent, or solid-phase) and sorbate fission product gas-phase vapor pressure assuming 1) spontaneous adjustment to equilibrium between these quantities, and 2) continued existence of equilibrium between these two quantities. This empirical model is useful in the context of in-core fission product transport calculations because it can describe:

- Fission product behavior around the small gap between the fuel compact outer surface and the graphite webbing inner surface (PMR using fuel compacts within graphite hex blocks)
- Fission product behavior at a surface/coolant boundary (PMR in coolant holes, PBR at pebble outer surface)

The sorption isotherm model can be leveraged to fit within a finite volume diffusion calculation scheme.

Vapor pressure and solid-phase concentration per unit mass (typical units of Pa and μmol/g, respectively) are related by empirical sorption isotherm equations. Generally:

$$P = P(\boldsymbol{C}, T) \tag{A-21}$$

More specifically, the total vapor pressure of a sorbate is the sum of two vapor pressure components corresponding to two different regimes or regions: Henrian/Langmuir and Freundlich. Within each component, the logarithm of vapor pressure is correlated to the logarithm of sorbate concentration via the empirical model formulation:

$$\begin{aligned} P &= P_F + P_H \\ \ln(P_F) &= \left(A + {}^{B}/_{T}\right) + \left(D + {}^{E}/_{T}\right)\ln(\boldsymbol{C}) \\ \ln(P_H) &= \left(A + {}^{B}/_{T}\right) + \left(D - 1 + {}^{E}/_{T}\right)\ln(\boldsymbol{C}_{tr}) + \ln(\boldsymbol{C}) \end{aligned} \tag{A-22}$$

Where:

P_F = Freundlich pressure [Pa]

P_H = Henrian pressure [Pa]

A, B, D, E = Empirical constants

T = Temperature [K] indicative of sorbate (solid surface temperature)

$\boldsymbol{C}$ = Mass concentration of sorbate [μmol/g]

$\boldsymbol{C}_{tr}$ = Transition (Henrian to Freundlich) mass concentration [μmol/g]

The transition concentration is sometimes modeled as a constant and sometimes modeled empirically as:

$$\ln(\boldsymbol{C}_{tr}) = F + {}^{G}/_{T} \tag{A-23}$$

Where:

F, G = Empirical constants

Note also that the volumetric concentration is related to the mass concentration via:

$$\boldsymbol{C} = {}^{c}/_{\rho} \tag{A-24}$$

Where:

c = Volumetric concentration [µmol/cm^3]

ρ = Mass density [g/cm^3]

The convention chosen here is that c represents volumetric concentration while $\boldsymbol{C}$ represents mass concentration. Note also that – under an ideal gas assumption – the volumetric concentration of a sorbate can be uniquely determined as a function of vapor pressure:

$$c = \frac{P}{RT} \tag{A-25}$$

Where:

P = Vapor pressure of sorbate [Pa]

R = Universal gas constant, e.g. 8.314 [Pa*m^3/mol/K]

T = Temperature [K] indicative of sorbate (solid surface temperature)

Note the importance of dimensional consistency between R, P, and the desired concentration units when employing the ideal gas equation. The final form of the pressure/concentration relationship is:

$$P(\boldsymbol{C}, T) = e^{X(T)}\left[\boldsymbol{C}_{tr}^{Y(T)-1}\boldsymbol{C} + C^{Y(T)}\right] \tag{A-26}$$

Where:

$X(T) = A + {}^{B}/_{T}$

$Y(T) = D + {}^{E}/_{T}$

The above relationships may be deployed alongside other assumptions to treat – in the context of a finite volume diffusion solution - situations where fission product species transport is not by pure diffusion through solid graphite:

- A gas gap exists in a fuel element (e.g., the helium gap between a fuel compact and the graphite hex block webbing in a PMR)
- A graphite surface interfaces to coolant (e.g., the graphite hex block coolant hole surfaces in a PMR or the pebble outer surface in a PBR)

When such an interface exists, it may be defined with respect to the sorption isotherm empirical model and the typical nodal difference equations for neighboring nodes can be rewritten accordingly.

Gas Gap Interface Treatment in Diffusion Solution

A gas gap interface model is meant to treat fission product transport across a small gas gap between two solid surfaces. This occurs, for example, in a PMR where a fuel compact outer surface interfaces with the graphite webbing of its host graphite hexagonal block across a small helium gap. Separate sorption isotherm empirical models are applied for each surface, and the diffusion equations of each node for which the gas gap interface makes up a boundary are modified. The key physical assumption is vapor pressure equilibrium on both sides of the gas gap. Thus, for a given fission product species, the solid sorbate concentration at the inner surface generates a vapor pressure that is equal to the vapor pressure generated by the solid sorbate concentration at the outer surface. Since the sorption isotherm empirical model describes these equivalent vapor pressures, a nonlinear relationship between the sorbate concentrations at each surface may be derived. This condition is then effectively imposed as a nonlinear boundary condition, which requires linearization and iteration to solve in the context of the diffusion solution. Note this strategy resembles that of the TRAFIC code that is meant to model release of metallic fission products from an HTGR core [78]. The key difference in MELCOR is that a finite volume solution of the diffusion problem is used as opposed to alternative methods (e.g., of TRAFIC). Thus, the specific near-gap nodal difference equations must be derived.

In the MELCOR diffusion solution, a gas gap interface may exist in a model between two zones such that the nodes on either side of the interface belong to separate, neighboring zones. The intent is that the gas gap interface would be used in a PMR type reactor in the matrix model between a fuel-bearing inner zone (that gets fission product source terms from all fuel models) representing the fuel compact and an unfueled outer zone representing graphite webbing. A full complement of sorption isotherm data must be given for both surfaces.

The equilibrium vapor pressure assumption is borne out mathematically as:

$$P = P^{-}(\boldsymbol{C}^{-}, T^{-}) = P^{+}(\boldsymbol{C}^{+}, T^{+}) \tag{A-27}$$

Where:

P^{-} = Vapor pressure on inner surface ; P^{+} = on outer surface [Pa]

$\boldsymbol{C}^{-}$ = Mass concentration on inner surface ; $\boldsymbol{C}^{+}$ = on outer surface [μmol/g]

T^{-} = Temperature on inner surface ; T^{+} = on outer surface [K]

Given the form of the sorption isotherm empirical model, this relation between $\boldsymbol{C}^{-}$ and $\boldsymbol{C}^{+}$ is nonlinear. Note that for a guessed $\boldsymbol{C}^{+}$, a value for $\boldsymbol{C}^{-}$ can be obtained. This suggests a solution involving an iterative strategy with linearization of the nonlinearity. Note also that the MELCOR diffusion solution works with volumetric concentration [kmol/m^3] so that the nodal difference equations must be posed in those terms. This implies that conversions must occur for volumetric concentration to and from mass concentration as dictated by the sorption isotherm model.

The linearized relationship between c^{-} and c^{+} is a first-order Taylor expansion of c^{-} about a latest-iterate (guess) c^{+}, i.e.:

$$c_i^{-} = c_{i-1}^{-} + \left(\frac{dc^{-}}{dc^{+}}\right)(c_i^{+} - c_{i-1}^{+}) \tag{A-28}$$

Where:

i = current iterate

$i-1$ = last iterate

The chain rule of differentiation, the sorption isotherm pressure/concentration relationship, and the volume/mass concentration relationship may be used to evaluate the derivative:

$$\frac{dc^{-}}{dc^{+}} = \gamma = \frac{\frac{dP}{d\boldsymbol{C}^{+}}}{\frac{dP}{d\boldsymbol{C}^{-}}} = \frac{\left(\frac{dP}{d\boldsymbol{C}^{+}}\right)\left(\frac{d\boldsymbol{C}^{+}}{dc^{+}}\right)}{\left(\frac{dP}{d\boldsymbol{C}^{-}}\right)\left(\frac{d\boldsymbol{C}^{-}}{dc^{-}}\right)} \tag{A-29}$$

Where:

$$\frac{d\boldsymbol{C}^{+}}{dc^{+}} = \frac{d}{dc^{+}}\left(c^{+}/\rho^{+}\right) = 1/\rho^{+}$$

$$\frac{d\boldsymbol{C}^{-}}{dc^{-}} = \frac{d}{dc^{-}}\left(c^{-}/\rho^{-}\right) = 1/\rho^{-}$$

$$\frac{dP}{d\boldsymbol{C}^{+/-}} = \frac{d}{d\boldsymbol{C}^{+/-}}\left(e^{X(T^{+/-})}\left[\boldsymbol{C}_{tr}^{(Y(T^{+/-})-1)}\boldsymbol{C}^{+/-} + \boldsymbol{C}^{+/-^{Y(T^{+/-})}}\right]\right)$$
$$= e^{X(T^{+/-})}\left[\boldsymbol{C}_{tr}^{(Y(T^{+/-})-1)} + \left(Y\left(T^{+/-}\right)\right)\boldsymbol{C}^{+/-^{(Y(T^{+/-})-1)}}\right]$$

- The iterative strategy employed alongside the specialized nodal difference equations is:
- Guess or use last iterate outer surface volumetric concentration c_{i-1}^{+} [kmol/m^3]
- Evaluate the corresponding mass concentration $\boldsymbol{C}_{i-1}^{+}$ [kmol/kg]
- Use the sorption isotherm model and the outer surface empirical constants to evaluate the vapor pressure corresponding to $\boldsymbol{C}_{i-1}^{+}$
- Using a Newton's method approach, the sorption isotherm model, and the inner surface empirical constants, find the corresponding inner surface mass concentration $\boldsymbol{C}_{i-1}^{-}$ [kmol/kg]
- Evaluate the inner-to-outer volumetric concentration derivative $\gamma = \frac{dc^{-}}{dc^{+}}$
- Using i-1 level information and the derivative γ, assign values for modified nodal equation coefficients (a, b, c, d) and do a tridiagonal matrix solve to obtain updated cell-centered volumetric concentrations near the gas gap interface
- Obtain updated estimates c_i^{-} and c_i^{+} and convert to mass concentrations $\boldsymbol{C}_i^{-}$ and $\boldsymbol{C}_i^{+}$
- Check pressure convergence, i.e. $P(\boldsymbol{C}_i^{-}) \sim P(\boldsymbol{C}_i^{+})$ within tolerance
- If converged on gas gap vapor pressure then cease iteration, otherwise revise the guess of outer surface volumetric concentration by shuffling values $\boldsymbol{C}_{i-1}^{+} = \boldsymbol{C}_i^{+}$

Assuming the gas gap is the outer boundary of the Ith node, the finite volume solution grid locally near the gas gap looks like:

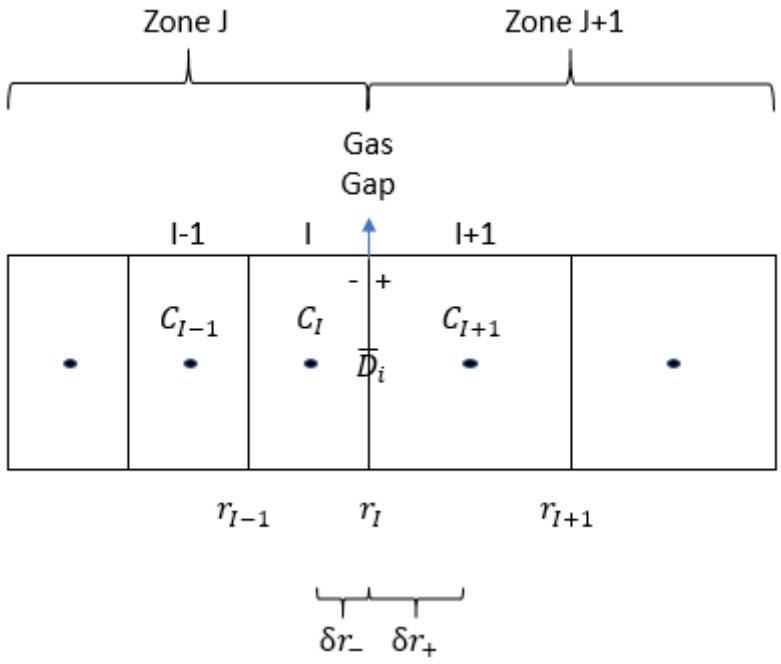


Figure A.2 Finite volume diffusion grid near gas gap interface

The gas gap interface exists between nodes I and $I+1$ and hence between zones J and $J+1$ as node I is the last/outermost node of zone J and node $I+1$ is the first/innermost node of zone $J+1$. The conventional interface condition of volumetric concentration equality $c^- = c^+$ must be replaced by the functional form that results from (1) the sorption isotherm model, and (2) the assumption of equilibrium gas gap vapor pressure: $c^- = f(c^+)$. Furthermore, mass flux continuity across the gas gap is a tacit assumption of vapor pressure equilibrium. Thus, there is no kinetic resistance in the gas gap, i.e. once a fission product atom is desorbed – from the inner solid surface - as a vapor in the gap, that atom is immediately available for adsorption on the outer solid surface at the other end of the gap. No time delays are caused by vapor diffusion in the gap.

Recall the generic nodal equation for nodes on the interior of the problem domain:

$$\Delta V_i \chi \left(\frac{C_i^n - C_i^{n-1}}{\Delta t} \right) = A_i D_i \frac{\partial C}{\partial r}_{r_i} - A_{i-1} D_{i-1} \frac{\partial C}{\partial r}_{r_{i-1}} - \lambda C_i \Delta V_i + \beta \Delta V_i \qquad \text{(A-30)}$$

For node I and $I+1$, the gas gap assumptions must be utilized to write new expressions for diffusion flux at the gas gap interface. Utilizing mass flux continuity and assigning int to represent the gas gap interface:

$$D_{int} \left(\frac{c_{I+1} - c_I}{\delta r_- + \delta r_+} \right) = D_I \left(\frac{c_i^- - c_I}{\delta r_-} \right) = D_{I+1} \left(\frac{c_{I+1} - c_i^+}{\delta r_+} \right) \qquad \text{(A-31)}$$

Using the right-hand side of this equality and the linearization of c_i^-, one can write c_i^+ in terms of c_i^-, c_{i-1}^-, $\gamma = \frac{dc^-}{dc^+}$, and c_{i-1}^+:

$$c_i^+ = \frac{c_i^- - c_{i-1}^-}{\gamma} + c_{i-1}^+ \tag{A-32}$$

Then, one can substitute for c_i^- such that the node I gas gap interface flux can be written purely in terms of c_{i-1}^-, γ, and c_{i-1}^+:

$$c_i^- = \frac{\left({}^{D_I}/_{\delta r_-}\right) c_I + \left({}^{D_{I+1}}/_{\delta r_+}\right)\left[c_{I+1} - c_{i-1}^+ + {}^{c_{i-1}^-}/_{\gamma}\right]}{{}^{D_I}/_{\delta r_-} + {}^{D_{I+1}}/_{\gamma \delta r_+}} \tag{A-33}$$

$$D_{int}\left(\frac{c_{I+1} - c_I}{\delta r_- + \delta r_+}\right) = \frac{\gamma(c_{I+1} - c_{i-1}^+) + (c_{i-1}^- - c_I)}{{}^{\gamma \delta r_+}/_{D_{I+1}} + {}^{\delta r_-}/_{D_I}} \tag{A-34}$$

Likewise, for node $I + 1$:

$$c_i^+ = \frac{\left({}^{D_{I+1}}/_{\delta r_+}\right) c_{I+1} + \left({}^{D_I}/_{\delta r_-}\right)[c_I + \gamma c_{i-1}^+ - c_{i-1}^-]}{\gamma {}^{D_I}/_{\delta r_-} + {}^{D_{I+1}}/_{\delta r_+}} \tag{A-35}$$

$$D_{int}\left(\frac{c_{I+1} - c_I}{\delta r_- + \delta r_+}\right) = \frac{\gamma(c_{I+1} - c_{i-1}^+) + (c_{i-1}^- - c_I)}{{}^{\gamma \delta r_+}/_{D_{I+1}} + {}^{\delta r_-}/_{D_I}} \tag{A-36}$$

For node I, two nodal equation coefficients change and the effective interface diffusion coefficient for the gas gap is modified:

$$D_{int} = \overline{D}_I = \frac{1}{{}^{\gamma f_{int}}/_{D_{I+1}} + {}^{(1 - f_{int})}/_{D_I}} \tag{A-37}$$

Where:

$$f_{int} = \frac{\delta r_+}{\delta r_+ + \delta r_-} \quad and \quad (1 - f_{int}) = \frac{\delta r_-}{\delta r_+ + \delta r_-}$$

While a and d change:

$$a_i = \frac{-\gamma A_I \overline{D}_I}{(\delta r_+ + \delta r_-)_I} \tag{A-38}$$

$$d_i = \beta \Delta V_I + \frac{\chi \Delta V_I C_I^{n-1}}{\Delta t} - \left(\frac{\gamma A_I \overline{D}_I}{(\delta r_+ + \delta r_-)_I} \right) c_{I-1}^+ + \left(\frac{A_I \overline{D}_I}{(\delta r_+ + \delta r_-)_I} \right) c_{I-1}^- \quad \text{(A-39)}$$

For node $I+1$, the effective interface diffusion coefficient $D_{int} = \overline{D}_I$ is the same, but nodal equation coefficients become:

$$b_{I+1} = \frac{A_{I+1} \overline{D}_{I+1}}{(\delta r_+ + \delta r_-)_{I+1}} + \frac{\gamma A_I \overline{D}_I}{(\delta r_+ + \delta r_-)_I} + \lambda \Delta V_{I+1} + \frac{\chi \Delta V_{I+1}}{\Delta t} \quad \text{(A-40)}$$

$$d_I = \beta \Delta V_I + \frac{\chi \Delta V_I C_I^{n-1}}{\Delta t} + \left(\frac{\gamma A_I \overline{D}_I}{(\delta r_+ + \delta r_-)_I} \right) c_{I-1}^+ - \left(\frac{A_I \overline{D}_I}{(\delta r_+ + \delta r_-)_I} \right) c_{I-1}^- \quad \text{(A-41)}$$

Coolant Boundary Treatment in Diffusion Solution

A coolant boundary model is meant to treat fission product transport to a cooling gas from a surface accounting for boundary layer effects (kinetic resistance to advection of the sorbate). This occurs, for example, in a PMR where graphite webbing interfaces to helium in coolant holes or in a PBR where the unfueled pebble matrix outer surface interfaces to helium. A sorption isotherm empirical model is applied for the cooled surface, and the diffusion equation of the node at the surface is modified. The key physical assumption is again vapor pressure equilibrium, but now between the pressures implied by (1) the sorbate concentration at the surface, and (2) the concentration in the boundary layer near the surface. Thus, for a given fission product species, the sorbate concentration at the cooled surface generates a vapor pressure that is equal to the vapor pressure in the boundary layer implied from an ideal gas assumption. Since the sorption isotherm empirical model describes the vapor pressure at the cooled surface, a nonlinear relationship is implied between the sorbate concentration at the surface and the concentration in the near-surface boundary layer. This condition is then effectively imposed as a nonlinear boundary condition, which requires linearization and iteration to solve in the context of the diffusion solution. Note this strategy resembles that of the TRAFIC code that is meant to model release of metallic fission products from an HTGR core [78]. The key difference in MELCOR is that a finite volume solution of the diffusion problem is used as opposed to alternative methods (e.g. of TRAFIC). Thus, the specific nodal difference equation (near the cooled surface) must be derived.

In the MELCOR diffusion solution, a coolant boundary interface may exist in a model at the end of the outermost zone (periphery of the given model) such that the outermost node on the outermost zone has a modified nodal equation. The intent is that the coolant boundary interface would be used in (1) a PMR type reactor in the matrix model in the unfueled outer zone representing graphite webbing, or (2) a PBR type reactor in the matrix model in the unfueled outer zone representing the outer region of a pebble. A full complement of sorption isotherm data must be given for the cooled surface.

The equilibrium vapor pressure assumption is borne out mathematically as:

$$P = P(\boldsymbol{C}_{surf}, T_{surf}) = P_{IDG}(c_{BL}, T_{surf}) \quad \text{(A-42)}$$

Where:

P = Vapor pressure at cooled surface [Pa]

P_{IDG} = Vapor pressure according to ideal gas law [Pa]

$\boldsymbol{C}_{surf}$ = Mass concentration at cooled surface [μmol/g]

c_{BL} = Volumetric concentration in boundary layer - cooled surface [kmol/m^3]

T_{surf} = Temperature at cooled surface [K]

According to the ideal gas law:

$$P_{IDG} = ART_{surf}c_{BL} \quad \text{(A-43)}$$

Where:

R = Universal gas constant = 8.314 [m^3*Pa/K/mol]

A = 1000, is a conversion factor, [mol] to [kmol]

Given the form of the sorption isotherm empirical model, this implied relation between $\boldsymbol{C}_{surf}$ and c_{BL} is nonlinear. Note that for a guessed $\boldsymbol{C}_{surf}$, a value for c_{BL} can be obtained. This suggests a solution involving an iterative strategy with linearization of the nonlinearity. Note also that the MELCOR diffusion solution works with volumetric concentration [kmol/m^3] so that the nodal difference equations must be posed in those terms. This implies that conversions must occur for volumetric concentration to and from mass concentration as dictated by the sorption isotherm model.

The linearized relationship between c_{surf} – volumetric concentration units - and c_{BL} is a first-order Taylor expansion of c_{BL} about a latest-iterate (guess) c_{surf}, i.e.:

$$c_{BL,i} = c_{BL,i-1} + \left(\frac{dc_{BL}}{dc_{surf}}\right)(c_{surf,i} - c_{surf,i-1}) \quad \text{(A-44)}$$

Where:

i = current iterate

$i-1$ = last iterate

The chain rule of differentiation, the sorption isotherm pressure/concentration relationship, the volume/mass concentration relationship, may be used to evaluate the

derivative $\frac{dc_{BL}}{dc_{surf}}$. Additionally, a formulation of kinetic resistance to diffusive mass transport in the near-surface boundary layer is required to evaluate the derivative:

$$\left(D\frac{dc}{dr}\right)_{surf} = G_{\infty}(c_{BL} - c_{\infty}) \tag{A-45}$$

Thus, the coolant interface boundary condition differs from the gas gap interface treatment in that kinetic resistance is not neglected.

$$\frac{dc_{BL}}{dc_{surf}} = \beta = \left(\frac{dc_{BL}}{dP}\right)\left(\frac{dP}{d\boldsymbol{C}_{surf}}\right)\left(\frac{d\boldsymbol{C}_{surf}}{dc_{surf}}\right) \tag{A-46}$$

Where:

$$\frac{dc_{BL}}{dP} = {}^{1}/_{RT_{surf}}$$

for

R = 8.314 [Pa*m^3/mol/K] ; T_{surf} in [K]

$$\frac{dP}{d\boldsymbol{C}_{surf}} = e^{X(T_{surf})}\left[\boldsymbol{C}_{tr}^{(Y(T_{surf})-1)} + Y(T_{surf})(\boldsymbol{C}_{surf})^{Y(T_{surf})}\right]$$

for $\boldsymbol{C}_{tr}$ and $\boldsymbol{C}_{surf}$ in [µmol/g]

$$\frac{d\boldsymbol{C}_{surf}}{dc_{surf}} = {}^{1}/_{\rho_{surf}}$$

for ρ_{surf} in [g/cm^3]

Units must be correct as noted above when computing β. The iterative strategy employed alongside the specialized nodal difference equation is:

- Guess or use last iterate outer surface volumetric concentration $c_{surf,i-1}$ [kmol/m^3]
- Evaluate the corresponding mass concentration $\boldsymbol{C}_{surf,i-1}$ [kmol/kg]
- Use the sorption isotherm model and the surface empirical constants to evaluate the vapor pressure corresponding to $\boldsymbol{C}_{surf,i-1}$
- Using the ideal gas assumption and vapor pressure equilibrium assumption, find the corresponding boundary layer volumetric concentration $c_{BL,i-1}$ [kmol/ m^3]

- Evaluate the boundary layer-to-surface concentration derivative $\beta = \frac{dc_{BL}}{dc_{surf}}$
- Using i-1 level information and the derivative β, assign values for modified nodal equation coefficients (a, b, c, d) and do a tridiagonal matrix solve to obtain an updated volumetric concentration at the surface, $c_{surf,i}$ and thus $\boldsymbol{C}_{surf,i}$
- Obtain updated estimate $c_{BL,i}$ from linearization equation
- Check pressure convergence, i.e. $P(\boldsymbol{C}_{surf,i}) \sim P(c_{BL,i})$ within tolerance
- If converged on vapor pressure then cease iteration, otherwise revise the guess of outer surface volumetric concentration by shuffling values $c_{surf,i-1} = c_{surf,i}$

Assuming the coolant boundary is at the outer surface of the N[th] node, the finite volume solution grid locally near the surface looks like:

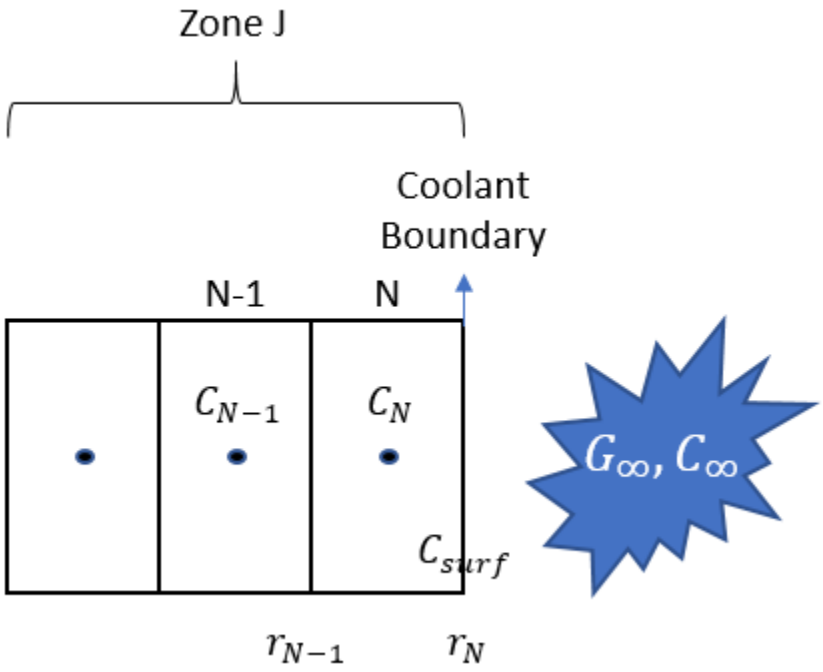


Figure A.3 Finite volume diffusion grid near coolant boundary

The coolant boundary interface exists beyond node N and hence at the end of zone J. The functional form of the interface condition results from (1) the sorption isotherm model, (2) the assumption of equilibrium vapor pressure, and (3) boundary layer diffusion mass transfer accounting for kinetic resistance. Mass flux continuity at the coolant boundary is an assumption, but the condition is complicated by the presence of an extra unknown: the boundary layer concentration. Thus, the iterative pressure convergence scheme is required to resolve both the surface and boundary layer concentrations.

Recall the generic nodal equation for nodes on the interior of the problem domain:

$$\Delta V_i \chi \left(\frac{C_i^n - C_i^{n-1}}{\Delta t} \right) = A_i D_i \frac{\partial C}{\partial r}_{r_i} - A_{i-1} D_{i-1} \frac{\partial C}{\partial r}_{r_{i-1}} - \lambda C_i \Delta V_i + \beta \Delta V_i \tag{A-47}$$

For surface node N:

$$\Delta V_N \chi \left(\frac{C_i^n - C_i^{n-1}}{\Delta t} \right) = A_N D_N \frac{\partial C}{\partial r}_{r_N} - A_{N-1} D_{N-1} \frac{\partial C}{\partial r}_{r_{N-1}} - \lambda C_N \Delta V_N + \beta \Delta V_N \tag{A-48}$$

Stated assumptions must be utilized to write a new expression for diffusion flux at the coolant boundary interface. Assigning int to the represent the boundary, letting i denote the current-iteration level, imposing mass flux continuity, and using the expression for boundary layer diffusion flux:

$$\left(D \frac{dc}{dr} \right)_{surf} = D_{int} \left(\frac{c_{surf,i} - c_N}{\delta r_{int}} \right) = G_\infty \left(c_{BL,i} - c_\infty \right) \tag{A-49}$$

Using the right-hand side of this equality and the linearization of $c_{BL,i}$, one can write $c_{surf,i}$ in terms of $c_{BL,i}$, $c_{BL,i-1}$, $\beta = \frac{dc_{BL}}{dc_{surf}}$, and $c_{surf,i-1}$:

$$c_{surf,i} = \frac{c_{BL,i} - c_{BL,i-1}}{\beta} + c_{surf,i-1} \tag{A-50}$$

Then, one can substitute for $c_{BL,i}$ such that the node N coolant boundary interface flux can be written purely in terms of $c_{BL,i-1}$, β, and $c_{surf,i-1}$:

$$c_{surf,i} = \frac{c_N + \left({G_\infty \delta r_{int}}/{D_{int}} \right) \left(\beta c_{surf,i-1} - c_{BL,i-1} + c_\infty \right)}{1 + {G_\infty \delta r_{int} \beta}/{D_{int}}} \tag{A-51}$$

$$D_{int} \left(\frac{c_{surf,i} - c_N}{\delta r_{int}} \right) = \frac{G_\infty \left(c_\infty - c_{BL,i-1} \right) + G_\infty \beta \left(c_{surf,i-1} - c_N \right)}{1 + {G_\infty \delta r_{int} \beta}/{D_{int}}} \tag{A-52}$$

For node N, nodal equation coefficients change:

$$a_N = \frac{-A_{N-1}\bar{D}_{N-1}}{(\delta r_+ + \delta r_-)_{N-1}}$$

$$b_N = \frac{A_{N-1}\bar{D}_{N-1}}{(\delta r_+ + \delta r_-)_{N-1}} + \left(\frac{A_N D_{int}}{\delta r_{int}}\right)\left(1 - \frac{\frac{D_{int}}{\delta r_{int}}}{1 + \frac{G_\infty \delta r_{int}\beta}{D_{int}}}\right) + \lambda \Delta V_N + \frac{\chi \Delta V_N}{\Delta t} \quad \text{(A-53)}$$

$$c_N = 0$$

$$d_N = \beta \Delta V_N + \frac{\chi \Delta V_N C_N^{n-1}}{\Delta t} - \left(\frac{A_N D_{int}}{\delta r_{int}}\right)\left(\frac{\left(\frac{G_\infty \delta r_{int}}{D_{int}}\right)\left(\beta c_{surf,i-1} - c_{BL,i-1} + c_\infty\right)}{1 + \frac{G_\infty \delta r_{int}\beta}{D_{int}}}\right)$$

A.3 Partition Coefficients

In the present context of fission product species diffusion, a partition coefficient describes the “jump” in concentration seen at/across an interface – usually a solid/gas interface. It is a less-complicated alternative to the sorption isotherm empirical model for capturing fission product species evaporation/transport across a gas gap or to coolant. The formulation consists of a prescribed jump condition on fission product species concentration at a given interface. The computational grid (between nodes I and $I+1$):

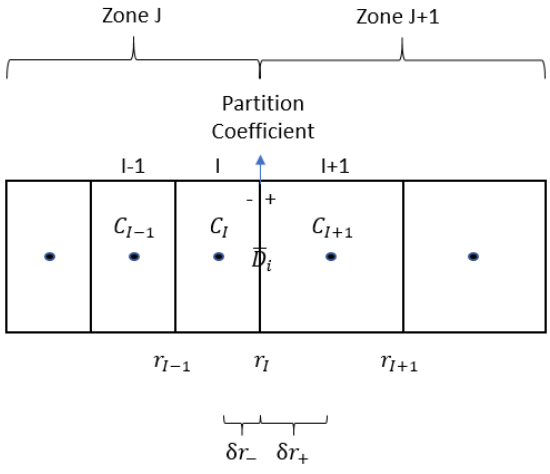


Figure A.4 Finite volume diffusion grid near partition coefficient interface

No iteration is required to resolve a nonlinear boundary condition, as the jump condition imposed at a partition coefficient interface is linear:

$$c^+ = \alpha c^- \tag{A-54}$$

Where:

α = Partition coefficient

c^- = Volumetric concentration [kmol/m^3], “inner part” of interface

c^+ = Volumetric concentration [kmol/m^3], “outer part” of interface

Nodal equation coefficients are altered for both nodes that border the partition coefficient interface. For node I, a nodal equation coefficient changes and the effective interface diffusion coefficient for the gas gap is modified:

$$D_{int} = \overline{D}_I = \frac{1}{f_{int}/_{D_{I+1}} + \alpha(1 - f_{int})/_{D_I}} \tag{A-55}$$

Where:

$$f_{int} = \frac{\delta r_+}{\delta r_+ + \delta r_-} \quad and \quad (1 - f_{int}) = \frac{\delta r_-}{\delta r_+ + \delta r_-}$$

Coefficient b changes:

$$b_i = \frac{\alpha A_I \overline{D}_I}{\Delta \bar{r}_I} + \frac{A_{I-1}\overline{D}_{I-1}}{\Delta \bar{r}_{I-1}} + \lambda \Delta V_I + \frac{\chi \Delta V_I}{\Delta t} \tag{A-56}$$

For node $I + 1$, the effective interface diffusion coefficient $D_{int} = \overline{D}_I$ is the same, but nodal equation coefficient a becomes:

$$a_{I+1} = \frac{-\alpha A_I \overline{D}_I}{\Delta \bar{r}_I} \tag{A-57}$$

Appendix B. Spheroidal Lower Head Derivations

A schematic of the new geometry is presented in Figure B.1. For an oblate spheroid the vertical semi-axis, c, is less than the horizontal semi-axes, a. A hemisphere is the limiting case of the oblate spheroid when c equals a.

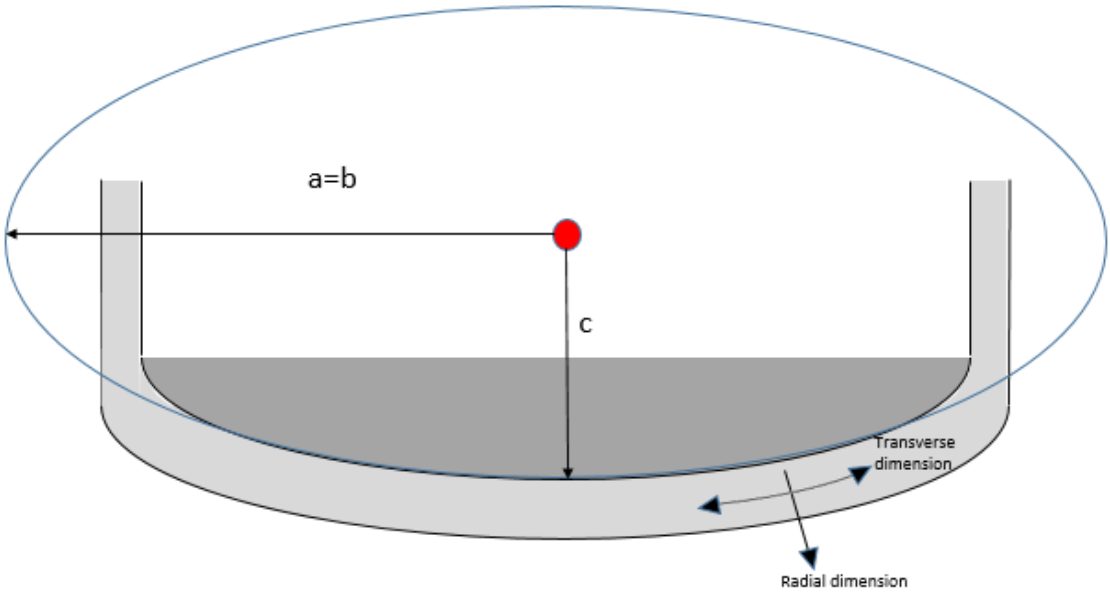


Figure B.1 Basic geometry of the spheroidal lower head.

For ease of calculation, it is assumed that both the inside and outside walls of the curved lower head vessel can be represented as concentric spheroids, with the user-specified thickness of the vessel wall applying along the principal axes of the spheroid. Elsewhere, the thickness of the vessel wall differs from the specified thickness to maintain the spheroidal concentricity, but the difference from the user-specified constant thickness is small if the wall is thin compared to the semi-axes of the spheroid. An exception to the concentricity assumption is in the modelling of lateral heat transfer in the curved lower head (i.e. thermal conduction between lower head wall segments along segment layers), for which it is assumed that the vessel has uniform thickness in calculating the heat transfer surface areas associated with conduction between each lower head segment. This is convenient but does not compromise the integrity of the calculation to a significant extent.

B.1 Lower Head Inside Surface Segment Boundary Heights and Angles

MELCOR requires the positions of the segment boundaries in order to assign the correct CVH control volume to interface with the segment heat structure. In Figure B.2, r is the

horizontal radial distance to a lower head segment boundary at P and h is the height of the segment boundary above the bottom of the lower head inside surface.

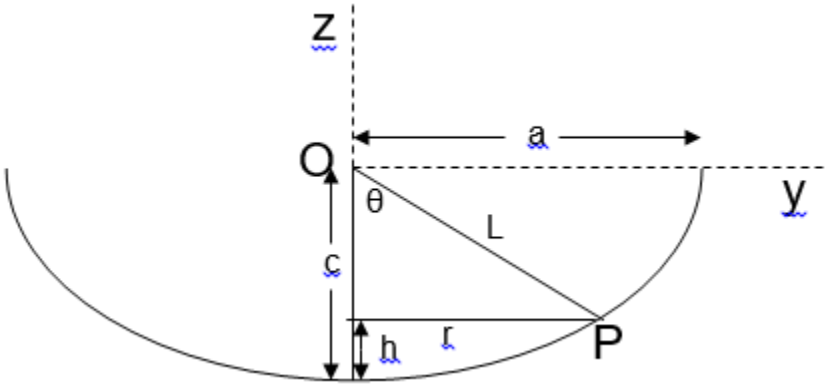


Figure B.2 Location of segment boundary on inside surface of lower head.

The equation of the ellipse representing the 2D profile of the inner wall is

$$\frac{y^2}{a^2}+\frac{z^2}{c^2}=1 \qquad \text{(B-1)}$$

Thus, at point P

$$\frac{r^2}{a^2}+\frac{(c-h)^2}{c^2}=1 \qquad \text{(B-2)}$$

which can be rearranged to give

$$h=c-c\sqrt{1-\frac{r^2}{a^2}} \qquad \text{(B-3)}$$

The angle to the vertical, subtended at the center of curvature, of the boundary of each lower head segment on the inside wall of the lower head, i.e., the parameter θ in Figure B.2, is defined by

$$\theta=\sin^{-1}\left(\frac{r}{L}\right) \qquad \text{(B-4)}$$

The value of L can be derived using the Pythagorean theorem:

$$L = \sqrt{(c-h)^2 + r^2} \qquad \text{(B-5)}$$

B.2 Segment Boundary Geometry

The geometric definition of a segment, in the context of a spheroidal lower head model, can be expressed in terms of the coordinates of the segment corners. Consider Figure B.3, which shows a segment boundary at angle θ (as derived using Equation B-4) running between the inside wall of the lower head and the segment layer of interest. t is the thickness at the pole and equator of the spheroidal lower head from the inside wall to the layer of interest. This is obtainable from the DZHEAD values (or DZWALL values for segments beyond the lower head transition point) which are input to the COR_LHN record. Thus, the semi-major and semi-minor axes of the ellipse for the layer of interest are a+t and c+t, respectively.

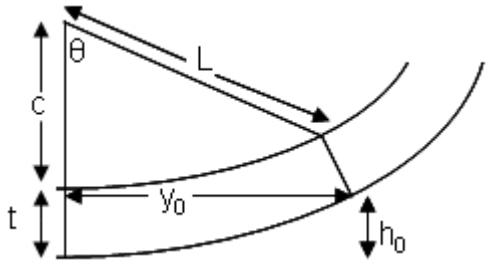


Figure B.3 Location of segment boundary on the layer of interest.

The location of the segment boundary on the layer of interest is derived by assuming that the boundary is normal to the inside wall. The gradient at any point on the inside wall ellipse is given by differentiating Equation (B-1):

$$\frac{dz}{dy} = \frac{-cy}{a^2\sqrt{1-\frac{y^2}{a^2}}} \qquad \text{(B-6)}$$

The gradient G of the inside wall ellipse at the segment boundary is thus

$$G = \frac{-cL\sin\theta}{a^2\sqrt{1-\frac{L^2\sin^2\theta}{a^2}}} \qquad \text{(B-7)}$$

Where

$$L = \frac{1}{\sqrt{\frac{\sin^2\theta}{a^2} + \frac{\cos^2\theta}{c^2}}} \tag{B-8}$$

The gradient m of the line forming the segment boundary normal to this is then

$$m = -\frac{1}{G} \tag{B-9}$$

Since this line passes through the point (Lsinθ, -Lcosθ) on the inside wall, the equation of the line forming the segment boundary is thus:

$$z + L\cos\theta = m\left(y - L\sin\theta\right) \tag{B-10}$$

The equation for the ellipse forming the layer of interest is:

$$\frac{y^2}{\left(a+t\right)^2} + \frac{z^2}{\left(c+t\right)^2} = 1 \tag{B-11}$$

By using Equation (B-10) to substitute for z in Equation (B-11) and solving the resulting quadratic equation in y, the value of y at which the segment boundary intersects this ellipse (y_0) can be obtained. The height of this point above the bottom of the elliptical layer of interest, h_0, is then given by applying Equation (B-3) to the ellipse defined by Equation (B-11):

$$h_0 = \left(c+t\right) - \left(c+t\right)\sqrt{1 - \frac{y_0^2}{\left(a+t\right)^2}} \tag{B-12}$$

y_0 and h_0 thus define the location of this segment corner.

The gradient of the ellipse defined by Equation (B-11) at the segment corner corresponding to the horizontal radial distance y_0 is given by:

$$G_0 = \frac{(c+t)\,y_0}{(a+t)^2\sqrt{1 - \frac{y_0^2}{(a+t)^2}}} \tag{B-13}$$

The angle of the slope of the ellipse at this point is then $\tan^{-1} G_0$. The value of this angle for the outer wall surface is needed for the calculation of convective heat transfer from the outer wall of the lower head vessel to the cavity pool.

B.3 Segment Transverse Surface Areas

The transverse surface area of each segment, used in the calculation of heat transfer between the segment and materials in the COR cell, is calculated from the difference in the areas of oblate spheroidal caps formed by the two ends of the segment. If the cap formed by the upper segment boundary at height h_2 in Figure B.4 has an area of A_{Cap2} and the cap formed by the lower segment boundary at height h_1 has an area of A_{Cap1}, then the transverse surface area of that segment is $A_{Cap2} – A_{Cap1}$.

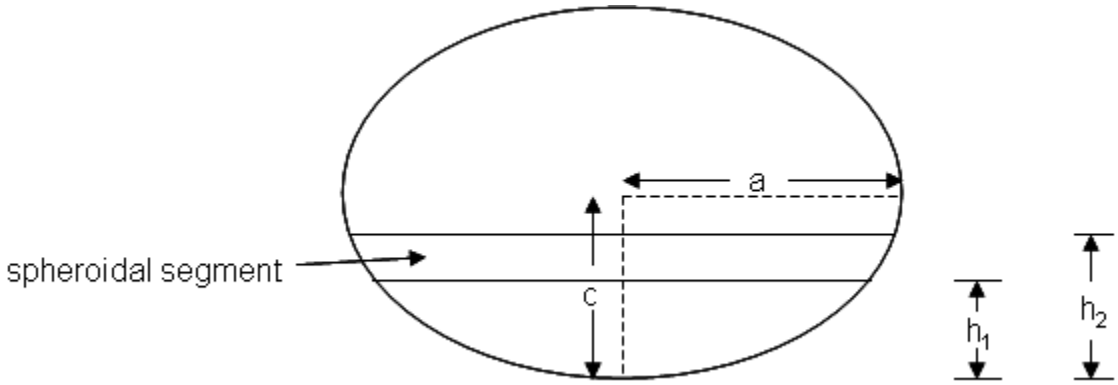


Figure B.4 Spheroidal segment between heights h_1 and h_2.

Consider the oblate spheroid represented by the equation

$$\frac{x^2 + y^2}{a^2} + \frac{z^2}{c^2} = 1 \qquad \text{(B-14)}$$

For the definition in Equation (B-14), the total surface area of the spheroid is given by

$$S_{total} = 2\pi a \int_{-c}^{c} \sqrt{1 + \frac{(a-c)(a+c)z^2}{c^4}}\, dz \qquad \text{(B-15)}$$

The surface area of a spheroidal cap of height h is therefore given by this integral between the limits –c and –c+h:

$$S_{cap} = 2\pi a \int_{-c}^{-c+h} \sqrt{1 + \frac{(a-c)(a+c)z^2}{c^4}}\, dz \tag{B-16}$$

$$= \pi a \left[\frac{c^2 \sinh^{-1}\left(\frac{z\sqrt{a^2-c^2}}{c^2} \right)}{\sqrt{a^2-c^2}} + z\sqrt{\frac{a^2 z^2}{c^4} - \frac{z^2}{c^2} + 1} \right]_{-c}^{-c+h} \tag{B-17}$$

$$= \pi a \left[a + \frac{c^2 \sinh^{-1}\left(\frac{\sqrt{a^2-c^2}}{c} \right)}{\sqrt{a^2-c^2}} + \frac{c^2 \sinh^{-1}\left(\frac{(h-c)\sqrt{a^2-c^2}}{c^2} \right)}{\sqrt{a^2-c^2}} + (h-c)\sqrt{\frac{a^2(h-c)^2}{c^4} - \frac{(h-c)^2}{c^2} + 1} \right] \tag{B-18}$$

Note that this equation cannot be used for the limiting case of a hemisphere (a=c), as this would result in zero over zero. In the limit of c→a, Equation (B-18) becomes equal to 2πah (making use of the fact that as x→0, $\sinh^{-1}(x) \rightarrow x$), which is the surface area of the corresponding spherical cap. For a hemisphere the existing (hemispherical) equations for the transverse surface area of a segment are used. Note also that the a and c in Figure B-4 and Equations (B-14) to (B-18) are the semi-axes of the spheroid containing the segment boundaries for the layer of interest, i.e. they correspond to a+t and c+t, respectively.

B.4 Segment Arc lengths

The arc length of each segment (i.e. the length along the ellipse between two segment corners in the same segment layer), is required for calculating the characteristic length used in the effective heat transfer coefficient for conduction between the segment layers. The expression for the heat transfer coefficient at the interface of adjacent segments m and n is

$$HTC = \frac{2}{\left(\frac{S_m}{TC_m} + \frac{S_n}{TC_n} \right)} \tag{B-19}$$

where S_m and S_n are the arc lengths of the segments m and n on the ellipse, and TC_m and TC_n are the thermal conductivities of the segments m and n. The arc length of a segment is derived by solving general elliptic integrals of the second kind for each of the two segment corners. This cannot be done analytically. Instead, an iterative solution using Bulirsch's algorithm has been implemented in MELCOR. This was taken from Chapter 6 of [81].

B.5 Segment Mean Distances

The mean distance from the center of the ellipse to the segment transverse boundary is needed in the application of the Larson-Miller failure model equations to model creep failure of the lower head. This is calculated by subdividing the segment transverse boundary into 10 subsegments and calculating a weighted average of the distance to each subsegment boundary point where the distance to each subsegment boundary point is weighted by the length of the segment associated with that boundary:

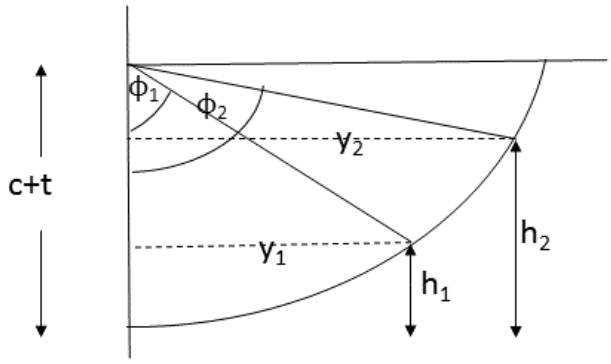


Figure B.5 Geometry of segment boundary ends.

In Figure B.5 the segment boundary ends are at (y_1, h_1) and (y_2, h_2). The boundary angles φ_1 and φ_2 are given by

$$\sin\phi_1 = \frac{y_1}{\sqrt{(c+t-h_1)^2 + y_1^2}} \qquad \text{(B-20)}$$

and

$$\sin\phi_2 = \frac{y_2}{\sqrt{(c+t-h_2)^2 + y_2^2}} \qquad \text{(B-21)}$$

The segment is subdivided so that each subsegment subtends the same angle at the center of the ellipse, i.e. the interval $\varphi_2 - \varphi_1$ is subdivided into 10 equal angular intervals.

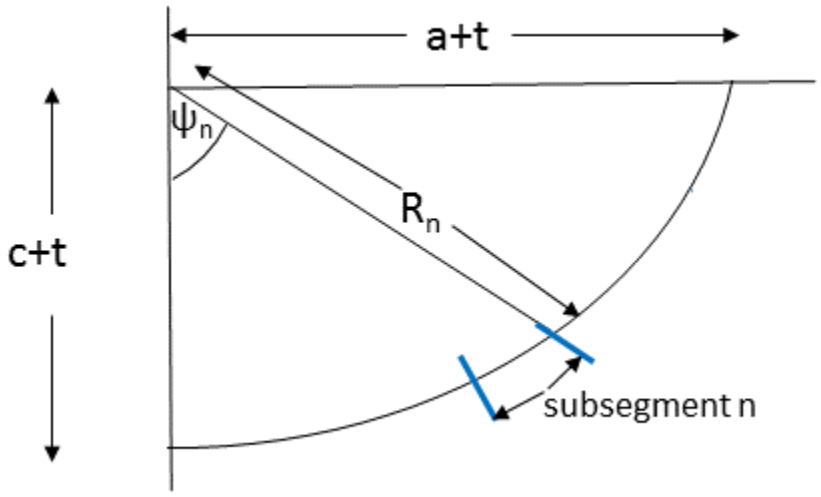


Figure B.6 Geometry of subsegment.

Figure B.6 shows the situation for the n'th subsegment of a segment, the upper boundary of which subtends an angle ψ_n to the vertical at the center of the ellipse, i.e.

$$\psi_n = (\phi_2 - \phi_1) \times \frac{n}{10} + \phi_1 \qquad \text{(B-22)}$$

The distance to the upper boundary from the center of the ellipse, R_n, can now be obtained as a function of ψ_n. Referring to Figure B.2, rearrange Equation (B-3) to give c-h, then substitute this for c-h in Equation (B-5), substitute Lsinθ for r and solve for L, giving

$$L = \frac{c}{\sqrt{1 - \sin^2\theta\left(1 - \frac{c^2}{a^2}\right)}} \qquad \text{(B-23)}$$

In the notation used in Figure B.6, this becomes

$$R_n = \frac{c + t}{\sqrt{1 - \sin^2\psi_n\left(1 - \left(\frac{c+t}{a+t}\right)^2\right)}} \qquad \text{(B-24)}$$

The arc lengths of each subsegment are then calculated in the same way as for the arc length of a complete segment, giving the arc length l_n for the n-th subsegment.

The mean, $\overline{R}$, weighted for the length of subsegment associated with each subsegment boundary, is then

$$\overline{R} = \frac{0.5\left(R_0 l_1 + R_1\left(l_1 + l_2\right) + \ldots + R_n\left(l_n + l_{n+1}\right) + \ldots + R_{10} l_{10}\right)}{\sum_{n=1}^{10} l_n} \qquad \text{(B-25)}$$

B.6 Segment Volumes

The volume of each segment layer can be calculated by considering the figure below, which shows two lower head layers for simplicity:

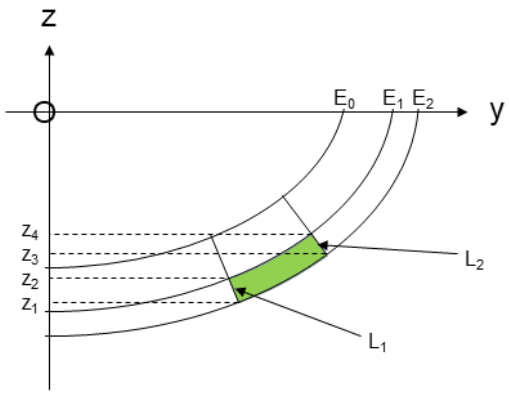


Figure B.7 Segment volume calculation.

It is assumed that the transverse boundaries of the segment layers are defined in the diagram by ellipses E_0, E_1, E_2 etc., i.e. the layers are bounded by concentric spheroids. Also, the radial boundaries of the segments are depicted in the diagram by straight lines L_1, L_2, etc., which are defined to be normal to the ellipse forming the inner surface, E_0. The four corners of the cell bounded by E_1, E_2, L_1, and L_2 have z-coordinates z_1, … z_4, as shown.

In general, the volume of revolution created by rotating the curve or line defined by the equation y=f(z) between z_n and z_m around the z-axis is given by

$$V\left(z_n \rightarrow z_m\right)(y) = \pi \int_{z_n}^{z_m} \left(y(z)\right)^2 dz \qquad \text{(B-26)}$$

which for the equation of an ellipse or a straight line can be readily integrated analytically. For the four corners of the cell bounded by E_1, E_2, L_1 and L_2 (the area shaded green in

Figure B.7) the volume of the cell V_{seg} can therefore be obtained by considering the volumes of revolution created by rotating the four boundaries of the cell around the z-axis, i.e.

$$V_{seg} = V\left(z_1 \to z_3\right)\left(E_2\right) + V\left(z_3 \to z_4\right)\left(L_2\right) - V\left(z_1 \to z_2\right)\left(L_1\right) - V\left(z_2 \to z_4\right)\left(E_1\right) \qquad \text{(B-27)}$$

where $V(z_1 \to z_3)(E_2)$ denotes the volume of revolution created by rotating the section of the ellipse E_2 between z_1 and z_3 around the z-axis, etc. The equations for the straight lines L_1 and L_2 and the z-coordinates of where they intersect the ellipses E_1 and E_2 can be calculated by applying Equations (B-10) to (B12).

B.7 Lateral Heat Transfer Surface Areas

Lateral heat transfer surface areas are required for modelling thermal conduction between lower head segments along segment layers. Figure B.8 shows the spheroidal lower head divided radially into three layers, with four "node" boundaries indicated by j=1 to j=4. The node numbers j increase in a radially inwards direction, so j=1 corresponds to the outside wall of the lower head. Note that a heat transfer area is associated with a specific node, j, not a length between two nodes, as the lower head temperatures are calculated at the layer boundaries (i.e. in the situation illustrated there are four temperatures per segment, located at each node j). The segment boundary of interest is at angle θ at the inner surface of the lower head.

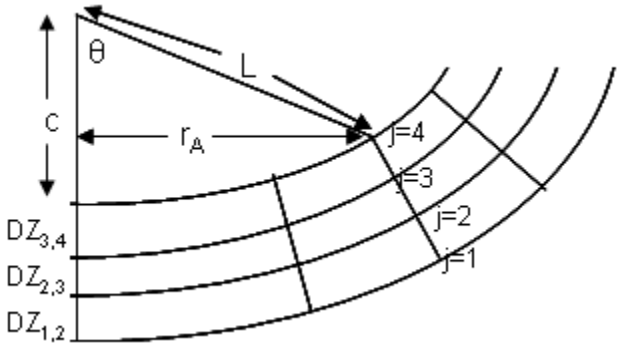


Figure B.8 Calculation of lateral heat transfer surface areas.

For the purposes of lateral heat transfer it is assumed that there is no lateral variation in the thickness of each segment, i.e. the thickness of a segment between j and j+1 is independent of θ (except at the crossing of the lower head transition boundary, above which the thicknesses become specified by DZWALL instead of DZHEAD in the COR_LHN record). This is the exception to the assumption generally made, that the segment layers form concentric spheroids.

The basic principle in calculating the heat transfer area is to calculate the length associated with a node along the segment radial boundary and then rotate this length around the vertical axis to produce an area.

The form of the equation used to calculate the length m associated with a node depends on the location of the node, j, with nodes on the surface (j=1 and j=4 in Figure B.8) being treated differently than internal nodes. For internal nodes the length, m, associated with the node is half of the segment thickness on each side of the node. Figure B.9 shows the situation for the node j=2:

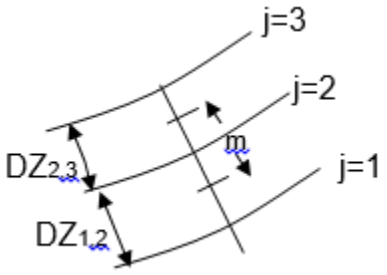


Figure B.9 Determination of length m associated with a node.

Thus

$$m = \frac{1}{2}\left(DZ_{1,2} + DZ_{2,3} \right) \qquad \text{(B-28)}$$

where $DZ_{x,y}$ is the thickness of the segment between j=x and j=y.

For surface nodes, the length associated with the node is half of the segment layer thickness inside the surface, as there is no lower head material outside of the surface. E.g. for the node j=1

$$m = \frac{1}{2} DZ_{1,2} \qquad \text{(B-29)}$$

The calculation of the horizontal distance r from the vertical axis to the middle of the length m can be broken down into two parts. The first part, r_A, is the horizontal distance from the vertical axis to the innermost node (j=4 in this example) lying on the segment radial boundary of interest. Thus

$$r_A = L \sin\theta \qquad \text{(B-30)}$$

where L is as shown in Figure B.8. From the equation of the ellipse of the inner surface, L is given by

$$L = \frac{1}{\sqrt{\frac{\sin^2\theta}{a^2} + \frac{\cos^2\theta}{c^2}}} \tag{B-31}$$

The second part of the horizontal distance, r_B, is the horizontal distance between the innermost node and the middle of the length m, as shown in Figure B.10:

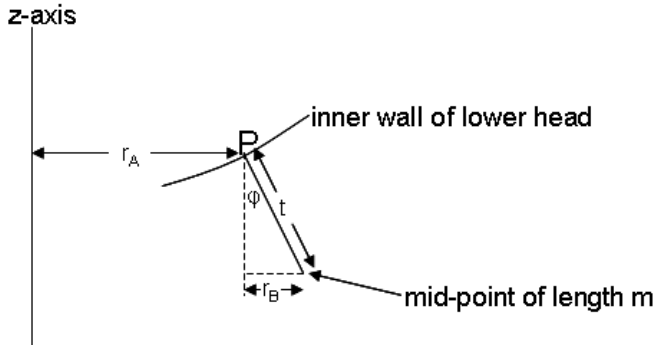


Figure B.10 Horizontal distance to middle of lateral length

In Figure B.10, t is the thickness, readily calculable from the DZs, between the innermost node at P and the middle of the length m, along the line which is normal to the ellipse at the innermost node. Then

$$r_B = t\sin\phi \tag{B-32}$$

Hence

$$r = r_A + r_B = L\sin\theta + t\sin\phi \tag{B-33}$$

Since the length t in the diagram is normal to the ellipse at P, tanφ is the gradient of the ellipse at this point, which is given by

$$\tan\phi = \frac{cr_A}{a^2\sqrt{1-\frac{r_A^2}{a^2}}} = \frac{cL\sin\theta}{a^2\sqrt{1-\frac{L^2\sin^2\theta}{a^2}}} \tag{B-34}$$

The area for lateral heat transfer is then:

$$A = 2\pi rm \tag{B-35}$$

B.8 Volume of COR Cells in Contact with Lower Head

To calculate the volume of the COR cell interfacing each segment on the inside surface of the lower head consider Figure B.11 below. Let h_1 and h_2 be the heights, above the base of the inside surface of the lower head, of the lower and upper ends of the segment. These heights intersect the lower head surface at horizontal radii r_1 and r_2 respectively, as shown.

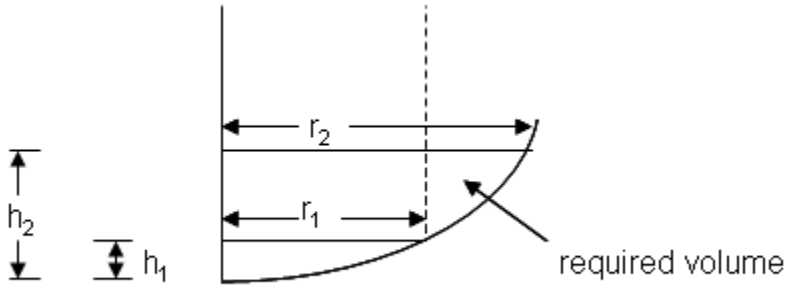


Figure B.11 Boundaries and volume for COR cell in contact with lower head.

Re-arranging Equation (B-3), the horizontal radius r is related to the height h relative to the base of the spheroid by

$$r^2 = \frac{a^2 h}{c^2}\left(2c - h\right) \tag{B-36}$$

The volume of a spheroidal cap of height h is given by

$$V = \frac{\pi a^2 h^2}{3c^2}\left(3c - h\right) \tag{B-37}$$

The volume of the spheroidal segment between h1 and h2 can be written as the difference between two concentric spheroidal caps:

$$V = \frac{\pi a^2}{3c^2}\left[h_2^2\left(3c - h_2\right) - h_1^2\left(3c - h_1\right)\right] \tag{B-38}$$

The required volume, the volume of the COR cell interfacing the segment, is the volume of the portion of this segment bounded by r_1 and r_2:

$$V = \frac{\pi a^2}{3c^2}\left[h_2^2\left(3c - h_2\right) - h_1^2\left(3c - h_1\right)\right] - \pi r_1^2\left(h_2 - h_1\right) \quad \text{(B-39)}$$

B.9 Surface Area of Molten Pool

For a spheroidal lower head, if the top surface of any molten pool is located within the spheroidal part of the lower head then the surface area of the top surface can be derived by re-arranging Equation (B-2) for r and then calculating using πr^2:

$$A = \pi a^2\left(1 - \left(1 - \frac{h}{c}\right)^2\right) \quad \text{(B-40)}$$

where h is the depth of the pool.

References

1 F. Kreith, Principles of Heat Transfer, 3rd Edition, Intext Educational Publishers, New York, NY, pp. 251-273 (1973).

2 L. J. Ott, C. F. Weber, and C. R. Hyman, “Station Blackout Calculations for Browns Ferry,” Proceedings of the Thirteenth Water Reactor Safety Research Information Meeting, Gaithersburg, MD (October 1985).

3 R. O. Wooton, P. Cybulskis, and S. F. Quayle, “MARCH 2 (Meltdown Accident Response Characteristics) Code Description and User's Manual,” NUREG/CR-3988, BMI-2115 (August 1984).

4 D. L. Hagrman, G. A. Reymann, and R. E. Mason, MATPRO-VERSION 11 (Revision 1) A Handbook of Materials Properties for Use in the Analysis of Light Water Reactor Fuel Rod Behavior, NUREG/CR-0497 and TREE-1280 Rev. 1 (February 1980).

5 L. J. Ott, Thermal-Hydraulic Test Facility Bundle 3 In-Core Instrumentation and Operating History, NUREG/CR-2609, Chapter 8 (August 1982).

6 C. B. Ludwig and C. C. Ferriso, “Prediction of Total Emissivity of Nitrogen-Broadened and Self-Broadened Hot Water Vapor,” J. Quant. Spectrosc. Radiat. Transfer, 7, pp. 7-26 (1982).

7 J. P. Holman, Heat Transfer, McGraw-Hill, Inc., New York, NY, pp. 305-307 (1976).

8 R. K. Cole, Jr. and M. F. Young, “Status of Implementation of a Reflood Model in MELCOR,” Letter Report to Ali Behbahani, US Nuclear Regulatory Commission, (March 19 2001).

9 S. S. Dua and C. L. Tien, Intl. J. Heat and Mass Transfer 20, pp.174-176 (1977).

10 J. J. Carbajo and A. D. Siegel, “Review and Comparison among the Different Models for Rewetting in LWR’s,” Nucl. Eng. Design 58, No. 1, p.33 (1980).

11. P. Zehner and E.U. Schlunder, “Thermal conductivity of granular materials at moderate temperatures (in German),” Chemie. Ingr.-Tech. **42**, p.933 (1970).

12. C.T. Hsu, P. Cheng, K.W. Wong, “Modified Zehner-Schlunder models for stagnant thermal conductivity of porous media,” Intl. J. Heat Mass Transfer, Vol.37, No.17, p.2751 (1994).

13. IAEA, *Heat Transfer and Afterheat Removal for Gas Cooled Reactors Under Accident Conditions"*, Section 4.2.2, INET Analysis of SANA-1 Experiment Benchmark Problems, IAEA-TECDOC-1163, IAEA, Vienna, Austria (2000).

14. E. Tsotsas and H. Martin, "Thermal Conductivity of Packed Beds: A Review," Chem. Eng. Process. **22**, p.19 (1987).

15. G. Breitbach and H. Barthels, "The Radiant Heat Transfer in the High Temperature Reactor Core After Failure of the Afterheat Removal Systems," Nucl. Tech. Vol. 49 p.392 (1980).

16. E.R. Eckert and R. M. Drake, Jr., *Analysis of Heat and Mass Transfer*, McGraw-Hill (1972).

17. M. P. Manahan, "An Improved Zircaloy-Steam Reaction Model for Use with the MARCH 2 (Meltdown Accident Response Characteristics) Code," Proceedings of the International Meeting on Light-Water Reactor Severe Accident Evaluation, Cambridge, MA (August 1983).

18. F. W. Dittus and L. M. K. Boelter, University of California Pubs. Eng., 2, p. 433 (1930).

19. M. Jacob, Heat Transfer, Vol. I, John Wiley & Sons, Inc., New York, NY (1949).

20. R. B. Bird, W. E. Stewart, and E. N. Lightfoot, Transport Phenomena, John Wiley & Sons, Inc., New York, NY (1960).

21. R. J. Lipinski, A Model for Boiling and Dryout in Particle Beds, NUREG/CR-2646, SAND82-0765 (June 1982).

22 K. Kendall, J.P. Monroe, "The Viscosities of Liquids, III Ideal Solutions of Solids in Liquids", Journal of the American Chem. Soc., Vol. 39, No. 8 p.1802 (September 1917).

23 J.M. Bonnet, J.M. Seiler, "In-Vessel Corium Pool Thermalhydraulics for the Bounding Cases," RASPLAV Seminar, Munich, 2000.

24 Theofanous T.G., Angelini S., "Natural Convection for In-Vessel Retention at Prototypic Rayleigh Numbers", Eighth International Topical Meeting on Nuclear Reactor Thermal-Hydraulics, Kyoto, Japan, September 30-October 4, 1997.

25 Globe S., Dropkin D., "Natural-Convection Heat Transfer in Liquids Confined by Two Horizontal Plates and Heated from Below", J. Heat Transfer, 81, pp24-28, 1959.

26 J. Stefan, *SB Wien Akad. Mat. Natur.,* **98** 473-84, 965-83, 1889.

27. G. W. Parker and A. L. Sutton, Jr., “Boron Control Material Behavior in Large-Scale, Core-Melt Experiments,” presented at the Severe Fuel Damage and Source Term Research Program Review Meeting, Oak Ridge Associated Universities Conference Center, Oak Ridge, TN, April 7-10, 1986.

28. V. F. Urbanic and T. R. Heidrich, “High-Temperature Oxidation of Zircaloy-2 and Zircaloy-4 in Steam,” J. Nuc. Matls., 75, pp. 251-261 (1978).

29. A. S. Benjamin, D. J. McCloskey, D. A. Powers, and S. A. Dupree, Spent Fuel Heatup Following Loss of Water During Storage, SAND77-1371, NUREG/CR-0649, Sandia National Laboratories, Albuquerque, NM, March 1979.

30. J. F. White et al., “Fifth Annual Report--High Temperature Material Programs, Part A,” GEMP-400A (February 1966).

31 M.B. Richards, A.W. Barsell, “Catalysis Effects by Impurities on the Steam-Graphite Reaction under Nuclear Reactor Accident Conditions,” GA-C19161 Rev. 1 (1988).

32. M.B. Richards et al., “A Computational Model for Graphite Oxidation under Nuclear Reactor Accident Conditions,” AIChE Symposium Series, **83** 257, p.363 (1987).

33 F. Gelbard, “Graphite Oxidation Modeling for Application in MELCOR“, SAND2008-7852, January, 2009

34 E.L. Fuller, M.J. Okoh, J. Nucl. Mat. 240, p.241 (1997).

35. E. Merrill, R. Moore, P. Sharpe, “MELCOR Development for NGNP,” NRC/INL Mtg. on Methods for VHTRs, Idaho Falls, ID (2008).

36 E. Kim, H. NO, “Experimental study on the oxidation of nuclear graphite and development of an oxidation model”, J. Nucl Matls 349, p182 (2006).

37 F. Gelbard, “Graphite Oxidation Modeling for Application in MELCOR“, SAND2008-7852, January, 2009

38. Fred Griffin, “BWR Control Blade Channel Box Interaction and Melt Relocation Models for SCDAP,” Letter Report ORNL/NRC/LTR-92/12/R2 to Dr. Yi-Shung Chen, Accident Evaluation Branch, Division of Systems Research, RES USNRC, December 30, 1993.

39. G. Eriksson, “Thermodynamic Studies of High Temperature Equilibria: XII. SOLGASMIX, a computer program for calculation of equilibrium compositions in multiphase systems,” Chemica Scripta, 8, 1975, pp. 100-103.

40. M. Steinbruck et al., Results of the B_4C Control Rod Test QUENCH-07, FZKA-6746, Forschungszentrum Karlsruhe, Karlsruhe, Germany (May 2004).

41. M. Steinbruck et al., Degradation and Oxidation of B_4C Control Rod Segments at High Temperatures, FZKA-6980, Forschungszentrum Karlsruhe, Karlsruhe, Germany (May 2004).

42. M. Steinbruck et al., Experiments on the Oxidation of Boron Carbide at High Temperatures, FZKA-6979, Forschungszentrum Karlsruhe, Karlsruhe, Germany (May 2004).

43. R. K. Cole, Jr., MELCOR Calculation of International Standard Problem ISP45 at Sandia National Laboratories, Letter Report (July 2001).

44. "Reducing BWR Power by Water Level Control During an ATWS, a Quasi-Static Analysis," NSAC-69, S. Levy, Inc. Final Report (May 1984).

45. P. Hofmann et al., "Reactor Core Materials Interactions at Very High Temperatures," Nuclear Technology, 87, pp. 146-186, August 1989.

46. W. Hering and K. Muller, "Modelling of Eutectic Interactions in KESS-III (Module EUTECT)," International CORA Workshop 1992, Karlsruhe, FRG, October 5-8, 1992.

47. R. K. Cole, et al., CORCON-Mod2: A Computer Program for Analysis of Molten-Core Concrete Interactions, NUREG/CR-3920, SAND84-1246, August 1984, pp. 64-65.

48. M.R. Denman, "Development of the SharkFin Distribution for Fuel Lifetime Estimates in Severe Accident Codes", American Nuclear Society Winter Meeting, Las Vegas, NV, November 6-10, 2016.

49. D.A. Powers, "Thermochemistry of Core Debris: Partitioning of Uranium and Fission Products among Condensed Core Debris Phases", 2004.

50. D. A. Petti, "Silver-Indium-Cadmium Control Rod Behavior in Severe Reactor Accidents", Nucl. Tech. 84, pp. 128-151 (February1989).

51. D. A. Petti, Silver-Indium-Cadmium Control Rod Behavior and Aerosol Formation in Severe Reactor Accidents, NUREG/CR-4876, EGG-2501, Idaho National Engineering Laboratory, Idaho Falls, ID (1987).

52. R. J. Roark and W. C. Young, Formulas for Stress and Strain, Fifth Edition, McGraw-Hill Book Company, New York, NY, 1982, Case 8 of Table 26.

53. Ibid. The equation is extracted from Case 10 of Table 24.

54. R. K. Cole, Jr., “Modifications to MELCOR to Improve Modeling of Core Structure Supports, Final Implementation Report,” letter report to Mr. John Ridgely, USNRC, April 15, 1999.

55. R. J. Roark and W. C. Young, op. cit., Case 2a of Table 24.

56. Nuclear Systems Materials Handbook, Vol. 1 – Design Data, “Yield strength, minimum expected,” 1974.

57. T. H. Lin, Theory of Inelastic Structures, John Wiley and Sons, NY, 1983.

58. Nuclear Systems Materials Handbook, Vol. 1 – Design Data, “Young’s modulus, static,” 1974.

59. F. R. Larson and J. Miller, “A Time-Temperature Relationship for Rupture and Creep Stress,” Transactions of the ASME, pp. 765-775, July 1952.

60. American Society of Mechanical Engineers, “ASME Boiler and Pressure Vessel Code Case N-47-22,” April 5, 1984.

61. M. S. El-Genk and Z. Guo, “Transient Critical Heat Flux for Inclined and Downward-Facing Flat Surfaces,” ANS Proceedings, HTC-6 Volume 6, 1992, National Heat Transfer Conference, August 9-12, 1992, San Diego.

62. T. Y. Chu, “A Correlational Approach to Turbulent Saturated Film Boiling,” Journal of Heat Transfer, Volume 115, November 1993.

63. K. B. Cady, V. K. Dhir and R. J. Witt, “Peer Review of Models for Lower Vessel Head Heat Transfer and Larson-Miller Failure Criterion Proposed for Implementation into MELCOR,” ERI/NRC 94-202 March 1994.

64. T.Y. Chu, et al., Lower Head Failurer Experiments and Analyses, NUREG/CR5582, SAND98-2047, February 1999.

65. S. Timoshenko and S. Woinowsky-Krieger, Theory of Plates and Shells, Mcgraw-Hill Book Company, New York, NY, Second Edition, 1959. Parameters are taken from Table 58.

66. R. C. Schmidt et al., MELPROG-PWR/MOD1 Models and Correlations, SAND89-3123, June 1992.

67. M. Pilch and W. W. Tarbell, High Pressure Ejection of Melt from a Reactor Pressure Vessel—the Discharge Phase, NUREG/CR-4383, SAND85-0012 (September 1985).

68. P.E. MacDonald, ed., “NGNP Point Design – Results of the Initial Neutronics and Thermal-Hydraulic Assessments During FY-03,” INEEL/EXT-03-00870, INL, Idaho (2003).

69. S. Glasstone and A. Sesonske, *Nuclear Reactor Engineering*, Van Nostrand Reinhold (1967).

70. G. Keepin, *Physics of Nuclear Kinetics*, Addison-Wesley (1965).

71. T.A. Porsching, “Numerical Solution of the Reactor Kinetics Equations by Approximate Exponentials,” *Nucl. Sci. and Eng.* **25**, p.183 (1966).

72. S.B. Rodriguez et al., “Development of Design and Simulation Model and Safety Study of Large-Scale Hydrogen Production Using Nuclear Power,” SAND2007-6218, Sandia National Laboratories, Albuquerque, NM (2007).

73. P. Brocheny, *Modeling of the Transient Behavior of Heat Pipes with Room-Temperature Working Fluids*, Ph. D. Dissertation, Clemson University (2006).

74 . J. K. Fink and L. Leibowitz, THERMODYNAMIC AND TRANSPORT PROPERTIES OF SODIUM LIQUID AND VAPOR, Reactor Engineering Division, Argon National Laboratory, ANL/RE-95/2, Jan 1995.

75. T. Masuda, H. Imai, and S. Korenaga, "Thermodynamic properties of potassium" *Electrical Engineering in Japan*, Vol. 105, No.6, pp. 220-227, 1985.

76. M. Sanchez, HTPIPE: A Steady-State Heat Pipe Analysis Program, A User’s Manual, LA-11324-M, Group MEE-13, Los Alamos National Laboratory, undated.

77. Amir Faghri, HEAT PIPE SCIENCE AND TECHNOLOGY, Taylor and Francis 1995.

78 P.D. Smith and F. Tzung, “TRAFIC”, General Atomics, San Diego, CA (1978).

79 F. Gelbard, “Analytical Modeling of Fission Product Releases by Diffusion from Multicoated Fuel Particles “, SAND2002-3966, March, 2003

80. B. E. Boyack, et al., MELCOR Peer Review, LA-12240, Los Alamos National Laboratory (March 1992).

81. W.H. Press, et al., Numerical Recipes in C, Cambridge University Press, Cambridge, 1991.

Thermal Hydraulic (CVH and FL) Packages

Two packages in the MELCOR code, the Control Volume Hydrodynamics (CVH) package and the Flow Path (FL) package, are responsible for modeling the thermal-hydraulic behavior of coolant liquids and gases. The former is concerned with control volumes and their contents, the latter with the connections that allow transfer of these contents between control volumes. The distinction between CVH and FL is useful primarily for discussion of MELCOR input and output. It is frequently ignored in this reference manual, where many aspects of the thermal-hydraulic modeling are described without concern for which package contains the relevant coding.

If phenomena modeled by other packages in MELCOR influence thermal-hydraulic behavior, the consequences are represented as sources and sinks of mass, energy, or available volume, or as changes in the area or flow resistance of flow paths in CVH. [Changes involving flow paths may currently be handled only through use of the Control Function (CF) package.]

Equations of state for the hydrodynamic materials are contained in the Control Volume Thermodynamics (CVT) package, which in turn makes use of the water properties (H2O) and NonCondensible Gas (NCG) packages.

This reference manual describes the assumptions, models, and solution strategies used in the various subroutines which make up the CVH and FL packages. The user is referred to the appropriate reference manuals and other documentation for details of the equations of state and the boundary conditions provided by other packages in MELCOR.

Contents

List of Figures

List of Tables

1. Introduction

Thermal-hydraulic processes interact with and are coupled to all aspects of accident phenomenology. In the MELCOR code, thermal-hydraulic data calculated by the Control Volume Hydrodynamics (CVH) and Flow Path (FL) packages provide boundary conditions to other phenomenological packages such as Burn (BUR), Cavity (CAV), Core (COR), Fuel Dispersal Interactions (FDI), and Heat Structures (HS). These packages, in turn, calculate sources and sinks of mass and energy for CVH. COR and HS also calculate changes to the volumes available to hydrodynamic materials. In some cases, CVH results are used directly by another package. The RadioNuclide (RN) package uses CVH results for advection to transport aerosols and vapors from one calculational volume to another; RN also uses CVH results for the liquid water content of the atmosphere (fog) as the water content of aerosols, rather than integrating a separate equation for condensation and evaporation. Therefore, even though the primary interest in accident research is *not* solely thermal hydraulics, the thermal-hydraulic modeling in CVH and FL forms the backbone of the MELCOR code.

The choice of modeling in CVH and FL was influenced by several often conflicting requirements. The packages were desired to be computationally fast but also reliable and accurate. They should not produce minor nonphysical variations in behavior that would adversely affect the performance of other packages and should not be unduly sensitive to such variations in the conditions calculated by other packages. They should permit great flexibility in nodalization to simplify sensitivity studies and should extract the maximum amount of information from coarse nodalizations while allowing more detailed ones for comparison to more specialized codes. In addition, they should be user friendly with respect to input.

The calculational method chosen uses a control volume/flow path approach similar to RELAP4 [1], HECTR [2], and CONTAIN [3]. The same models and solution algorithms are used for all volumes (i.e., the primary, secondary, and containment volumes are modeled consistently, and the resulting equations are solved simultaneously). Within the basic control volume formulation, the treatment is quite general; unlike the MAAP code [4], no specific nodalization is built in. No component models are explicitly included; pipes, vessels, pressurizers, steam generators, for example, are built through user input from control volumes, flow paths, and elements of other packages such as heat structures. Control logic used to simulate active or passive systems is introduced using control functions. (There are separate models for a few special safety systems including fan coolers and containment sprays.) We anticipate that, as experience with MELCOR grows, a set of “standard” nodalizations are developed, validated, and employed for most calculations. However, the freedom exists to investigate sensitivities to variations in nodalization (and to develop representations of systems) entirely from code input, without modification to MELCOR itself.

A semi-implicit (linearized) formulation of the governing equations is used to permit timesteps greater than the acoustic Courant limit. The numerical-solution technique is similar to that in RELAP4 [1], with two major differences: (1) MELCOR uses a full two-fluid treatment rather than the drift-flux formulation of RELAP4 and (2) the resulting equations are iterated when necessary so that the result is fully implicit with respect to pressures used in the momentum equation. A significant feature of this method is that the resulting equations are exactly conservative (to within machine roundoff) with respect to masses and to thermal energy.

All hydrodynamic material in a MELCOR calculation, together with its energy, resides in *control volumes*. “Hydrodynamic material” includes the coolant (water), vapor (steam), and noncondensible gases; it does not include the core or core debris, other structures, fission products, aerosols, or water films on heat structures. The hydrodynamic materials are divided into two independent fields referred to as *pool* and *atmosphere*. The names refer to the frequently employed picture of separation under gravity within a control volume, but the actual interpretation is less restrictive. The shape of the volume is defined in enough detail to allow the elevation of the pool surface to be determined. Beyond this, a control volume has no internal structure and is characterized by a single pressure and two temperatures; one temperature for the pool and one for the atmosphere. (Of course, various constitutive models in CVH/FL and other packages may *infer* greater detail such as boundary and interface temperatures, and temperature or pressure gradients, but they are not part of the CVH/FL database.)

The control volumes are connected by *flow paths* through which the hydrodynamic materials may move without residence time, driven by a separate momentum equation for each field. Each control volume may be connected to an arbitrary number of others and parallel flow paths (connecting the same pair of volumes) are permitted. There are no restrictions on the connectivity of the network built up in this way. Both pool and atmosphere, pool only, or atmosphere only may pass through each flow path based on the elevations of the pool surfaces in the connected control volumes relative to the junctions with the flow paths. Appropriate hydrostatic head terms are included in the momentum equations for the flow paths, allowing calculation of natural circulation.

The control volumes and flow paths may be used to model physical systems in a variety of ways. In some cases, the control volumes may correspond to physical tanks, with the flow paths representing pipes (of negligible volume) connecting them. In others, the volumes may be geometrical regions—perhaps portions of larger physical rooms—with the flow paths representing the geometrical surfaces separating them. Representations approaching a finite difference approximation to the one-, two-, or three-dimensional hydrodynamic equations may be built up using the latter approach. However, because the momentum equation for each flow path is only one-dimensional and there is no momentum associated with a control volume, multidimensional effects associated with advection of momentum (“momentum flux”) cannot be correctly calculated. (The one-dimensional momentum flux term for the direction of flow may be optionally included.)

In addition to phenomena within the CVH and FL packages, calculations performed in other packages in MELCOR may lead to sources and sinks of mass or energy in control volumes or to changes in the volume available to hydrodynamic materials. These are imposed as numerically explicit boundary conditions in CVH/FL. In addition to heat sources from the Decay Heat (DCH) package, mass and energy source/sinks include heat from the HS, COR, CAV, and FDI packages, water from condensation or evaporation of films or melting of ice in the HS package or deposition of aerosol droplets in the RN package, and various gas sources from outgassing of structures in the HS package or from concrete ablation in the CAV package.

Oxidation chemistry in the COR and BUR packages is modeled as a sink of reactants (water vapor or oxygen in COR, hydrogen or carbon monoxide in BUR) and a source of reaction products (hydrogen in COR, water vapor or carbon dioxide in BUR). All equations of state referenced by the Control Volume Thermodynamics (CVT) package employ consistent thermochemical reference points, with the heat of formation included in the enthalpy functions as in JANAF tables [5]. Therefore, no energy source is involved in such a reaction; total energy is conserved, and the “heat of reaction” associated with changes in chemical bonding energies appears as sensible heat because of changes in the reference-point enthalpy of the system.

Changes in available volume result from such phenomena as candling (relocation of molten core materials by downward flow along fuel rods) and core collapse, which move nonhydrodynamic materials into or out of a control volume. Nonhydrodynamic materials may be moved by other packages either independently of CVH/FL flows (e.g., core relocation) or piggybacked on the flows (e.g., motion of aerosols and associated radionuclides).

2. Basic Control Volume Concepts

The basic concepts, definitions, and terminology associated with control volumes are described in this section. Discussion of most of the details of the models is deferred until after the conservation equations have been presented and discussed.

2.1 Control Volume Geometry

The spatial geometry within a control volume is defined by a *volume/altitude* table. (The terms “altitude” and “elevation” are used interchangeably in this manual.) Each point in the table gives an altitude and the total volume available to hydrodynamic materials in the CVH package below that altitude in that control volume. In this usage, “altitude” means elevation with respect to some reference point. This reference point is arbitrary but must be consistent throughout all input for any problem (i.e., the same for all CVH, FL, COR, HS, and other data) to allow differences in elevation to be evaluated correctly.

The volume at the lowest altitude must be zero; the volume is assumed to be a linear function of altitude between table entries. This is equivalent to assuming a piecewise-

constant cross-sectional area as illustrated in Figure 2.1, which shows a simple geometric volume and a plot of the corresponding four-point volume/altitude table. Note that the independent variable, altitude, is plotted vertically to facilitate comparison with the sketch.

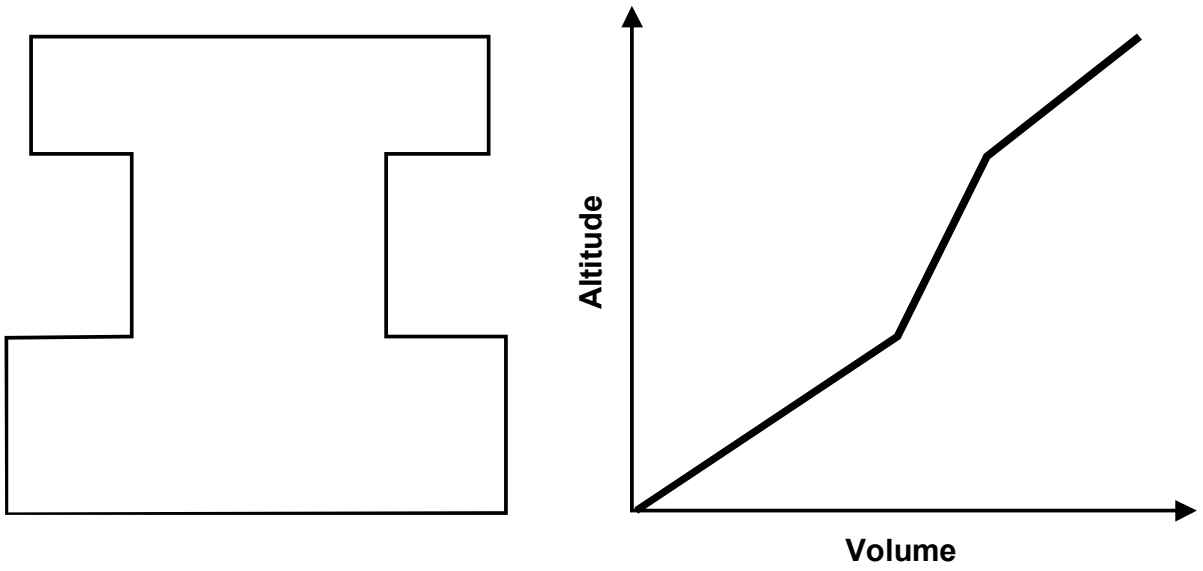


Figure 2.1 Relation of spatial volume to volume/altitude table

In addition to the hydrodynamic volume, a control volume may also contain *virtual volume* associated with nonhydrodynamic material (in some other package) that occupies space but is subject to relocation. If this material is relocated, the space that it occupied becomes available to hydrodynamic materials. The principal example of this is the core, which initially occupies a large volume in the primary system, but which may melt down and relocate to another part of the primary or containment system. This frees some or all the original space to be occupied by hydrodynamic materials, while denying space to such materials in the new location.

The initial hydrodynamic volume is defined by input of the CV_VAT record to CVH in MELGEN, and the initial virtual volume is defined by input to other packages. Their sum is calculated in MELGEN for the set of altitude points in the CVH input to define a total volume/altitude table that becomes part of the CVH database and does not change with time. The virtual volume is also carried in the CVH database as a volume/altitude table defined for the set of altitudes input to CVH. The difference between total and virtual volume is available to hydrodynamic materials and initially coincides with that specified in CVH input.

Virtual volume is illustrated by Figure 2.2, where the total volume is shown in grey and the virtual volume as the white space (i.e., volume on the RHS graph) between the virtual volume and the cell boundary. Note that the points in the virtual-volume/altitude table

correspond to the altitudes in the CVH database and not to those in whatever package defined the occupied (shaded) region.

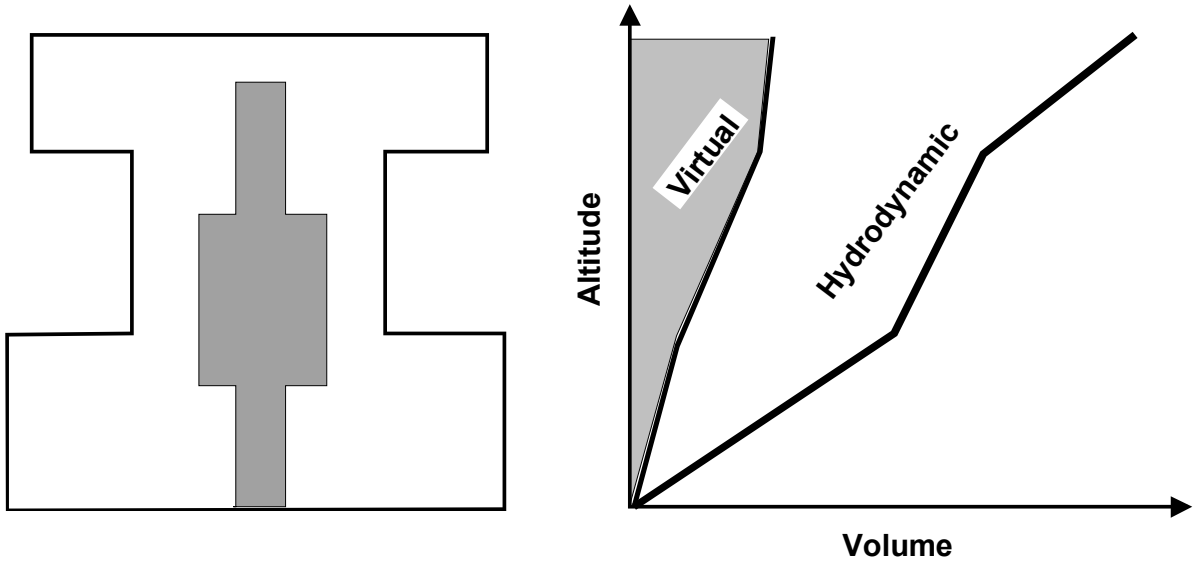


Figure 2.2 Virtual volume and associated volume/altitude tables

Virtual volume within a control volume is modified as nonhydrodynamic materials are relocated by their controlling packages. As a consequence, the hydrodynamic volume is also modified as the space that was occupied by nonhydrodynamic materials becomes available, and the space it now occupies is denied to the hydrodynamic materials. The other packages may track the location of their materials in more (or less) detail than is permitted by the set of altitudes recognized by CVH; this has no effect on hydrodynamic calculations.

2.2 Control Volume Contents

The contents of each volume are divided into a so-called *pool* and an *atmosphere*. These terms reflect a static, gravitationally separated situation, such as would exist in containment or in a primary system in the absence of strong forced circulation by pumps, and we conventionally depict the pool as occupying the lower portion of the control volume while the atmosphere fills the remainder. However, as discussed later, this picture is not interpreted so narrowly that it invalidates the use of MELCOR hydrodynamics in other situations.

The pool can be single-phase liquid water or, in nonequilibrium volumes as discussed below, two-phase (bubbly) water. No noncondensible gases are resident in the pool, although they may flow through and interact with it during a timestep. The atmosphere contains water vapor and/or noncondensible gases and may also include suspended water droplets, referred to as *fog*. The total volume is divided among pool, gaseous atmosphere, and fog, as shown in Figure 2.3.

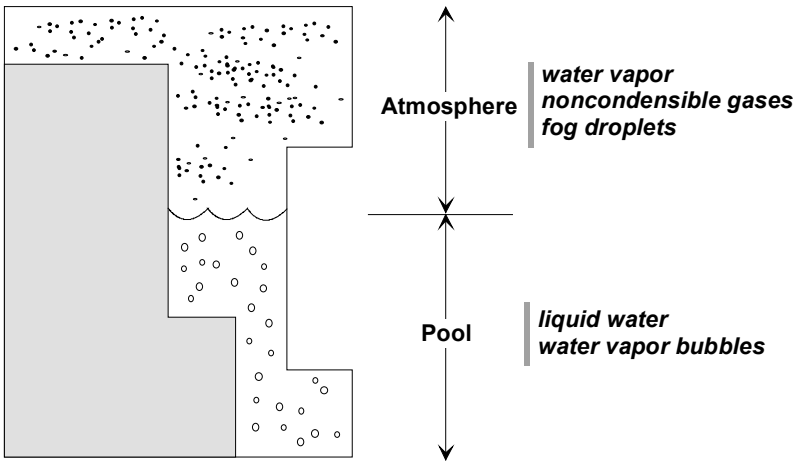


Figure 2.3 Control volume contents and pool surface

When needed by submodels, the pool surface is assumed to be a horizontal plane. Its elevation is defined from the volume of the pool by interpolation in the volume/altitude table for the control volume. Only the average void fraction in the pool is part of the CVH database, although a variation of void fraction with elevation may be assumed in submodels.

Materials are numbered in MELCOR. Materials 1, 2, and 3 are always pool, fog, and atmospheric water vapor, respectively. In particular, material 1 includes all of the pool, both liquid water and vapor bubbles. Materials with numbers greater than 3 are noncondensible gases. They are present in a calculation only if specified by the user, in which case their identities depend on input to the NCG package.

2.3 Control Volume Thermodynamic Properties

Given the volume and the mass and energy contents of a control volume, all of its thermodynamic properties are defined by an equation of state. There are two basic options available, selected by user input on record CV_THR: equilibrium and nonequilibrium.

In MELCOR, equilibrium thermodynamics assumes that the pool and the atmosphere are in thermal and mechanical equilibrium (i.e., that they have the same temperature and pressure). The two subvolumes, pool and atmosphere, are also assumed to be in equilibrium with respect to condensation/evaporation of water.

Nonequilibrium thermodynamics, on the other hand, assumes that while each subvolume is in internal equilibrium, it is in only mechanical equilibrium with the other. That is, neither thermal nor phase equilibrium is assumed between the pool and the atmosphere. (Note that this is *not* nonequilibrium in the sense of TRAC [6] or RELAP5 [7].) While the pressures of the pool and the atmosphere are equal, their temperatures may be different, and there may be a substantial driving force for condensation or

evaporation. The distinction between equilibrium and nonequilibrium thermodynamics exists only if a control volume contains both a pool and an atmosphere. The calculations required to determine the necessary thermodynamic properties (e.g., pressure, temperature) in either case are performed in the CVT; for a detailed description, see the CVT reference manual.

For equilibrium thermodynamics, only the total energy content of a control volume is relevant because CVT reapportions the total energy so as to obtain equilibrium among species in the atmosphere and between the atmosphere and the pool. This implies effectively instantaneous mass and energy transfer between pool and atmosphere, and the explicit calculation of the exchange terms is eliminated in favor of simple assumptions. All water vapor is currently assumed to be in the atmosphere. Liquid water, however, can exist both in the pool and as fog in the atmosphere. An auxiliary calculation is used to determine the partition. For more details, see Section 2.4 of the CVT reference manual.

The exchange terms must be calculated, however, for volumes in which nonequilibrium thermodynamics is prescribed. An additional term, the PdV work done by the pool on the atmosphere (or vice versa) as a result of motion of the pool surface, must also be kept in mind in the nonequilibrium case; it is actually accounted for (as $P \Delta V$) in CVT.

When nonhydrodynamic materials are relocated, changing the volume available to hydrodynamic materials, work is done in the process. This work is currently ignored in the package responsible for the relocation; that is, the energy inventory of that package is not affected. The error involved is insignificant in most cases because nonhydrodynamic materials are not ordinarily relocated through large pressure differentials, and the net work done is, therefore, very small. Pressure differentials can be large during high-pressure-melt ejection in the Fuel Dispersal Interactions (FDI) package, but even there the work term is small compared to other energy exchanges. However, the work must be included in CVH; for purposes of global energy accounting, it is treated as being created there.

The single pressure that CVH assigns to a control volume is assumed to correspond to the elevation of the pool/atmosphere interface. If there is no pool, this is taken as the bottom of the control volume; if there is no atmosphere, it is taken as the top. This choice (as opposed to a volume-centered pressure) simplifies the treatment of condensation/ evaporation rates at the interface. As discussed below, the hydrostatic head corresponding to the difference between the pool-surface reference elevation and the junction of a flow path to a control volume is accounted for in the momentum equation—such a head term would be necessary for *any* definition of the reference elevation for the pressure in a control volume.

3. Basic Flow Path Concepts

The basic concepts, definitions, and models associated with flow paths are described in this section. Discussion of most of the details is deferred until after the conservation equations have been presented and their solutions discussed.

3.1 Flow Path Definition

Each flow path connects two control volumes, specified on input record FL_FT. One is referred to as the *from* volume and the other as the *to* volume, thus defining the direction of positive flow. An arbitrary number of flow paths may be connected to or from each control volume; parallel paths (connecting the same two volumes) are allowed.

Mass and energy are advected through the flow paths, from one volume to another, in response to solutions of the momentum (flow) equation. No volume, mass, or energy is associated with a flow path itself, and no heat structures are allowed to communicate directly with the material passing through it. Therefore, the effect of advection through a flow path is to remove mass and energy from one control volume and to deposit it directly into another control volume. The formulation is manifestly conservative with respect to both mass and energy because there is a detailed balance between gains and losses in the two volumes connected by each flow path.

The cross-sectional area of a flow path is shared by pool and atmosphere in accordance with a calculated void fraction based on geometry and flow directions. The velocities of pool and atmosphere may be different if both are permitted to flow by the void fraction model; the directions of flow may even be opposite (i.e., countercurrent).

3.2 Flow Path Geometry

Flow path geometry is described on input records FL_FT and FL_GEO. Each flow path is characterized by a nominal area and a length. The area may be further modified by a user-controlled *open fraction*, which models (among other things) the effects of valves. The area is used in the conversion of volumetric flows to linear velocities and is, therefore, involved in form-loss and critical-flow modeling. The length is used in the momentum equation to define the inertia of the flow; as in other codes of this type, the ratio of length to area is the relevant parameter. It should be noted that (unlike some other codes) this inertial length is *not* used in the calculation of frictional pressure drops resulting from wall friction; *segment* data are used instead. Each flow path may be described in terms of several segments (i.e., differing lengths, areas, hydraulic diameters, and surface roughnesses). The details are discussed in Section 5.4; here it is sufficient to note that in the calculation segment data are combined with the flow path form-loss coefficient (optionally defined on input record FL_USL for both forward and reverse flow) to form a single effective loss coefficient applied to the flow-path velocity.

Each connection of a flow path to a control volume is referred to as a *junction* and is characterized by a nominal elevation and an opening height. The opening height defines a range of elevations about the junction elevation over which the flow path sees the contents of the control volume. These two quantities, in conjunction with the elevation of the pool surface, therefore, determine whether pool, atmosphere, or both are available for outflow. The junction elevations and heights are also used in the calculation of hydrostatic head terms; the lengths of the flow paths are not.

A flow path may be defined through user input on record FL_JSW to be *horizontal* or *vertical*. In a control volume/flow path formulation, the orientation of a flow path cannot be rigorously defined. The specification affects the definition of junction geometry, as shown below in Figure 3.1, and the (default) definition of the length over which interphase forces act, as described in Section 5.5.

The definition of a junction opening is illustrated in Figure 3.1, which also illustrates the possible truncation of the opening to match the associated control volume.

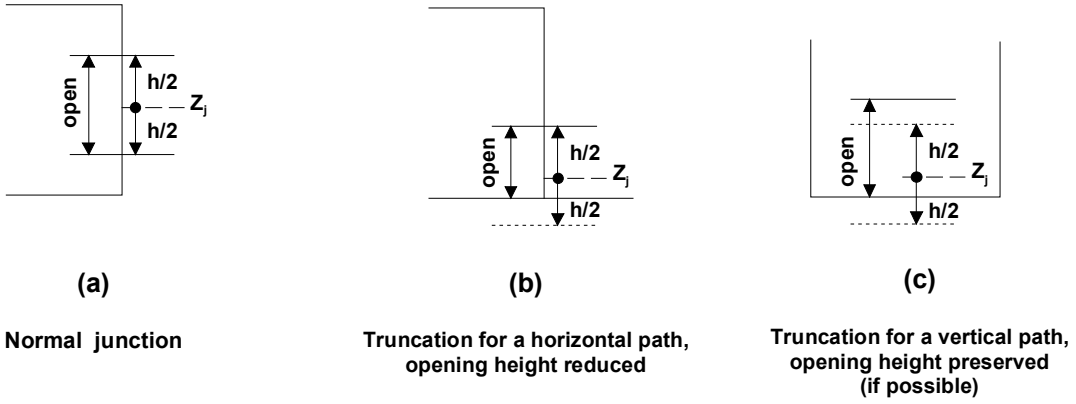


Figure 3.1 Junction geometry

Each junction elevation is required to lie within the range of elevations associated with the control volume with which it connects; that is, the junction elevation Z_J is required to lie between the bottom, Z_B, and the top, Z_T, of the control volume (inclusive). The junction height, h, is normally considered to be centered on the junction elevation, one half below and the remainder above, and, if the resulting junction opening (between $Z_J - h/2$ and $Z_J + h/2$) extends beyond the limits of the volume, it is truncated. (The nominal junction elevation, Z_J, is not modified.) In the case of a flow path specified by input as vertical (and in this case only), an attempt is made to preserve the full junction height. If the bottom of the junction opening is truncated, its top is raised a corresponding amount above $Z_J + h/2$ (but not above Z_T). A similar modification is applied if the top of the opening extends above the top of the volume. Input directives allow direct specification of the elevations of the top and bottom of junction openings. In this case, no adjustments are made, and the input is rejected if the opening extends beyond the limits of the associated volume. As currently implemented, the default definition of junction opening

heights and the treatment of the interphase force are the only differences in treatment between horizontal and vertical flow paths. (Details of the interphase force model are presented in Section 5.5.)

The junction void fraction is determined from the relative positions of the junction opening and the pool surface and is taken as the fraction of the opening height occupied by atmosphere (in effect, the opening is treated as rectangular). This is illustrated in Figure 3.2. (*Atmosphere fraction* would be a more precise term than *void fraction* because fog flows with the gaseous component of the atmosphere and bubbles flow with the pool.)

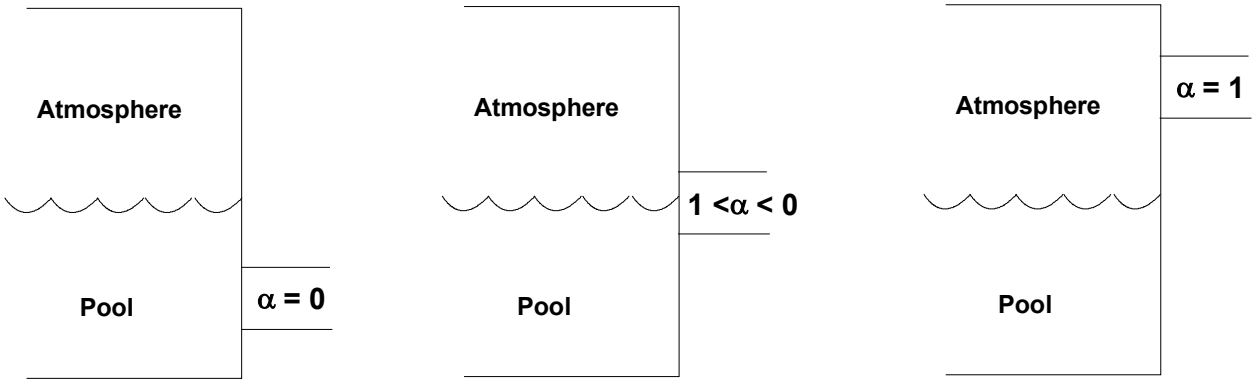


Figure 3.2 Relationship among junction opening, pool S=surface elevation, and void fraction

In the tank-and-pipe limit of hydrodynamic modeling, the length, junction elevation, and height have relatively clear physical interpretations. It is recommended that the junction height for connection of a *vertical* pipe to a tank should be taken as something like the pipe radius; this models to some extent the two-dimensional distortion of the pool surface when there is flow through such a connection, as well as eliminating the discontinuity in behavior that would otherwise occur when the pool surface crosses through the junction elevation. Because of this role in eliminating discontinuous behavior, the junction height may not be input as zero.

In the finite-difference limit, a flow path represents a surface that is a common boundary between the volumes connected; the length should be taken as roughly the center-to-center distance between volumes, and the elevations of both ends of the junction should be taken as the midpoint elevation of the common boundary. For horizontal flow through a vertical boundary, the junction height should be specified as large enough to include the entire boundary. For vertical flow through a horizontal boundary, the height has no rigorous interpretation; it serves only to define the range of elevations from which material may be drawn.

The flow equations include a term for the interphase force acting between the pool flow and the atmosphere flow in a single flow path. Among other things, this force tends to

limit the relative velocity between the phases and can cause entrainment through a vertical flow path whenever both phases (pool and atmosphere) are present within the junction opening and the interphase force is large enough to overcome the head difference for them. In particular, a flow of atmosphere from a lower volume to a higher one can entrain an upward pool flow (and a downward pool flow can entrain a corresponding downward atmosphere flow) despite an opposing difference in pressure plus head, if the associated junction opening is sufficiently large that both pool and atmosphere are present within the opening height. This tends to "smear" the pool surface slightly for the purposes of flow calculations and reduces the computational effort in cases where a rising (or falling) pool surface passes through the top (or bottom) of a control volume. We have found that use of an opening height that is a substantial fraction of the volume height frequently works well.

It is also possible to modify the finite difference limit by dividing the common boundary between two control volumes into two or more parallel flow paths with different elevations, whose areas sum to the correct geometrical total, as illustrated in Figure 3.3. There is preliminary evidence that some aspects of natural convection may be calculable this way.

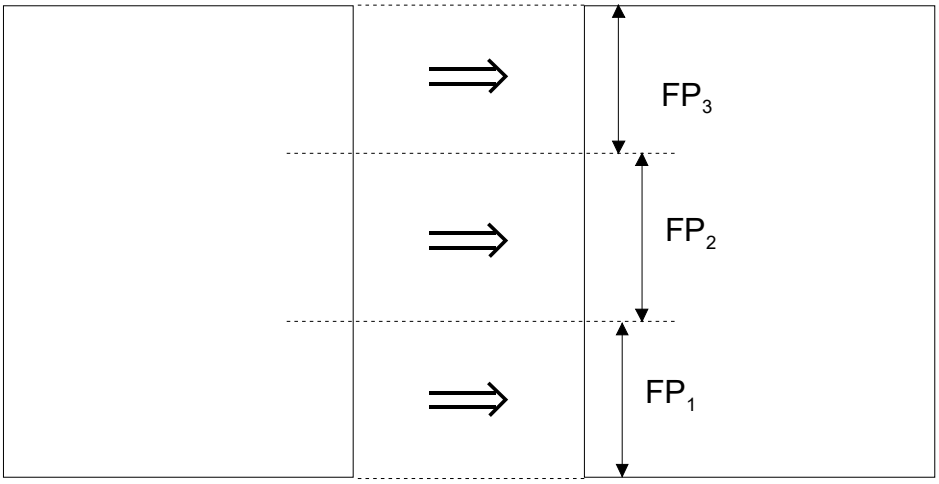


Figure 3.3 Multiple flow paths connecting two volumes to model natural circulation

4. Governing Equations

The governing equations for thermal-hydraulic behavior in MELCOR are the equations of conservation of mass, momentum, and energy. These equations are presented first as ordinary differential equations for the control volume formulation and then in the linearized-implicit finite difference form that is solved. They could, of course, be derived by suitable integration of the three-dimensional partial differential equations over a volume (for the scalar mass and energy equations) or along a line (for the vector momentum equation), but the insights to be gained do not justify including the derivation in this reference manual. See, for example, Reference [1].

4.1 Ordinary Differential Equations

The differential equation expressing conservation of mass for each material is

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = \Gamma \tag{4-1}$$

where Γ is the volumetric mass source density. Integrated over a control volume, the conservation of mass for material m in control volume i is then expressed by

$$\frac{\partial M_{i,m}}{\partial t} = \sum_j \sigma_{ij} \alpha_{j,\phi} \rho^d_{j,m} v_{j,\phi} F_j A_j + \dot{M}_{i,m} \tag{4-2}$$

Here, *M* is total mass; subscript *j* refers to flow path, with σ_{ij} accounting for the direction of flow in flow path *j* with respect to volume *i* as described below; subscript ϕ refers to the phase, pool or atmosphere (later abbreviated as "*P*" and "*A,*" respectively), in which material m resides; $\alpha_{j,\phi}$ is the volume fraction of ϕ in flow path-*j* ($\alpha_{j,A} + \alpha_{j,P} = 1$), see Section 5.2 for definitions); t is time, ρ is density; superscript "*d*" denotes "donor," corresponding to the control volume from which material is flowing; *v* is flow velocity; *A* is flow-path area; F is the fraction of this area that is open; and *M* includes all non-flow sources, such as condensation/evaporation, bubble separation, fog precipitation, user-defined sources in CVH, and contributions from other packages in MELCOR.

The summation in Equation (4-2) is over all flow paths, with

$$\sigma_{ij} = \begin{cases} +1 & \textit{if path } j \textit{ is connected "to" volume } i \\ -1 & \textit{if path } j \textit{ is connected "from" volume } i \\ 0 & \textit{if path } j \textit{ is not connected to volume } i \end{cases} \tag{4-3}$$

accounting for which flow paths are connected to volume *i*, and for the direction of positive flow in these paths. As used here, the density is defined by

$$\rho_m \equiv \frac{M_m}{V_\phi} \tag{4-4}$$

where V_ϕ is the volume of the phase containing material m. Recall that the pool phase contains single- or two-phase water while the atmosphere can contain water vapor, noncondensible gases, and liquid water fog.

The equations expressing conservation of energy in the pool and in the atmosphere are derived similarly from the partial differential equations, neglecting all gravitational

potential energy and volume-average kinetic energy terms. Conservation of energy in phase φ (pool or atmosphere) is then expressed by

$$\frac{\partial E_{i,\phi}}{\partial t} = \sum_j \sigma_{ij}\alpha_{j,\phi}\left(\sum_m \rho^d_{j,m}h^d_{j,m}\right)v_{j,\phi}F_jA_j + \dot{H}_{i,\phi} \qquad (4\text{-}5)$$

where *E* is total internal energy; *m* in the second summation runs over all materials in phase ϕ ; *h* is the specific enthalpy (the difference between *h* and the specific internal energy, *e*, accounts for flow work); and *H* is the nonflow energy source, including the enthalpy of all relevant mass sources in Equation (4-2).

Finally, the equations for pool flow and for atmosphere flow in a flow path are obtained from line integrals of the acceleration equations along a streamline from the center of the *from* volume to the center of the *to* volume. The temporal rate of change of the void fraction, $\partial\alpha / \partial t$, is neglected. The results (in nonconservative form) are expressed by

$$\begin{aligned}\alpha_{j,\varphi}\rho_{j,\varphi}L_j\frac{\partial v_{j,\varphi}}{\partial t} = {} & \alpha_{j,\varphi}\left(P_i - P_k\right) + \alpha_{j,\varphi}\left(\rho\, g\Delta z\right)_{j,\varphi} + \alpha_{j,\varphi}\Delta P_j \\ & - \frac{1}{2}K^*_{j,\varphi}\alpha_{j,\varphi}\rho_{j,\varphi}\left|v_{j,\varphi}\right|v_{j,\varphi} \\ & - \alpha_{j,\varphi}\alpha_{j,-\varphi}f_{2,j}L_{2,j}\left(v_{j,\varphi} - v_{j,-\varphi}\right) \\ & + \alpha_{j,\varphi}\rho_{j,\varphi}v_{j,\varphi}\left(\Delta v\right)_{j,\varphi}\end{aligned} \qquad (4\text{-}6)$$

where L_j is the inertial length of the flow path; *i* and *k* are the *from* and *to* control volumes, respectively, for flow path *j*; *g* is the acceleration of gravity; ΔP_j represents any pump head developed in the flow path; K^* is the net form- and wall-loss coefficient; $f_{2,j}$ is the interphase force (momentum exchange) coefficient; $L_{2,j}$ is the effective length over which the interphase force acts (not necessarily equal to the inertial length, see Section 5.5); $\left(\Delta v\right)_{j,\varphi}$ represents the change in velocity through the flow path (the "momentum flux"); and $-\varphi$ denotes the "other" phase relative to φ (atmosphere if φ is pool and vice versa).

Unless a phase is present within at least one of the junction openings associated with a flow path, flow of that phase through that path is impossible and the corresponding flow equation [Equation (4-6)] need not be solved; $v_{j,\varphi}$ is simply set to zero.

The density of a phase in a flow path is ordinarily taken as the density in the donor volume; the phase densities are evaluated from Equation (4-4), with a summation over the materials in the atmosphere. In general, the set of flow equations must be solved iteratively (see Section 4.2) with donor redefined, if necessary, for each iteration. If a

phase is present within only one of the junction openings so that flow of that phase within that flow path is only possible in one direction, the donor density is taken as that in the only possible donor volume.

The redefinition of a flow path density between iterations as a result of reversal of the associated flow introduces a discontinuity in the equations. We have observed that this can prevent convergence of the solution under some conditions. Therefore, the next-iterate flow path density is taken as

$$\rho_{j,\phi}^{(i)} = f\rho_{j,\phi}^{(i-1)} + (1-f)\rho_{j,\phi}^{d}. \qquad (4\text{-}7)$$

For the first third of the permitted total number of iterations, *f* is taken as zero, resulting in use of a pure donor density. If further iterations are required, *f* is increased linearly from zero to one for the next third of the permitted total number, introducing an increasing degree of averaging into the definition of density. Finally, *f* is taken as one for the last third of the iterations (if required), eliminating the numerical discontinuity.

The gravitational head term and the loss term are each somewhat complicated and are discussed in detail in Sections 5.3 and 5.4, respectively. The accounting for interphase forces represented by $f_{2,j}$ is described in Section 5.5, and the models available for the pump head are presented in Section 5.6. Note that, as written, the volume fraction, α_ϕ, cancels identically in the equation.

The last term in Equation (4-6), $v_{j,\varphi}(\Delta v)_{j,\varphi}$, represents the advection of momentum through the flow path and arises from integration of the term $v(\partial v/\partial x)$ in the continuum equations. The formulation presented here is essentially one-dimensional; in more general geometry, *v* in Equation (4-6) may be interpreted as the velocity component in the direction of flow (denoted by "*x*"); however, the treatment is incomplete because the cross terms arising from $v_v(\partial v_x/\partial y)$ are not included in the equations.

By default, even the diagonal momentum flux term in Equation (4-6) is neglected in the solution of the hydrodynamic equations in MELCOR. This is consistent with omission of the kinetic energy in Equation (4-5). These terms (momentum flux and kinetic energy) have traditionally been sources of difficulty in control volume codes because they involve a volume-centered velocity, which requires a multi-dimensional formulation for proper definition. (Note that codes such as RELAP5 [7] make very specific geometric assumptions concerning the relationship between control volumes and flow paths.) The neglected terms in both equations are of order Ma^2, where Ma is the Mach number based on volume-centered velocities and are ordinarily small (although they may be important for flow boiling with large density gradients). Velocities in flow paths may be sonic or near sonic, but constancy of $h + 1/2v^2$ for adiabatic (not necessarily isentropic) flow ensures that only volume-centered velocities appear in the equations. Choking is treated as an imposed limit on flows based on correlations (see Section 6.3). In any

case, consistent inclusion of the v^2 terms would require a proper definition of a volume-centered velocity, including multidimensional effects, and it is clear that this can be done in anything but a full finite difference code (see Section 6.5). In most cases, no difficulties arise if MELCOR pressures and enthalpies are stagnation pressures and stagnation enthalpies.

4.2 Finite Difference Equations

The ordinary differential equations presented in Section 4.1 are converted to linearized-implicit finite difference equations for solution in MELCOR.

For each timestep, Δt , the new (end-of-step) velocities are used in the advection (flow) terms in the mass and energy equations to write

$$M_{i,m}^{n} = M_{i,m}^{o} + \sum_{j} \sigma_{ij} \alpha_{j,\phi}^{n} \rho_{j,m}^{d} v_{j,\phi}^{n} F_j A_j \Delta t + \delta M_{i,m} \tag{4-8}$$

$$E_{i,\phi}^{n} = E_{i,\phi}^{o} + \sum_{j} \sigma_{ij} \alpha_{j,\phi}^{n} \left(\sum_{m} \rho_{j,m}^{d} h_{j,m}^{d} \right) v_{j,\phi}^{n} F_j A_j \Delta t + \delta H_{i,\phi} \tag{4-9}$$

where superscripts *n* and *o* refer to the new and old time levels, respectively; and δM and δH are the net external sources (integrals from t^o to $t^o + \Delta t$).

The time levels on the donor properties are not explicitly shown in Equations (4-8) and (4-9); they are essentially old values (at t^o). See Section 4.4 for further discussion.

It is clear that this formulation is conservative with respect to both masses and internal energies because every term representing a flow transfer *from* a volume is exactly balanced by a transfer *to* the volume at the other end of the flow path. Therefore, masses and energies are conserved to within the accumulation of roundoff on the computer used.

In the interest of numerical stability, linearized-implicit ("semi-implicit") differencing is used in several terms in the momentum equation [Equation (4-6)]. Specifically, the equation is differenced using projected end-of-step pressures and heads in the acceleration terms and end-of-step velocities in the frictional loss and momentum exchange terms. Because of the nonlinearity of the frictional loss term, the resulting finite difference equation must be solved iteratively. (Because of nonlinearity of the equation of state used to project the end-of-step pressures, a further iteration may be required. We return to this in Section 4.3) We first discuss the treatment of velocities, then define and discuss the other terms in the finite difference equation.

At each velocity iteration, the form- and wall-loss term is linearized about the best available estimate of v^n, denoted by v^{n-} (this is initially v^o), to obtain the finite difference equation for the estimated new, end-of-step velocity:

$$v_{j,\varphi}^{n} = v_{j,\varphi}^{o+} + \frac{\Delta t}{\rho_{j,\varphi} L_j}\left(P_i^{\tilde{n}} + \Delta P_j - P_k^{\tilde{n}} + \left(\rho\, g\Delta z\right)_{j,\varphi}^{\tilde{n}} + v_{j,\varphi}^{o}\left(\rho\,\Delta v\right)_{j,\varphi}^{o}\right)$$
$$- \frac{K_{j,\varphi}^{*}\Delta t}{2L_j}\left(v_{j,\varphi}^{n-} + v'_{j,\varphi}\left|v_{j,\varphi}^{n} - \left|v'_{j,\varphi}\right|v_{j,\varphi}^{n-}\right.\right) \qquad (4\text{-}10)$$
$$- \frac{\alpha_{j,-\varphi} f_{2,j} L_{2,j}\Delta t}{\rho_{j,\varphi} L_j}\left(v_{j,\varphi}^{n} - v_{j,-\varphi}^{n}\right)$$

The nature of the linearization in velocity is determined by the choice of v'. For the first iteration, v' is taken as v^o, giving a tangent (Taylor series) linearization. For later iterations, it is taken as v^{n-} from the previous iteration if the velocity did not reverse during that iteration or as zero otherwise. The result is to approximate v^2 by the secant from the latest iterate through the next oldest iterate or by the secant through the origin, respectively. Note that the interphase force term is fully implicit with respect to velocities and that the length over which this force acts, $L_{2,j}$, may differ from the inertial length of the flow path, L_j. See discussion for definition of $P^{\tilde{n}}$.

The superscript "$o+$" on the velocity on the right-hand side of Equation (4-10) indicates that it has been modified from the old value to account for changes in the flow path void fraction, as discussed below. This was found necessary to prevent initiation of a nonphysical transient whenever the motion of a pool surface through a small junction opening produced a major change in void fraction during a single timestep.

The problem is that the old velocities, v^o, were computed with the old void fraction, α^o; with α^n, they may correspond to a quite different flow state both in mass flow and in total volumetric flow. This may require large accelerations (and pressure differentials) to maintain the "correct" flow. The cause is, in part, that the time derivative of the void fraction does not appear in the momentum equation. (There are no further problems involving the time level of data on which α is based and the fact that its treatment is not numerically implicit.)

The definition of void fraction in MELCOR is necessarily much more complicated than in a simple fine-zoned finite difference code, and an attempt to include $\partial\alpha/\partial t$ in the momentum equation seemed unlikely to be productive. Therefore, we have chosen to employ an *ad hoc* modification of the old velocities to account for changes in void fractions. (Sensitivity coefficient 4408 may be used to disable this modification.) The criteria used are preservation of the total volumetric flux, expressed by

$$\alpha_{j,A}^{n} v_{j,A}^{o+} + \alpha_{j,P}^{n} v_{j,P}^{o+} = \alpha_{j,A}^{o} v_{j,A}^{o} + \alpha_{j,P}^{o} v_{j,P}^{o} \tag{4-11}$$

and preservation of the relative velocity between the phases, expressed by

$$v_{j,A}^{o+} - v_{j,P}^{o+} = v_{j,A}^{o} - v_{j,P}^{o} \tag{4-12}$$

This results in

$$v_{j,\phi}^{o} = v_{j,\phi}^{o} + \left(\alpha_{j,A}^{o} - \alpha_{j,A}^{n}\right)\left(v_{j,A}^{o} - v_{j,P}^{o}\right) \tag{4-13}$$

It is interesting to note that there is an analogous relationship implicit in drift-flux codes. In such codes, the total mass flux (momentum density) is determined by a single momentum equation for each flow path, and a constitutive relation (the drift flux correlation) is then used to partition this flux into liquid and vapor components as a function of void fraction. Thus, when a new void fraction is computed at the start of a timestep, the total mass flux is preserved but the individual phase velocities and the total volumetric flux are not. MELCOR calculations more often involve quasi-steady flows than pressure waves; therefore, preservation of the volumetric flow rather than the momentum density (mass flux) was chosen as the default treatment. (Note that there is no way that both the mass fluxes and volumetric flows could be preserved as the void fraction changes.)

As noted previously, the momentum flux term, $v(\rho\,\Delta v)$ in Equation (4-10), are omitted by default. We have found no need for implicit treatment of this term if it is included; therefore, start-of-step velocities are used in its evaluation. If the term is to be included in the momentum equation for flow path *j*, the user is required to specify on input record FL_MFX the flow paths that are logically upstream and downstream from flow path *j*, as described in the FL Users' Guide. The specification of "no such flow path" is permitted to allow treatment of a flow path connected to a dead-end volume or to one with no other appropriately oriented connection.

The term $(\rho\,\Delta v)$, representing a spatial difference in momentum density, is treated as a donored quantity. It is evaluated based on the direction of flow through flow path *j*, as

$$(\rho\ \Delta v)_{j,\phi}^{o} = \begin{cases} \rho_i \left(\dfrac{F_{j-} A_{j-} v_{j-,\phi}^{o}}{A_i} - \dfrac{F_j A_j v_{j,\phi}^{o}}{A_k} \right) & v_{j,\phi}^{o} \geq 0 \\ \rho_k \left(\dfrac{F_j A_j v_{j,\phi}^{o}}{A_i} - \dfrac{F_{j+} A_{j+} v_{j+,\phi}^{o}}{A_k} \right) & v_{j,\phi}^{o} < 0 \end{cases} \tag{4-14}$$

Here, subscripts *i* and *k* denote the donor and acceptor volumes, respectively; A_i and A_k are the corresponding user-defined flow areas for these volumes *in the direction of flow*

appropriate to flow path j; and subscripts *j*- and *j*+ refer to the designated flow paths that are logically upstream and downstream of *j* and must connect to volumes *i* and *k*, respectively.

The area ratios in Equation (4-14) serve to convert the momentum density in each flow path to corresponding densities at the volume center, under the assumption of incompressible flow. The volume areas, which may differ from those used in the control volume velocity calculation, must be specified by the user on record FL_MFX. This allows a more accurate description of the actual flow geometry. For example, most of the momentum of a small jet entering a large room is dissipated close to the point of entry, leaving little momentum to be advected through a second flow path, and in general, this effect is captured through the ratio of the small flow path area to the large volume area. However, if the two flow paths are closely aligned so that a fluid jet from one is captured by the other, the user may capture the effect by specifying a volume flow area appropriate for the jet.

If either flow path *j*- or *j*+ is absent (as defined by user input), the corresponding term in Equation (4-14) is neglected, which is equivalent to setting the associated flow path area to zero.

As noted previously, the pressures, $P^{\tilde{n}}$, used in the acceleration terms in Equation (4-10) are *predicted* end-of-step pressures; they are calculated from the linearization of the equation of state about a reference point (denoted by "*") as

$$\begin{aligned} P_i^{\tilde{n}} = P_i^* + \sum_m \frac{\partial P_i^*}{\partial M_{i,m}} \left(M^n_{i,m} - M^*_{i,m} \right) \\ + \frac{\partial P_i^*}{\partial E_{i,P}} \left(E^n_{i,P} - E^*_{i,P} \right) + \frac{\partial P_i^*}{\partial E_{i,A}} \left(E^n_{i,A} - E^*_{i,A} \right) \end{aligned} \tag{4-15}$$

The choice of the linearization point is discussed in detail in Section 4.3.

The static head terms, $(\rho\, g\Delta z)^{\tilde{n}}$, are also predicted values at end-of-step. However, only changes in pool mass and hydrodynamic volume are included in the projection with changes in atmosphere mass and phase densities neglected. Specifically,

$$\begin{aligned} (\rho\, g\Delta z)^{\tilde{n}}_{j,\varphi} = (\rho\, g\Delta z)^{o}_{j,\varphi} + \frac{\partial (\rho\, g\Delta z)_{j,\varphi}}{\partial M_{i,P}} \left(M^n_{i,P} - M^{o+}_{i,P} \right) \\ + \frac{\partial (\rho\, g\Delta z)_{j,\varphi}}{\partial M_{k,P}} \left(M^n_{k,P} - M^{o+}_{k,P} \right) \end{aligned} \tag{4-16}$$

In this equation $M_{i,P}^{o+}$ is the mass of pool that can be accommodated below the former elevation of the pool surface at the old pool density. It differs from $M_{i,p}^{o}$, only if there has been a change in the volume/altitude table resulting from a change of virtual volume in control volume *i*. In this case, the difference accounts for the change in pool surface elevation—and, therefore, in static head—in the absence of a change in pool mass.

The new masses and new energies in Equations (4-15) and (4-16) are given by Equations (4-8) and (4-9), respectively. The derivatives $\partial P / \partial M$ and $\partial P / \partial E$ are calculated by the CVT package and represent the linearized effect of changing mass and energy contents of the control volumes. See the CVT Reference Manual for further details. The derivatives $\partial(\rho\, g\Delta z)/\partial M$ reflect the linearized effect of changing pool mass on the flow-path head terms; they are defined in Section 5.3.

When all terms associated with each flow are collected together for a given volume, the projected new pressure in Equation (4-15) has the form

$$P_i^{\tilde{n}} = \hat{P}_i + \sum_{s,\Psi} \frac{\partial P_i}{\partial V_{s,\psi}} \sigma_{is} \alpha_{s,\psi} F_s A_s \, v_{s,\psi}^{n} \Delta t \tag{4-17}$$

where

$$\begin{aligned} \hat{P}_i = P_i^* &+ \sum_m \frac{\partial P_i^*}{\partial M_{i,m}} \left(\hat{M}_{i,m} - M_{i,m}^* \right) \\ &+ \frac{\partial P_i^*}{\partial E_{i,P}} \left(\hat{E}_{i,P} - E_{i,P}^* \right) + \frac{\partial P_i^*}{\partial E_{i,A}} \left(\hat{E}_{i,A} - E_{i,A}^* \right) \end{aligned} \tag{4-18}$$

$$\hat{M}_{i,m} = M_{i,m}^{o} + \delta M_{i,m} \tag{4-19}$$

$$\hat{E}_{i,\phi} = E_{i,\phi}^{o} + \delta H_{i,\phi} \tag{4-20}$$

and

$$\frac{\partial P_i}{\partial V_S} = \sum_m \frac{\partial P_i^*}{\partial M_{i,m}} \rho_{S,m}^{d} + \frac{\partial P_i^*}{\partial E_{i,P}} (\rho\ h)_{S,P}^{d} + \frac{\partial P_i^*}{\partial E_{i,A}} (\rho\ h)_{S,A}^{d} \tag{4-21}$$

Here “S” is used as an abbreviation for “s,ψ”, *and*

$$(\rho\ h)_{S,\phi}^{d} \equiv \sum_{m\ in \phi} \rho_{S,m}^{d} h_{S,m}^{d} \tag{4-22}$$

Because donor densities are used in the advection terms, they appear in the definition of $\partial P / \partial V$ in Equations (4-21) and (4-22). Therefore, $\partial P / \partial V$ depends on the direction of flow. In general, if s,ψ represents a pool (atmosphere) flow, only the pool (atmosphere) energy and materials are associated with nonzero densities in the evaluation of $\partial P_i / \partial V_{s,\psi}$. However, the code is written with the greater generality of allowing atmosphere (pool) materials to be associated with pool (atmosphere) flows, and different donor density arrays are used to describe flows entering and leaving a flow path. This allows some interactions to be treated as occurring within a flow path. This capability is currently used in conjunction with the SPARC model, as described in Section 6.1.

Substitution of the predicted pressures and heads into the velocity equation leads to a set of linear equations to be solved for the new velocities:

$$\left(1+\frac{K^*_{j,\varphi}\Delta t}{2L_j}\left|v^{n-}_{j,\varphi}+v'_{j,\varphi}\right|+\frac{\alpha_{j,-\varphi}f_{2,j}L_{2,j}\Delta t}{\rho_{j,\varphi}L_j}\right)v^{n}_{j,\varphi} \quad - \quad \frac{\alpha_{j,-\varphi}f_{2,j}L_{2,j}\Delta t}{\rho_{j,\varphi}L_j}v$$
$$+\sum_{s,\psi}C(j,\varphi:s,\psi)v^{n}_{s,\psi}$$

$$= v^{o+}_{j,\varphi}+\frac{K^*_{j,\varphi}\Delta t}{2L_j}\left|v'_{j,\varphi}\right|v^{n-}_{j,\varphi}+\frac{\Delta t}{\rho_{j,\varphi}L_j}\left(\hat{P}_i+\Delta P_j-\hat{P}_k\right)$$
$$+\left(\rho\, g\Delta z\right)^{o}_{j,\varphi}+\frac{\partial\left(\rho\, g\Delta z\right)_{j,\varphi}}{\partial M_{i,P}}\left(\hat{M}^{o}_{i,P}-M^{o+}_{i,P}\right) \qquad (4\text{-}23)$$
$$+\frac{\partial\left(\rho\, g\Delta z\right)_{j,\varphi}}{\partial M_{k,P}}\left(\hat{M}^{o}_{k,P}-M^{o+}_{k,P}\right)\Bigg]$$

The summation on the left-hand side is over both phases, ψ, in all flow paths, *s*, although only those paths that connect either to volume *i* or to volume *k* contribute, as seen below. The coefficients in the sum are given by

$$C(j,\varphi:s,\psi)=\frac{(\Delta t)^2}{\rho_{j,\varphi}L_j}\alpha_{s,\psi}A_sF_s \left\{-\sigma_{is}\left[\frac{\partial P_i}{\partial V_{s,\psi}}+\delta_{\psi P}\rho^d_{s,P}\frac{\partial(\rho\, g\Delta z)_{j,\varphi}}{\partial M_{i,P}}\right] +\sigma_{ks}\left[\frac{\partial P_k}{\partial V_{s,\psi}}+\delta_{\psi P}\rho^d_{s,P}\frac{\partial(\rho\, g\Delta z)_{j,\varphi}}{\partial M_{k,P}}\right]\right\} \quad (4\text{-}24)$$

where

$$\delta_{ij}=\begin{cases}1 & i=j\\ 0 & i\neq j\end{cases} \quad (4\text{-}25)$$

by the Kronecker delta. Because of the appearance of σ_{is} and σ_{ks}, the coefficient given by Equation (4-24) is nonzero only for flow paths that connect to volume *i* or to volume *k*; because of the appearance of $\delta_{\psi P}$, the head term appears only in cases where s,ψ is a pool flow.

Equation (4-24) could be made somewhat more compact by obtaining the two sets of terms on the right (for volumes *i* and *k*) from a sum over *all* volumes with appropriate coefficients to pick out the desired terms with the correct signs and eliminate the contributions of all others. However, this would only further conceal the essential point that two flows are coupled by the matrix if and only if there is a volume to which both connect, allowing each flow to affect the pressure differential driving the other.

As mentioned previously, the nonlinearity of the loss (friction) terms and the possibility of flow reversals affecting donor quantities require that the solution of the set of linear Equation (4-23) be repeated until all the new velocities have converged. The control of this iteration is described in Section 4.3.

4.2.1 Inclusion of Bubble-Separation Terms within the Implicit Formulation

To this point, only the contribution of advection terms has been treated within a numerically implicit formulation. The effects of all sources were included in the δM and δH terms in Equations (4-8) and (4-9), which are then treated explicitly. These sources were considered to include several processes that could transfer mass and energy between the pool and atmosphere of a single volume within CHV: condensation/evaporation, bubble separation, and fog deposition. Experience has shown that inclusion of the effects of bubble separation as part of the explicit sources could lead to severe numerical instabilities, particularly in problems involving boiling pools at low pressures. One problem is that the resulting large oscillations in the calculated elevation of the pool surface resulting from large oscillations in the calculated

void (bubble) fraction in the pool can have a significant impact on heat transfer in the COR and HS packages. This was identified as a deficiency in the FLECHT SEASET assessment calculations [8].

The finite difference equations were modified in MELCOR 1.8.3 and later versions to include the transfer of vapor mass and energy from the pool to the atmosphere of a control volume within the implicit formulation. Because bubble separation is an intravolume process, its effects may be included along with those of the equation of state in defining a generalized form of Equation (4-15) in which bubble separation is included *implicitly* and then eliminated algebraically before proceeding with the solution. The effect is to define net derivatives that include the linearized effect of bubble separation.

The rates of separation of mass and energy by bubbles are primarily functions of the pool void fraction, α, and geometry, and the fact that the observed problems arise from instability in the calculated pool void fraction. We, therefore, linearize the bubble separation terms within volume i with respect to the pool void fraction in that volume as

$$\delta M_{B,i} = \delta M^*_{B,i} + \frac{\partial\left(\delta M^*_{B,i}\right)}{\partial \alpha_i}\left(\alpha_i^{\tilde{n}} - \alpha_i^*\right) \tag{4-26}$$

$$\delta H_{B,i} = \delta H^*_{B,i} + h_v \ \frac{\partial\left(\delta M^*_{B,i}\right)}{\partial \alpha_i}\left(\alpha_i^{\tilde{n}} - \alpha_i^*\right) \tag{4-27}$$

where δM^* and δH^* are evaluated using the pool void fraction at the linearization point, and α^*, and $\alpha^{\tilde{n}}$ is the projected end-of-step pool void fraction. (The details of the bubble separation model itself are presented in Section 5.1.3.)

The pool void fraction is a natural function of the specific enthalpy of the pool and the enthalpies of saturated liquid and vapor and may, therefore, be considered as a function of the total pool mass, the total pool energy, and the control volume pressure. In response to a variation in these quantities, the change in α is

$$d\alpha_i = \frac{\partial \alpha_i}{\partial M_{i,1}}\left(dM'_{i,1} - dM_{B,i}\right) + \frac{\partial \alpha_i}{\partial E_{i,P}}\left(dE'_{i,P} - h_v dM_{B,i}\right) + \frac{\partial \alpha_i}{\partial P_i} dP_i \tag{4-28}$$

where the primes denote changes in addition to bubble separation (i.e., other sources and advection). Using the same convention, the linearization of the volume pressure (from which Equation (4-15) was derived) becomes

$$dP_i = \sum_m \frac{\partial P_i^*}{\partial M_{i,m}} \left(dM'_{i,m} - \delta_{m1} dM_{B,i} + \delta_{m3} dM_{B,i} \right) + \frac{\partial P_i^*}{\partial E_{i,P}} \left(dE'_{i,P} - h_v dM_{B,i} \right) + \frac{\partial P_i^*}{\partial E_{i,A}} \left(dE'_{i,A} + h_v dM_{B,i} \right) \quad (4\text{-}29)$$

Equation (4-27) can be used to eliminate $dM_{B,i}$ from Equations (4-28) and (4-29), and the resulting equations are solved for ∂P_i and $\partial \alpha_i$ as linear functions of the variables $\partial M'_{i,m}$ and $\partial E'_{i,\varphi}$. The results take the form

$$dP_i = \sum_m \frac{\partial P_i^*}{\partial M'_{i,m}} d\, M'_{i,m} + \frac{\partial P_i^*}{\partial E'_{i,P}} dE'_{i,P} + \frac{\partial P_i^*}{\partial E'_{i,A}} dE'_{i,A} \quad (4\text{-}30)$$

$$d\alpha_i = \sum_m \frac{\partial \alpha_i^*}{\partial M'_{i,m}}\ dM'_{i,m} + \frac{\partial \alpha_i^*}{\partial E'_{i,P}}\ dE'_{i,P} + \frac{\partial \alpha_i^*}{\partial E'_{i,A}}\ dE'_{i,A} \quad (4\text{-}31)$$

Here the modified pressure derivatives are

$$\frac{\partial P_i^*}{\partial X'} = \frac{\frac{\partial P_i^*}{\partial X} - C_i \left(\frac{\partial \alpha_i^*}{\partial M_{B,i}} \frac{\partial P_i^*}{\partial X} - \frac{\partial P_i^*}{\partial M_{B,i}} \frac{\partial \alpha_i^*}{\partial X} \right) \Delta t}{1 - C_i \left(\frac{\partial \alpha_i^*}{\partial M_{B,i}} + \frac{\partial \alpha_i^*}{\partial P_i} - \frac{\partial P_i^*}{\partial M_{B,i}} \right) \Delta t} \quad (4\text{-}32)$$

where dX' represents any of the variables $dM'_{i,m}$ and $\partial E'_{i,\varphi}$ and

$$\frac{\partial \alpha_i^*}{\partial M_{B,i}} \equiv -\left(\frac{\partial \alpha_i^*}{\partial M_{i,1}} + h_v \frac{\partial \alpha_i^*}{\partial E_{i,P}} \right) \quad (4\text{-}33)$$

$$\frac{\partial P_i^*}{\partial M_{B,i}} \equiv \left(\frac{\partial P_i^*}{\partial M_{i,3}} + h_v \frac{\partial P_i^*}{\partial E_{i,A}} \right) - \left(\frac{\partial P_i^*}{\partial M_{i,1}} + h_v \frac{\partial P_i^*}{\partial E_{i,P}} \right) \quad (4\text{-}34)$$

are convenient combinations of the derivatives in Equations (4-28) and (4-29).

The momentum equation is constructed and solved as before but now using Equation (4-30) to project the new pressures. The only differences that result are that the derivatives $\partial P^* / \partial X'$ appear in Equations (4-18) and (4-21) rather than $\partial P^* / \partial X$ and that only δM^* and δH^* from Equations (4-26) and (4-27) are included in the source terms in Equations (4-27) and (4-20). During the solution, any change in bubble separation is implicitly included by virtue of the modified pressure derivatives.

Once the new velocities are determined, the contribution of advection to new mass and energy inventories [the sums over flow paths in Equations (4-8) and (4-9)] is determined as before. The additional mass and energy transfers resulting from the implicit change in bubble separation in Equations (4-26) and (4-27) must also be included—in addition to δM^* and δH^*—in defining the new mass and energy inventories in Equations (4-8) and (4-9). Once the contribution of advection has been determined, the contribution of implicit bubble separation is evaluated from

$$\begin{aligned}\alpha_i^{\tilde{n}} = \alpha_i^* &+ \sum_m \frac{\partial \alpha_i^*}{\partial M'_{i,m}} \left(\hat{M}_{i,m} + \delta M_{i,m,advect} - M^*_{i,m} \right) \\ &+ \frac{\partial \alpha_i^*}{\partial E'_{i,P}} \left(\hat{E}_{i,P} + \delta H_{i,P,advect} - E^*_{i,P} \right) \\ &+ \frac{\partial \alpha_i^*}{\partial E'_{i,A}} \left(\hat{E}_{i,A} + \delta H_{i,A,advect} - E^*_{i,A} \right)\end{aligned} \tag{4-35}$$

where the derivatives of the pool void fraction are given by

$$\frac{\partial \alpha_i^*}{\partial X'} = \frac{\frac{\partial \alpha_i^*}{\partial X} + \frac{\partial \alpha_i^*}{\partial P_i} \frac{\partial P_i^*}{\partial X}}{1 - C_i \left(\frac{\partial \alpha_i^*}{\partial M_{B,i}} + \frac{\partial \alpha_i^*}{\partial P_i} \frac{\partial P_i^*}{\partial M_{B,i}} \right) \Delta t} \tag{4-36}$$

in analogy with Equation (4-32), with the understanding that $\partial \alpha^* / \partial X$ is zero unless X is M_1 or E_p.

4.3 Solution Strategy

As written, Equation (4-23) represents a set of linear equations for the latest estimates of the new velocities, $v^n_{j,\varphi}$, and is solved by use of a standard linear equation solver. The complete solution procedure, however, is iterative on two levels. As already mentioned, the code requires convergence of the velocity field, so that the velocities $v^{n-}_{j,\varphi}$ used in the loss terms in Equation (4-23) are acceptably close to the new velocities $v^n_{j,\varphi}$ found by solution of these equations. In general, this involves iteration. In addition, the code requires that the final new pressures and pool void fractions, P^n and α^n, found from the full equation of state for the new masses and energies [Equations (4-8) and (4-9)] agree well with the linearly projected new pressures and void fractions, $P^{\tilde{n}}$ and $\alpha^{\tilde{n}}$, given by Equations (4-17) and (4-35). Once again, iteration may be required, this time on the definition of the point (denoted by "*") about which pressure is linearized in Equation (4-15).

In general, the advancement of the hydrodynamic equations proceeds as shown in Figure 4.1 (details are presented after the general approach has been described).

If either iteration fails to converge, the solution attempt is abandoned, the timestep, Δt is reduced, the external sources are redefined appropriately, and the entire procedure repeated starting from the original "old" state. As already intimated and discussed in detail in Section 4.5, the thermal-hydraulic packages (CVH and FL) may "subcycle" (i.e., several successive advancements may be used to advance the thermal-hydraulic solutions through a full MELCOR system timestep). In general, repetition of the solution with a reduced timestep affects only a subcycle and is restricted to the hydrodynamic packages. Sources are redefined under the assumption that external source rates are constant over a system timestep. If the resulting subcycle timestep would be excessively small with respect to the system step, CVH calls for a MELCOR fallback with *all* packages required to repeat their calculations with a reduced *system* timestep.

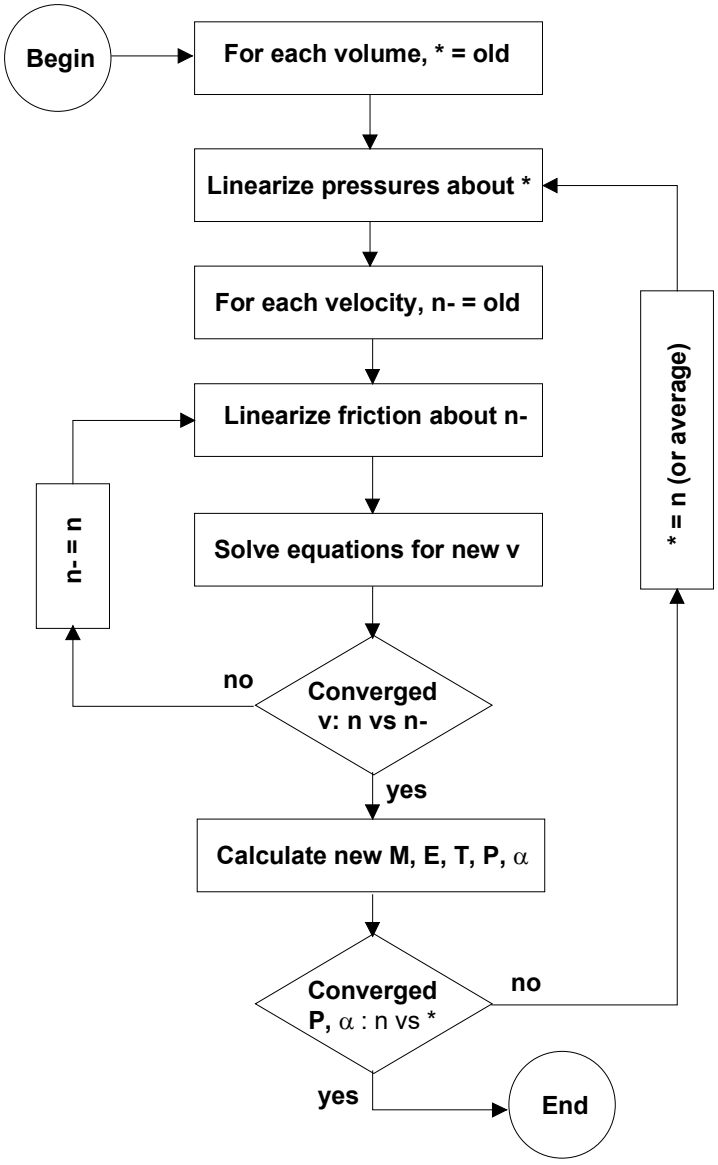


Figure 4.1 Solution flow chart of hydrodynamics equations

To avoid problems with coupling to other packages in MELCOR, large changes in conditions are not permitted to occur during a single system timestep. If any excessive change is observed after the advancement through a system timestep has been completed, the solution is abandoned, and CVH calls for a MELCOR system fallback.

The remainder of this section expands on the general outline given above, discusses special cases, and includes specific details such as convergence criteria.

In the inner (velocity) iteration, the solution of Equation (4-23) is repeated until the new velocities have converged. Convergence requires that no velocity has reversed with

respect to the direction assumed in defining quantities and that no velocity has changed in magnitude by more than 9% compared to the value that was used in linearizing the friction terms. (The latter criterion is coded using an absolute tolerance and a relative tolerance included as sensitivity coefficients in array C4401.) Note that the relatively loose tolerance on magnitudes affects *only* friction terms; conservation of mass and energy is assured by the form of the equations. Our experience has shown that tightening the convergence criterion affects only the details of very rapid transients, which are of little significance in typical MELCOR calculations.

At each iteration, the friction terms are updated, replacing the velocity, v^{n-}, about which they are linearized by the latest iterate, v^{n}, for flows that have not converged. If one or more of the new velocities has reversed with respect to the direction assumed in defining donor quantities, these quantities are also redefined to reflect the correct flow direction. If there are no flow reversals, new velocities are also accepted if the corresponding volumetric flows have converged (subject to the same tolerances), starting with the second iteration. The user may also require that after a number of iterations specified by sensitivity coefficient C4401(4) new velocities are accepted—even if they have not converged to the stated tolerance—if the projected new pressures, $P^{\tilde{n}}$, have converged within 0.05% (comparing successive velocity iterations). The current default is not to accept convergence on this basis.

In some cases, a phase (pool or atmosphere) is available within the junction opening height at only one end of a flow path, and its flow is, therefore, possible in one direction only. If the donoring assumed in construction of Equation (4-23) makes such a flow “impossible,” the corresponding momentum equation is still carried as part of the equation set, but with its coupling to predicted new pressures eliminated by setting the contribution to new mass and energy inventories to zero in Equations (4-8) and (4-9). Therefore, a calculated “impossible” flow has no effect on “real” flows, but its sign indicates the direction the flow would take (if possible) in response to projected end-of-step pressures. If the sign indicates that the calculated new flow remains impossible, the flow is set to zero. If the sign is reversed—and the flow is therefore possible—the equations must be re-solved with the assumed donor definition reversed.

If the iteration fails (either by exceeding the permitted number of iterations or by entering an invalid region of the equation of state defined by the CVT package), the entire set of equations is reformulated with a shorter timestep and re-solved. In general, this is handled within the CVH/FL package by subcycling, rather than by calling for a fallback and a reduction of the MELCOR timestep.

After the new velocities are determined (by convergence of the iterative solution to the finite difference equations), they are used to update the masses and energies in the control volumes through Equations (4-8) and (4-9); in the process, the masses moved by flows are limited to the contents of the donor control volumes. While the mass, momentum, and energy equations could be solved simultaneously, this procedure

ensures that mass and energy are conserved as accurately as possible. Final end-of-step pressures and pool void fractions, P^n and α^n, corresponding to the new masses and energies are now evaluated using the full nonlinear equation of state. If the discrepancy between P^n and $P^{\tilde{n}}$ or α^n and $\alpha^{\tilde{n}}$ in one or more volumes is too great, the entire iterative solution of the momentum equation is repeated (for a maximum of six times), with a modified definition of the point (denoted by "*") about which the equation of state is linearized (described later). The general criterion for convergence of pressure is agreement of P^n and $P^{\tilde{n}}$ within 0.5% [coded as a sensitivity coefficient C4408(2)]. This is tightened to 0.1%, if there is no pool in the control volume and relaxed to 1.0%, if there is no atmosphere. The criterion for convergence of pool void fraction is agreement of α^n and $\alpha^{\tilde{n}}$ within 1.0% [coded as a sensitivity coefficient C4412(1)]. If the outer iteration fails to converge within this tolerance, the subcycle timestep is cut.

The acceptable discrepancy between projected and actual new pressures should *not* be viewed simply as an accuracy tolerance for pressures; it comes into play only when conditions change sufficiently during a timestep that the nonlinearity of the equation of state becomes significant. For example, a large discrepancy between the projected and actual new pressures in a control volume can arise if the state in the volume has crossed the saturation line, going from saturated conditions ($\partial P/\partial M$ relatively small) to subcooled conditions ($\partial P/\partial M$ very large) or vice versa. It can also occur if there has been a change in the hydrodynamic volume (reflecting relocation of virtual volume) as a result of the omission of the term $(\partial P/\partial V)\delta V$ in writing Equation (4-15). In either case, a projection over the entire timestep is invalid. Therefore, in the outer (pressure) iteration, the linearization point is taken as the best available estimate of the "new" state. On the first iteration, it is the "old" state "o"; on subsequent iterations, it is the latest "new" solution. This is illustrated (in a nonrigorous way) by Figure 4.2, which shows the connection to a conventional Newton iteration for a single-variable problem. After the third iteration the linearization point is defined as the *average* of the last two "new" solutions.

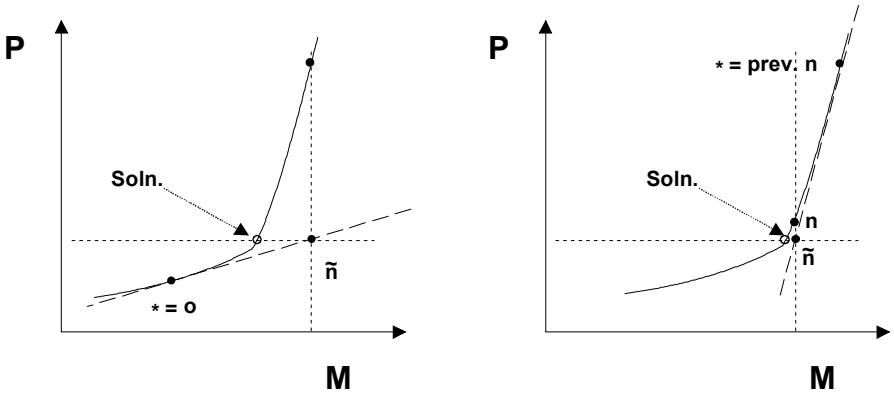


Figure 4.2 Linearization of pressure vs. mass

There is a slight subtlety in the redefinition of the linearization point because the PdV work done by the pool on the atmosphere (or vice versa) in a nonequilibrium volume is calculated in CVT rather than in CVH [note that it does not appear in Equation (4-9)]

$$W = P_i^o(V_P^o - V_P^n) \tag{4-37}$$

based on the old (start-of-step) volume and is transferred from the atmosphere to the pool in CVT. Therefore, if the new results *returned* by CVT are $M_{i,m}^n$, $E_{i,\varphi}^n$, and P_i^n and that solution is rejected, the work must be subtracted from these results to define conditions about which the equations may be linearized. That is, if a solution is rejected, the new linearization point is taken as

$$\begin{aligned} M_{i,m}^* &= M_{i,m}^n \\ E_{i,P}^* &= E_{i,P}^n - P_i^o(V_P^o - V_P^n) \\ E_{i,A}^* &= E_{i,A}^n + P_i^o(V_P^o - V_P^n) \end{aligned} \tag{4-38}$$

where *n* denotes the "new" solution returned by CVT. The essential point is that if M^n, E^n, and V^o are sent as nonequilibrium arguments to CVT, an additional PdV work term is computed, and the pressure and volumes returned are *not* P^n and V^n. If, on the other hand, the arguments sent to CVT are M^*, E^*, and V^o, the work computed there balances that subtracted off Equation (4-38), and the desired values, P^n and V^n, are returned.

Note that the choice of the point * should have little effect on the results obtained (if the solution is successful) because, while the predicted new (end-of-step) pressures are used in the flow equation, they are required by the convergence criteria described earlier to agree well with the actual new pressures. Any small variations in the predictions can have only a modest effect on the results. Therefore, the primary effect of the choice of the point * is on the success of the solution procedure; a poor choice can slow or even prevent convergence.

After the thermal-hydraulic state of the system has been advanced through a MELCOR system timestep, which may involve convergence of the entire calculation described above for several CVH subcycles, the new pressures and temperatures in all control volumes are examined to determine if the changes from old values are acceptably small. The criteria are less than 10% change in pressure and less than 20% plus 1 K change in the temperature of each phase containing more than 1% of the mass in the control volume. These are coded as sensitivity coefficients included in the array C4400. If any change exceeds that permitted, a fallback is requested, and the calculation is repeated with a reduced MELCOR system timestep.

4.4 Definition of Donor Quantities

The preceding discussion concerns only the finite-difference equations and the solution technique. The definition of the donor densities and enthalpies, ρ^d and h^d, in the matrix

coefficients on the left-hand side of the set of flow equations is a completely independent question. (Of course, the choice can affect the accuracy and/or the numerical stability of the entire scheme.)

In the conventional approach, donor quantities are start-of-step ("old") values in the volume from which material is moved; thus, they are not affected by sources. This is consistent with the fact that they are not affected by mass or energy removed—e.g., through flow paths—there are no implicit terms in the donor quantities.

In MELCOR, the sources include changes of material identity resulting from chemical reactions in other packages (COR, BUR, and FDI) as well as from phase changes involving boiling/flashing or fog precipitation within the CVH package itself. The existence of negative mass sources can easily lead to the computation of a negative mass contents in a control volume for one or more materials. An example would be a volume where water vapor was consumed by a clad-oxidation reaction and was also allowed to flow out of the volume through flow paths.

One approach to the problem, as employed by HECTR [2], is to retain the conventional donor definition in terms of pre-source conditions and to use timestep controls to prevent catastrophes. Non-negativity checks on individual material masses are a necessary part of this approach, and negative-mass adjustments must sometimes be employed.

This does not seem practical for use in MELCOR, where, for example, clad oxidation may be extremely rapid. There may be conditions where, in the "real world," *no* steam leaves the volume where the reaction is taking place. However, if any is present at the start of the timestep, some would be calculated to leave it under the conventional definition of donor properties. Reduction of the timestep to follow the kinetics of the reaction is not a viable solution; all available steam is really consumed, leaving *none* available for flow out of the volume. Therefore, the problem is handled in MELCOR by modification of the donor quantities (mass and enthalpy) to include the effects of mass sources. The treatment of energy sources depends on the mass sources, as described below.

Mass additions are treated as taking place at constant pressure and temperature. This is a reasonable approximation if conditions in the control volume do not change much during a timestep. If conditions *do* change significantly, the timestep (or subcycle step) becomes too long, by definition, and is cut as a result of other checks. For each noncondensible gas, for liquid water, and for water vapor, constancy of pressure and temperature implies constancy of the specific volume and of the specific enthalpy. Thus, if liquid water and water vapor are separate materials, donor partial densities and specific enthalpies are unaffected by sources, and only the amount of each material available for flow is changed. In general, a modification of the volume of this material is involved.

Heat sources, as well as the difference between the enthalpy of added materials and the enthalpy that these materials would have at start-of-step conditions, are not included in this definition of donor quantities. For heat sources, this follows conventional practice. For mass sources, we argue that the enthalpy difference is exactly parallel to a simple heat source because new material is mixed and equilibrated with old and that it should, therefore, be treated in the same way as a heat source. The effect of this treatment of sources in MELCOR is to restrict the immediate heating effects of *all* sources to the control volumes in which they occur. While far from a rigorous proof of the correctness of our interpretation, it should be noted that all other approaches tried in the development of MELCOR led to violations of the second law of thermodynamics.

In the current coding, the total post-source mass of each material and its total enthalpy at the pre-source temperature and pressures are calculated, together with the corresponding volume of pool, of fog, and of the gaseous atmosphere. These are used to define donor quantities.

As implied above, addition of mass at constant pressure and temperature requires changes in the volume of the pool, of the fog, and/or of the atmosphere, which must be calculated. There is a complication in that temperature and pressure are not sufficient to define the state of saturated (two-phase) water. Thus, internal energy must be considered to determine the quality of water in the pool and the partition of atmospheric water between vapor and fog.

For a mixture of ideal gases, the total volume is given by

$$V = \sum_m \frac{M_m R_m T}{P} \tag{4-39}$$

where M_m is the mass of species m; R_m is the corresponding gas constant, equal to the universal gas constant divided by the molecular weight; T is temperature; and P is pressure.

This equation is applied to the gaseous atmosphere (subscript A) to yield

$$\delta V_A = \sum_m \frac{\delta M_m R_m T_A^o}{P_A^o} \tag{4-40}$$

where the superscript "o" again denotes old (start of step). The gas constant for water vapor is evaluated as

$$R_{H2O} = \frac{P_{A,\,H2O}}{\rho_{A,\,H2O}^o T_A^o} \tag{4-41}$$

As noted above, temperature and pressure are not sufficient to define the post-source state of two-phase water. It is assumed that sources of atmospheric vapor and fog

remain in those fields for the purposes of defining donor densities. Enthalpies and densities corresponding to the start of the advancement step are used if available; otherwise, appropriate saturation properties are assumed. Similarly, pool sources are now treated as having the same mass quality as the pool mass present at the start of the timestep. If there was none, saturated liquid properties at the old (total) pressure are used.

4.5 Timestep Control and Subcycling

As mentioned in previous sections of this Reference Manual, the thermal-hydraulic packages (CVH and FL) are permitted to subcycle. That is, they may employ several successive substeps to advance the state of the system through a MELCOR system timestep from t^o to $t^n = t^o + \Delta t$. Only the final state (at t^n) becomes part of the MELCOR database.

The code keeps track of the maximum subcycle timestep ($\Delta t_{sub,\max}$) that it is willing to attempt. Each attempted advancement starts from the last point successfully reached, t^{last}, with a step given by

$$\Delta t_{sub} = min\left(\Delta t_{sub,max}, t^n - t^{last}\right) \tag{4-42}$$

Following a failed attempt, $\Delta t_{sub,\max}$ is reduced by a factor of 2. (The possible reasons for failure of a subcycle were discussed in Section 4.3.) Following a successful advancement, it is reevaluated as

$$\Delta t^n_{sub,max} = max\left(\Delta t,\ 1.6F\Delta t^o_{sub,max}\right) \tag{4-43}$$

where F is a factor that allows a faster increase if the convergence of pressures in the outer iteration and the solution of the momentum equation was much closer than required by the tolerance. Specifically,

$$F = max\left(1,\ 2 - 10\frac{|\varepsilon P/P|^o_{max}}{(\varepsilon P/P)_{tol}}()\right) \tag{4-44}$$

where $\varepsilon P/P$ is the relative error in the predicted pressure (compared to the new pressure); the subscripts “max” and “tol” denote a maximum over volumes and a tolerance, respectively; and the superscript “o” again denotes the previous subcycle. The tolerance is coded as a sensitivity coefficient, part of the array C4408(2), with a default value of 0.005.

If the failure of an attempted advancement results in a subcycle length, Δt_{sub}, which is less than $0.01\Delta t$, the timestep is aborted, and the executive level of MELCOR is

directed to perform a fallback. That is, the advancement of *all* packages is repeated from t^o with a reduced value of Δt . As currently coded, this reduction is by a factor of 2.

When, as a result of one or more steps, the thermal-hydraulic packages have advanced the state of the system from t^o to t^n, the changes in pressures and temperatures in all control volumes are examined. As mentioned in Section 4.3, a change of more than 10% in pressure or more than 20% plus 1 K in the temperature of each phase containing more than 1% of the mass in a control volume results in a fallback, where the tolerances are coded as sensitivity coefficients included in the array C4400. As currently coded, the fallback is not performed if the MELCOR system timestep is already within a factor of 2 of the minimum. The change is accepted, and the calculation can continue.

If these tolerances are met, a maximum acceptable timestep is estimated for the next MELCOR step, such that certain stability and accuracy criteria is (most probably) met. This estimate considers several factors.

First, changes in pressures and temperatures must be acceptably small. An acceptable step is estimated based on the rates of change of temperatures and pressures for the just-completed step. For pressures, the change in the pressure of control volume *i* is desired to be no more than 0.0 + 0.05 P_i^o. This is (probably) the case if the timestep, based on pressure change, is not greater than

$$\Delta t_P^n = \min_i \left(\frac{0.0 + 0.05 P_i^o}{\left| P_i^n - P_i^o \right|} \right) \Delta t \qquad (4\text{-}45)$$

where *i* includes all control volumes in the problem. Similar limiting timesteps are estimated for changes in temperature, as

$$\Delta t_{T_\phi}^n = \min_i \left(\frac{1.0 + 0.1 T_{\phi,i}^o}{\left| T_{\phi,i}^n - T_{\phi,i}^o \right|} \right) \Delta t \qquad (4\text{-}46)$$

where ϕ is *P* or *A*. If a phase represents less than one percent of the mass in a control volume, it is excluded from these calculations. All of the constants in Equations (4-45) and (4-46) (including the zero) are coded as sensitivity coefficients, included in array C4400, and can be modified by user input if desired. The default values provide a safety factor of two between the desired maximum changes and the changes that lead to a fallback. Changes in timestep control should be made in parallel with changes in the corresponding fallback criteria.

The (material) Courant condition provides another restriction through the stability requirement that a timestep may not be long enough to permit replacement of all of the material in a volume. (While not a rigorous statement of the condition, this is a workable approximation to it.) This leads to the limitation that the timestep be no greater than

$$\Delta t_{Cou}^{n} = 0.5 \min_{i} \left(\frac{V_i^*}{\Delta V_{i,out}} \right) \Delta t \tag{4-47}$$

where V_i^* is the total volume of materials initially in the volume including mass sources (at the old temperature and pressure, see Section 4.4), and $\Delta V_{i,out}$ is the total volume—pool and atmosphere— moved out of the volume during the timestep. Note that $\Delta V_{i,out}$ accounts for flow *from* volume *i* and flow *to* volume i. The factor of 0.5 is coded as a sensitivity coefficient in the array C4400.

The accuracy of the solution of the momentum equation (as estimated by the linear equation solver) is also considered. It is used to define

$$\Delta t_{Mom}^{n} = \begin{cases} 0.9\Delta t & N < 2 \\ (N - 0.9)\ \Delta t & N \geq 2 \end{cases} \tag{4-48}$$

where *N* is the number of significant figures in the velocities, as estimated by the solver. Note that the factor 0.9 is coded as a sensitivity coefficient in array C4400.

Finally, the timestep given by the most restrictive of the desired CVH constraints

$$\Delta t_{CVH}^{n} = min\left(\Delta t_{P}^{n}, \Delta t_{T_P}^{n}, \Delta t_{T_A}^{n}, \Delta t_{Cou}^{n}, \Delta t_{Mom}^{n}\right) \tag{4-49}$$

is chosen as an upper bound on the acceptable timestep and communicated to the executive routines for consideration in setting the next system timestep.

5. Constitutive Relations

5.1 Pool/Atmosphere Mass and Energy Transfer

When equilibrium thermodynamics is used in a control volume, mass and energy transfer between the pool and the atmosphere is implicitly determined by the assumption that the pool and the atmosphere are in thermal and evaporative equilibrium. In this case, CVT performs the transfers that are, effectively, instantaneous.

If a volume in which nonequilibrium thermodynamics is specified contains both a pool and an atmosphere, CVT does not transfer mass between them and only transfers energy in the amount of the PdV work done by one on the other. CVH must, therefore, calculate the energy exchange at the pool surface—the evaporation or condensation and the phase separation in the pool as bubbles rise—and join the atmosphere as fog settles into the pool. The mass/energy transfer at the pool surface, which is driven by convection and/or conduction and any phase separation resulting from bubble rise, are treated as two separate processes. The deposition of fog is ordinarily treated by the aerosol dynamics portion of the RN package, but a simple, nonmechanistic limit on fog

density, described in Section 5.1.4, is imposed by the CVH package when large fog densities are encountered.

Bubble rise is accounted for only if nonequilibrium is specified. Given the assumption that there are no noncondensible gases in the pool, the equilibrium assumptions prohibit the presence in bubbles in the pool whenever such gases are present. (Total pressure exceeds saturation pressure by the partial pressure of the noncondensible gases. The liquid water is, therefore, subcooled and cannot be in equilibrium with a bubble containing only water vapor.) All water vapor in an equilibrium volume is, therefore, assumed to reside in the atmosphere to avoid a discontinuity in behavior, and the vapor content of the pool is *always* calculated as zero by CVT for equilibrium volumes.

5.1.1 Mass Transfer at the Pool Surface

Calculation of phenomena at the pool surface requires simultaneous solution of the equations of heat and mass transfer. It may be reduced to finding the temperature of the pool surface that satisfies the requirements that the

(1) mass flux (evaporation or condensation) is that given by the mass diffusion equation for the existing gradient in the partial pressure of water vapor between the surface and the bulk atmosphere;

(2) net heat flux delivered to the interface by convection, conduction, and radiation is equal to the latent heat required by the evaporation or condensation heat flux; and

(3) partial pressure of water vapor at the pool surface corresponds to saturation at the surface temperature.

In the presence of noncondensible gases, the mass flux, defined as positive for evaporation, is given by

$$\dot{m}'' = C\ ln\left(\frac{P_A - P_{w,A}}{P_A - P_{w,l}}\right) \qquad (5\text{-}1)$$

where P_A is the total pressure; $P_{w,A}$ is the partial pressure of water vapor in the bulk atmosphere; $P_{w,l}$ is the partial pressure of water vapor at the interface; and C is a coefficient.

This equation is also applied in the absence of noncondensibles, requiring only that $P_{w,l} = P_{w,A}$; it is used in a modified form [Equation (5-6)] in which there is not even the appearance of a singularity.

Using the analogy between mass transfer and heat transfer [9], C is obtained from

$$\frac{C\,L}{\rho_v D} = \frac{h_A L}{k}\left(\frac{Sc}{\mathrm{Pr}}\right)^{1/3} \tag{5-2}$$

where Pr and Sc are the Prandtl and Schmidt numbers given by

$$Pr = \frac{\mu_A c_{P,A}}{k_A} \tag{5-3}$$

and

$$Sc = \frac{\mu_A}{\rho_A D_{w,A}} \tag{5-4}$$

respectively. In these equations, L is a characteristic length that cancels in the final result; h is the coefficient of convective heat transfer; ρ_v is the density of saturated water vapor at total pressure; c_P is the specific heat at constant pressure; μ is dynamic viscosity; k is thermal conductivity; ρ is density; D_w is the mass diffusivity of water vapor; and subscript A refers to the atmosphere.

Properties are calculated for the current bulk atmosphere composition. Density and specific heat are calculated in the CVT package, as described in the CVT Package Reference Manual while the viscosity and thermal conductivity are calculated by the MP package, as described in the MP Package Reference Manual. The general model in the MP package (based on Reference [10] but using the complete composition of the atmosphere) is used.

Conditions at the interface are assumed to be saturated, thus relating the partial pressure at the interface, $P_{w,I}$, to the temperature, T_I, through

$$P_{w,I} = P_{sat}(T_I) \tag{5-5}$$

If Equation (5-1) is solved for $P_{w,I}$, the inverse of Equation (5-5) may be expressed as

$$\mathrm{T_I} = T_{sat}\left[P_A - \left(P_A - P_{w,A}\right)\exp\left(\frac{-\dot{m}''}{C}\right)\right] \tag{5-6}$$

Simultaneous with mass transfer, there are temperature-driven heat flows from the pool to the surface (interface), Q_{PS}, and from the atmosphere to the surface, Q_{AS}. These do not include mass-transfer effects and may be approximated by using ordinary heat transfer correlations. Processes (such as radiation) treated by other packages may also deposit energy directly “in” the surface at a rate Q_{RS}. The net heat flow to the surface is then related to the evaporation rate by

$$\dot{m} = \frac{Q_{PS} + Q_{AS} + Q_{RS}}{h_{fg}} \tag{5-7}$$

where

$$h_{fg} = h_g - h_f \tag{5-8}$$

is the latent heat of evaporation. In current coding, the enthalpies h_f and h_g are evaluated at bulk conditions for the pool and atmosphere, respectively. (Other interpretations are possible, but in all cases investigated, other choices had no significant effect on calculated results.)

The heat flows, Q_{PS} and Q_{AS}, from the pool and atmosphere to the surface, may both be considered to be proportional to the corresponding temperature differences

$$Q_{PS} = h_P^*(T_P - T_l)A_s \tag{5-9}$$

$$Q_{AS} = h_A^*(T_A - T_l)A_s \tag{5-10}$$

where A_S is the surface area of the pool and the h^* are effective heat transfer coefficients, including radiation within the CVH package, as discussed in Section 5.1.2. This allows Equation (5-7) to be solved for T_l in the form

$$T_l = \frac{h_P^* T_P + h_A^* T_A + \left(Q_{RS} - \dot{m}\ h_{fg}\right)/A_S}{h_P^* + h_A^*} \tag{5-11}$$

Equations (5-6) and (5-11) provide two simultaneous equations for T_l and $\dot{m}$ that are solved iteratively with a bound-and-bisect method. The fact that h_p^*, h_A^* and the mass transfer coefficient C are themselves functions of the interface temperature, T_l, is accounted for during the iteration.

In MELCOR, the rate given by this solution is calculated using start-of-step conditions and is then applied to the entire step, Δt .

The resulting transfers of mass and energy are

$$\Delta M_P = -\dot{m}\Delta t \tag{5-12}$$

$$\Delta E_P = -\left(\dot{m}h_f + Q_{PS}\right)\Delta t \tag{5-13}$$

$$\Delta M_{w,A} = \dot{m}\Delta t \tag{5-14}$$

$$\Delta E_A = -\left(\dot{m}\ h_g - Q_{AS}\right)\Delta t \tag{5-15}$$

If condensation is occurring at a rate that exceeds 90% of the total water vapor in the atmosphere during the timestep, the mass transfer is limited to this value to avoid numerical problems. Equations (5-12) through (5-15) are then recalculated so as to conserve mass and energy. The limiting value is coded as a sensitivity coefficient in array C4407.

The energy transfers are written as internal energies, ΔE s, because they are added to the internal energy of the material, but are actually enthalpies, ΔH s.. The difference, $P\Delta V$, is later cancelled by the volume work accounted for in calculations in the CVT package. The necessity for this may be seen by considering a case where essentially all of the pool is evaporated; its energy inventory must be decremented by its total *enthalpy* to ensure that the final energy content is near zero after the work term is accounted for in CVT.

This formulation clearly conserves both mass and energy, with the net heat added to the control volume being

$$\Delta E_P + \Delta E_A = Q_{RS}\Delta t \tag{5-16}$$

as is easily shown from the preceding equations. Note from Equations (5-13) and (5-15) that the use of *bulk* values for h_f and h_g eliminates the possibility of nonphysical cooling of an evaporating subcooled pool or heating of a condensing superheated atmosphere. Other nonphysical results from the explicit numerics are avoided by limiting the sensible heat flow from the pool or atmosphere to the heat content above the interface temperature as

$$Q_{\phi S}\Delta t = min\left[Q^0_{\phi S}\Delta t, M_\phi c_{P\phi}\left(T_\phi - T_l\right)\right] \qquad if T_\phi > T_l \tag{5-17}$$

where M_φ is the phase mass, φ is *P* or *A*, and $Q^0_{\varphi S}$ is the value calculated as described in the following section.

5.1.2 Heat Transfer at the Interface

The heat flows at the pool and atmosphere interface (surface) are calculated from

$$Q_{\phi S} = \left[h_\phi\left(T_\phi - T_l\right) + \sigma_B\left(T_\phi^4 - T_l^4\right)\right]A_S \tag{5-18}$$

where σ_B is the Stefan-Boltzmann constant and all other variables were defined above. Note that view factors and emissivities of unity are assumed in the radiation contributions. The effective heat transfer coefficient, including radiation, is then

$$h_\phi^* = h_\phi + \sigma_B\left(T_\phi^2 + T_l^2\right)\left(T_\phi + T_l\right) \quad (5\text{-}19)$$

The normal heat transfer coefficient, corresponding to convection or conduction in the absence of mass transfer, is defined by

$$h_\phi = max\left(h_{forced,\phi}, h_{free,\phi}, k_\phi/L_\phi\right) \quad (5\text{-}20)$$

The forced convection correlation, taken from TRAC [6], is appropriate for horizontal stratified flow:

$$h_{forced,\phi} = 0.02\rho_\phi c_{P\phi} v_{v,\phi} \quad (5\text{-}21)$$

The control volume average velocity, $v_{v,\varphi}$, is discussed in Section 6.5. The natural convection heat transfer used is taken as the maximum of laminar and turbulent correlations appropriate for horizontal surfaces [11] as

$$h_{free,P} = max\left[0.27(GrPr)_P^{1/4}, 0.27\left(GrPr\right)_P^{1/4}\right]\frac{k_P}{X_P} \quad (5\text{-}22)$$

$$h_{free,A} = \max\left[0.54\left(Gr\,\mathrm{Pr}\right)_A^{1/4}, 0.14\left(Gr\,\mathrm{Pr}\right)_A^{1/3}\right]\frac{k_A}{X_A} \quad (5\text{-}23)$$

where the characteristic dimension is

$$X = min(D_s, L) \quad (5\text{-}24)$$

Here *Pr* is the Prandtl number, defined in Equation (5-3), and *Gr* is the Grashof number,

$$Gr = g\ \beta|\Delta T|X^3(\rho/\mu)^2 \quad (5\text{-}25)$$

In these equations, in addition to variables previously defined, β is the thermal expansion coefficient; *L* is thickness (depth); *g* is the acceleration of gravity; and D_s is the diameter of the surface.

Note that the absolute value of the temperature difference is used in the Grashof number. Therefore, the same correlation is used for both signs of the temperature gradient although it is only appropriate for one of them. In fact, the correlations were derived for rather simpler geometries than currently seen in reactor primary and containment systems. In particular, the effects of other heated or cooled surfaces may well be more important in establishing convection than is the pool surface itself. A recent review of the modeling in MELCOR [12] concluded that “Wall effects are probably sufficiently important and dependent upon geometric details that no general correlation

could be constructed." This review also compared MELCOR to a number of other codes including TRAC [6], RELAP5 [7], HECTR [2], CONTAIN [3] and MAAP [4] and found that "there is no clearly accepted model. Treatment in the other codes suffers from limitations no less significant than those in MELCOR."

In Equations (5-22) and (5-23), the first expression refers to laminar convection and the second to turbulent. Note that the value for $(Gr\,\mathrm{Pr})$ at the laminar-turbulent transition is implicitly defined such that the heat transfer coefficient used is continuous. All of the numerical constants in Equations (5-21), (5-22), and (5-23) are coded as sensitivity coefficients in the array C4407 and may, therefore, be modified through user input. In particular, a laminar-turbulent transition may be introduced into the correlation for free convection in the pool even though there is none in the default version of Equation (5-22). The final term in Equation (5-20), *k/L*, is the conduction limit.

5.1.3 Bubble Rise and Phase Separation

Boiling, as a result of heat deposition in the pool, or flashing, in response to a reduction in the pressure of a control volume, may cause vapor bubbles to appear in the pool. As these bubbles rise to the surface, they transport mass and energy from the pool to the atmosphere. In general, the velocity is insufficient to remove all the bubbles, resulting in a two-phase pool.

The bubble rise model in MELCOR is very simple. It assumes steady state with an upward volume flow of bubbles that varies linearly from zero at the bottom of the control volume to a value of J_{max} at the top and a constant rise velocity, v_o, of 0.3 m/s for the bubbles. This value is approximately correct for typical gas bubbles rising in water under near-atmospheric pressures, where the effect is most important and is not seriously in error under other conditions. (The rise velocity is coded as a sensitivity coefficient in array C4407.) For a volume of constant cross-sectional area, the assumptions correspond to a uniform generation rate of vapor throughout the volume with no bubbles entering the bottom. Other assumptions would lead to different results but within roughly a factor of 2 of those presented here.

Under the stated assumptions, the average void fraction and the volume of bubbles that leave the volume during a time Δt are given by

$$\overline{\alpha} = \frac{J_{\max} Z_P}{2 v_0 V_P} \tag{5-26}$$

$$\Delta V_B \equiv V_B^{tot} - V_B^{final} = J_{max} \tag{5-27}$$

where V_P is the total (swollen) volume of the pool; Z_P is the depth; V_B^{tot} is the sum of the initial volume of bubbles and the volume created in the pool as a result of sources during

Δt ; J_{max} is the upward bubble velocity at the top of the volume; and V_B^{final} is the volume of bubbles remaining at the end of the step.

Therefore, since

$$V_B^{final} = \overline{\alpha} V_P \tag{5-28}$$

the average void fraction may be eliminated to show that only a fraction

$$f = \frac{V_B^{final}}{V_B^{tot}} = \frac{1}{1 + 2v_0 \Delta t / Z_P} \tag{5-29}$$

of the bubbles that were in the pool during the timestep remains after bubble rise is accounted for.

The total mass of vapor in the pool is calculated as

$$M_{v,P}^{tot} = \frac{h_P - h_l}{h_v - h_l} M_P \tag{5-30}$$

where M_P and h_P are the total pool mass and enthalpy, including the vapor component. The specific enthalpies, h_v and h_l, correspond to saturated vapor and liquid, respectively, at the pressure of the control volume ($M_{v,P}^{tot}$ is then limited to M_P). In accordance with Equation (5-29), all but a fraction f is moved to the atmosphere; if this is insufficient to reduce the average void fraction in the pool to 0.40 or less, additional mass is moved to reach that limit. (This limit is coded as a sensitivity coefficient in array C4407. The default value is the approximate upper limit of the bubbly flow regime [13].) The mass moved takes with it the enthalpy of saturated vapor, h_v. The limit is imposed after sources are accounted for and again after the entire flow solution for a CVH subcycle has been successfully completed.

5.1.4 Fog Deposition

Fog in MELCOR consists of water droplets suspended in the atmosphere. If the RN package is active, this fog also forms the water component of the aerosol field treated by the MAEROS [14] model and is subject to various deposition mechanisms. The CVH package has no mechanistic models for fog removal and ordinarily relies on the MAEROS model to calculate these mechanisms. For cases where the RN package is not active, an upper limit [coded as a sensitivity coefficient, C4406(1)] is imposed on the average density of fog in a control volume atmosphere, and excess fog is removed as "rain." (This procedure is also followed if the RN package is active but its calculated aerosol removal rate is insufficient to reduce the fog density below the limiting value.) The default value of the limit is 0.1 kg/m^3 and is based on the practical upper limit

observed in a number of MAEROS calculations. If the fog density in any volume exceeds that limit, the excess is summarily transferred to the pool in that volume. The possibility of such rain is considered after mass sources are added and again after the entire flow solution for a CVH subcycle has been successfully completed.

5.2 Flow Path Void Fractions

The void fraction assigned to a flow path determines the extent to which it is shared by pool and atmosphere. It depends in general on the conditions at the ends of the flow path (its junctions with the *from* and *to* control volumes) and on the direction of flow. Input options are provided to allow the user to override the geometrical calculation performed for normal flow paths and enforce preferential flow of pool or atmosphere. These options are discussed below.

5.2.1 Normal Flow Paths

A flow path connects two control volumes; a void fraction can, therefore, be defined at each junction, based on the fraction of the junction area that lies above the pool surface in the corresponding volume. The void fraction for the *from* connection is calculated as

$$\alpha_{fm} = \frac{z_{TJ,fm} - z_{P,fm}}{z_{TJ,fm} - z_{BJ,fm}} \tag{5-31}$$

where *TJ*, *BJ*, and *P* refer to the top of the junction, the bottom of the junction, and the pool, respectively, and "*fm*" denotes the *from* volume or connection. In effect, the opening is treated as if it were rectangular. The void fraction for the *to* connection is defined similarly.

From these two junction void fractions, a single flow path void fraction must be defined. Unless the flow, based on velocities from the previous iteration in the flow solution, is strictly countercurrent (i.e., pool and atmosphere velocities are nonzero and have opposite signs), the void fraction in the flow path is taken as that at the donor junction. That is, α_j is taken as α_{fm} if the flow is positive and as α_{to} if it is negative. (If there is no flow, so that both velocities are zero, α_j is taken as α_{fm}.)

If the previous iteration flows are countercurrent, the flow-path void fraction is taken as a weighted average of the junction values such that,

$$\alpha_j = \frac{\rho_A^{-1/2}|v_A|\alpha_{Ad} + \rho_P^{-1/2}|v_P|\alpha_{Pd}}{\rho_A^{-1/2}|v_A| + \rho_P^{-1/2}|v_P|} \qquad \text{(countercurrent)} \tag{5-32}$$

where *Ad* and *Pd* refer to the donor junction for atmosphere flow and pool flow, respectively. While there is no rigorous basis for this procedure, it is motivated by an

analysis of flooding and also ensures continuity in the definition as either velocity passes through zero.

There is a further check for over extraction of the pool from the donor volume. The void fraction is modified if necessary to ensure that the volume of pool that would be moved with the previous iteration velocity, $(1-\alpha_j)|v_P|F_j A_j \Delta t$, does not exceed the total volume of pool above the elevation of the bottom of the flow path opening in the pool-donor volume.

There is a similar check for over extraction of the atmosphere based on the previous iteration atmosphere velocity and the volume of atmosphere below the top of the *flow path* opening. These modifications were introduced to eliminate a number of problems with nonconvergence observed in test calculations.

5.2.2 Pool-First and Atmosphere-First Flow Paths

These options allow preferential movement of pool or atmosphere materials through a flow path. This is accomplished by overriding the normal definition of the void fraction for these flow paths. The void fraction is initially set to 0.0 for a pool-first path and to 1.0 for an atmosphere-first path if the preferred phase is present within the junction opening. This α is then subjected to the pool or atmosphere extraction limitation described in the preceding subsection. If the preferred phase is not available, the other phase is permitted to flow in the normal manner.

5.3 Hydrostatic (Gravitational) Heads

The pressure differential acting on phase φ in flow path *J*, connecting control volumes *i* and *k*, was abbreviated in Section 4 at $P_i - P_k + (\rho g \Delta z)_{j,\varphi}$. P_i and P_k are the thermodynamic pressures in control volumes i and k respectively and correspond to the altitudes of the pool surfaces. The term $(\rho\, g \Delta z)_{j,\varphi}$ contains all gravitational head terms within the control volumes and along the flow path. Figure 5.1 illustrates the elevation changes associated with a flow path.

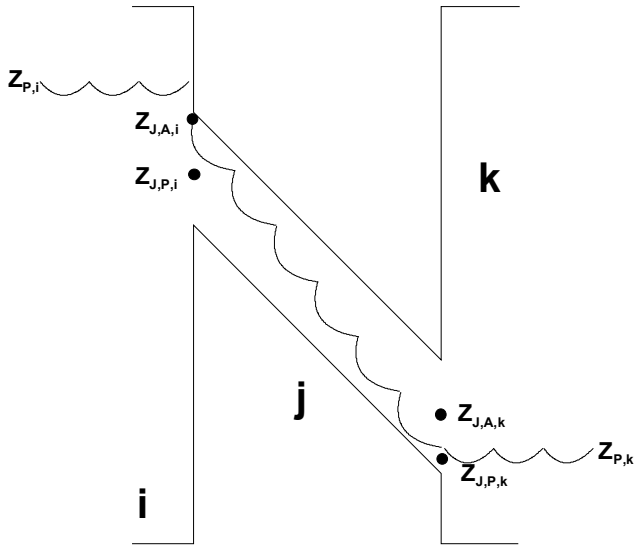


Figure 5.1 Elevations involved in gravitational head terms

Examination of Figure 5.1 shows that there are three contributions to the gravitational head. The first is the pressure difference between the pool surface at $z_{p,i}$ (where the volume pressure is defined) and that at the average elevation, $z_{J,\varphi,i}$, of the phase in the junction opening in volume *i*

$$\left(P_{J,i} - P_i\right)_\varphi = \begin{cases} \rho_{P,i} g\left(z_{P,i} - z_{J,\varphi,i}\right) & z_{P,i} \geq z_{J,\varphi,i} \\ \rho_{A,i} g\left(z_{P,i} - z_{J,\varphi,i}\right) & z_{P,i} < z_{J,\varphi,i} \end{cases} \tag{5-33}$$

In this equation, the average elevations of the phases in the junction openings are given by

$$z_{J,P,i} = max\left\{z_{BJ,i}, \frac{1}{2}\left[z_{BJ,i} + min\left(z_{P,i}, z_{TJ,i}\right)\right]\right\} \tag{5-34}$$

$$z_{J,A,i} = min\left\{z_{TJ,i}, \frac{1}{2}\left[z_{TJ,i} + max\left(z_{P,i}, z_{BJ,i}\right)\right]\right\} \tag{5-35}$$

where *BJ* and *TJ* again refer to the top and bottom of the junction opening, as in Section 5.2.

The second contribution to the static head comes from the corresponding pressure difference in volume *k,*

$$\left(P_{J,k} - P_k\right)_{\varphi} = \begin{cases} \rho_{P,k} g\left(z_{P,k} - z_{J,\varphi,k}\right) & z_{P,k} \geq z_{J,\varphi,k} \\ \rho_{A,k} g\left(z_{P,k} - z_{J,\varphi,k}\right) & z_{P,k} < z_{J,\varphi,k} \end{cases} \quad (5\text{-}36)$$

and the third term is the gravitational head in phase φ along the flow path

$$\left(P_{J,k} - P_{J,i}\right)_{\phi} = \bar{\rho}_{J,\phi} g\left(z_{J,\phi,i} - z_{J,\phi,k}\right) \quad (5\text{-}37)$$

based on the density of that phase in the flow path. The density of a phase in a flow path is taken as the maximum of the volume values

$$\bar{\rho}_{J,\phi} = max\left(\rho^{d}_{J,\phi}, \rho^{a}_{J,\phi}\right) \quad (5\text{-}38)$$

because use of a donor value would introduce an unacceptable discontinuity in the gravitational head whenever the direction of a flow reversed. The donor (*d*) or acceptor (*a*) density, whichever is greatest, is used rather than a simple average of the two because the value in a volume where the phase is not present may not be well defined.

The net gravitational head term is then defined as the sum of these three contributions:

$$(\rho\ g\Delta z)_{j,\phi} = \left(P_{J,i} - P_i\right)_{\phi} + \left(P_{J,k} - P_{J,i}\right)_{\phi} - \left(P_{J,k} - P_k\right)_{\phi} \quad (5\text{-}39)$$

Figure 5.1 shows only two of the three possible cases: $z_P > z_{TJ}$ and $z_{TJ} > z_P > z_{BJ}$, but the third $\left(z_{BJ} > z_P\right)$ should be easily visualized.

The derivatives of Equation (5-39) with respect to pool masses at constant densities are required for the implicit projection of the head terms as shown in Equation (4-16). These are then used in the implicit flow equation, Equations (4-23) and (4-24). Under the assumption of constant pool density, we have

$$\frac{\partial}{\partial M_P} = \frac{1}{\rho_P A_P} \frac{\partial}{\partial z_P} \quad (5\text{-}40)$$

where A_P is the cross-sectional area of the control volume at z_P (the area of the pool surface). Evaluation of the derivatives is greatly complicated by the fact that $\rho_{i,\varphi}, \rho_{k,\varphi}$, and $\bar{\rho}_{J,\varphi}$ are all potentially different. However, by ignoring this difference and neglecting all terms that contain ρ_A rather than ρ_P, we may obtain the approximate result

$$\frac{\partial\left(\rho\, g\Delta z\right)_{j,\varphi}}{\partial M_{P,s}} = \begin{cases} \dfrac{\sigma_{js} g}{A_{P,s}} & z_{P,s} > z_{J,\varphi,s} \\ 0 & z_{P,s} \le z_{J,\varphi,s} \end{cases} \qquad (5\text{-}41)$$

where s is either i or k, and σ_{js} provides the appropriate sign. This approximation has been found to be adequate in practice and is currently employed in MELCOR.

Equation (5-41) may be derived from the preceding equations by performing the indicated derivative under the stated assumptions and approximations. These assumptions and approximations are equivalent to considering only the effect of changes in z_P on the pool contribution to the static head; this observation also allows the equation to be written down by inspection of Figure 5.1.

5.4 Form Loss and Wall Friction

The frictional pressure drops resulting from material flows contain contributions from both form loss and wall friction. The form-loss contribution is based on user input coefficients; the wall-friction terms are computed within MELCOR, based on segment lengths and roughnesses input by the user. Because a single MELCOR flow path may be used to represent a rather complicated hydraulic path, the wall-friction terms may be computed for a path composed of one or more segments that are connected in series. (As noted below, a MELCOR segment may represent a number of parallel pipes.) This approach may also be used to approximately account for frictional losses within the control volumes themselves—MELCOR does not calculate any loss terms based on volume-centered velocities (see Section 6.5).

The flow resistances (and open fractions) for specified flow paths involving core cells are automatically adjusted to represent partial or total blockage of the flow by core debris, as calculated by the COR package. See Section 6.7 for a discussion of this model.

5.4.1 Flow Path Segments

If a flow path j is imagined to consist of a number of pipe-like segments, the total frictional pressure drop for phase ϕ (P or A) is given by

$$\Delta P^{f}_{j,\phi} = \frac{1}{2} K_{j,\phi} \rho_{j,\phi} \left|v_{j,\phi}\right| v_{j,\phi} + \sum_{s} \frac{2 f_{\phi,s} L_s}{D_s} \rho_{\phi,s} \left|v_{\phi,s}\right| v_{\phi,s} \qquad (5\text{-}42)$$

where K is the form loss coefficient for the entire flow path, and f is the Fanning friction coefficient for segment s, which has length L_s and hydraulic diameter D_s. The sum is over the segments in the flow path.

In Equation (5-42), the pressure drops associated with sudden area changes or bends (the *K* term) and wall friction losses for the pipe segments (the *f* terms) are quadratic in velocity but, as written, each term involves a different velocity. For each flow path, MELCOR computes phase velocities $v_{j,P}$ and $v_{j,A}$ for the pool and the atmosphere, respectively. These define the volumetric flow of pool and atmosphere through the flow path,

$$J_{j,\phi} = \alpha_{j,\phi} F_j A_j v_{j,\phi} \tag{5-43}$$

where A_j is the flow path area, and F_j is the fraction of that area that is open. If the flow is assumed to be incompressible, i.e., $\rho_{\varphi,s} = \rho_{j,\varphi}$, the volumetric flow of each phase in the segments is constant, and the segment velocities are given by

$$v_{\phi,s} A_s = v_{j,\phi} F_j A_j \tag{5-44}$$

where A_s is the segment area. (Note that if a segment is to represent a number of parallel pipes, A_s should be the total flow area while D_s should be the hydraulic diameter of each pipe.) Therefore, all the loss terms may be combined to give an effective loss coefficient *K*.

$$K^*_{j,\phi} = K_{j,\phi} + \sum_s \frac{4 f_{\phi,s} L_s}{D_s} \left(\frac{F_j A_j}{A_s} \right)^2 \tag{5-45}$$

The frictional pressure loss can be cast in the following form

$$\Delta P^f_{j,\phi} = \frac{1}{2} K^*_{j,\phi} \rho_{j,\phi} |v_{j,\phi}| v_{j,\phi} \tag{5-46}$$

The input form-loss coefficient for positive or negative flow (FRICFO or FRICRO on input record FL_USL) is used for $K_{j,\varphi}$ depending on the sign of $v_{j,\varphi}$.

The wall-friction terms are calculated following the method of Beattie and Whalley [15]. A mixture Reynolds number is defined for each segment as

$$Re_s = \frac{(\alpha\ \rho_A |v_A| + (1-\alpha) \rho_P |v_P| D_s}{\mu_m} \left(\frac{F_j A_j}{A_s} \right) \tag{5-47}$$

using a mixture viscosity

$$\mu_m = \alpha\ \mu_A + (1-\alpha)(1 + 2.5\alpha) \mu_P \tag{5-48}$$

Here μ_A is calculated by the MP Package for a mixture of gases with the composition of the atmosphere. The viscosity of liquid water is used for μ_P (despite the fact that the pool may contain bubbles). Note that μ_m has the proper limits (μ_P or μ_A, respectively) as α goes to 0.0 or 1.0.

The flow path void fraction computed by MELCOR (Section 5.2.1) is used in Equations (5-47) and (5-48) rather than the homogeneous void fraction originally proposed in Reference [15]. The constants in Equation (5-48) are coded as sensitivity coefficients in array C4404 and may, therefore, be modified by user input if desired.

The Reynolds number calculated from Equation (5-47) is used in a standard single-phase flow friction correlation (which is described in Section 5.4.2) to determine a single-phase friction factor f_1 that is used directly for f_P.

The flow quality

$$x = \frac{\alpha\ \rho_A v_A}{\alpha\rho_A v_A + (1-\alpha)\rho_P v_P} \tag{5-49}$$

is used to interpolate the atmosphere friction factor f_A linearly between the single-phase value f_1 when only atmosphere is flowing in the path (x = 1.0) and zero for $x \le x_0$. (x_0 is coded as sensitivity parameter C4404(12) with a default value of 0.9.) This is intended to reflect the tendency toward annular flow, with the gas phase preferentially occupying the center of a flow path away from the walls and, therefore, is not directly affected by wall friction.

The wall friction terms depend only on the velocity in the segment. Therefore, for a given volumetric flow [Equation (5-43)], they are independent of F (the fraction of the flow path that is open). This is as it should be, since F is intended to model a local restriction such as a valve that has no effect on wall losses in pipe segments.

On the other hand, the entire form loss term (K) depends on the nominal flow path velocity that, for a given volumetric flow, is dependent on F. Thus, if F can vary (i.e., if the flow path contains a valve), F cannot be used to represent the effects of bends, contractions, and/or expansions in that flow path. This is not a serious defect because such losses may be modeled using equivalent lengths of pipe [16] in the segment data. In addition, most valves are either fully open or closed, and the current form is correct in either case. In the future, the restriction may be removed by allowing form loss coefficients to be input for each segment, in addition to a single coefficient value now permitted for the path, with the segment form losses based on the segment velocities rather than the MELCOR flow path velocities.

5.4.2 Single-Phase Friction Factor

The single-phase friction factor correlation used in MELCOR includes laminar, turbulent, and transition regions. In the laminar region, $0 \le \mathrm{Re} \le 2000.0$, the expression used is

$$f = \frac{16.0}{Re} \tag{5-50}$$

The Colebrook-White equation [17]

$$\frac{1}{\sqrt{f}} = 3.48 - 4.0\, log_{10}\left[\frac{2.0\ e}{D} + \frac{9.35}{Re\sqrt{f}}\right] \tag{5-51}$$

is used in the turbulent region $\mathrm{Re} \ge 5000.0$. Here ***e*** is the surface roughness. This equation must be solved iteratively. In the transition region $2000.0 \le \mathrm{Re} \le 5000.0$, $\log(f)$ is linearly interpolated as a function of log(Re) between the limiting values for the laminar and turbulent regimes.

The various constants in these equations, including the limiting Reynolds numbers, are coded as sensitivity coefficients in the array C4404 and may, therefore, be modified by user input.

5.4.3 Two-Phase Friction Factor

To accommodate various two-phase pressure-drop correlations found in the literature, explicit friction factors defined by users have been made available through the FL_F2P record. This record replaces the default friction computation when applied. Users can specify (1) a single friction factor, f, which is applied to both fields within the flow path, (2) a unique friction factor for each field, f_φ, or (3) a two-phase friction multiplication factor, ϕ^2. The two-phase friction factor is specified by the user explicitly and combined with the friction factor computed for the field-only flow, which is computed just prior to the solution of the velocity/pressure equations.

Unlike the default pressure gradient, these additional options are based on user-specified control functions and are therefore explicit and computed from prior timestep values. Both the single friction factor flow path and the unique field friction factors are direct replacements of the friction factors computed by default and should be specified with care as they directly impact the thermal hydraulic solutions for end of timestep velocities and pressures.

The two-phase friction multiplication factors modify a field-only computed friction pressure gradient to determine the two-phase friction pressure gradient. A field-only friction factor is computed assuming the total mass flow is represented by the identified field. By multiplying the two-phase friction multiplier with the field-only computed shear

pressure gradient gives the approximated two-phase friction pressure gradient. The two-phase friction multiplication factors permitted at this time are only for the field-only correlations, provided in open literature, which are determined experimentally and are limited as effects from phase shear are inseparable from wall shear. The multiplication factors are applied assuming homogeneous flow, whereas MELCOR is a separate flow framework. Therefore, investigation of the momentum exchange length should be performed.

Given the multiplication factor along with the assumed phase-only representative flow, the two-phase pressure drop can be determined as provided below.

$$-\frac{\Delta P}{\Delta z}_{2\varphi,fric} = -\varphi^2{}_{yo}\frac{\Delta P}{\Delta z}_{yo,fric} \tag{5-52}$$

$$\frac{\Delta P}{\Delta z}_{yo,fric} = \frac{4\,G_{total}^2\,f_{yo}}{D\,\rho_y} \tag{5-53}$$

Where, $\frac{\Delta P}{\Delta z}$ is the pressure drop per flow path segment length, φ^2 is the two-phase friction factor multiplier, G is the mass flux of flow, D is diameter, ρ is density, f is friction factor, $_{2\varphi}$ is a subscript indicating two-phase, $_{yo}$ is a subscript indicating field y only flow, and total flow is treated as belonging to field y, $_{fric}$ is a subscript indicating friction, and $_{y}$ is a subscript indicating either gas or liquid field and y is set by the two-phase factor multiplier correlation.

By equating a single field pressure drop to the two-phase pressure drop, a modified friction factor is determined for each individual field.

$$\frac{\Delta P}{\Delta z}_{x} = \frac{4\,G_x^2\,f_{mod,x}}{\vartheta_x^2\,D\,\rho_x} \tag{5-54}$$

Here, f_{mod} is the modified friction factor and in this context it is the same as f_ϕ, ϑ is the field flow path area fraction, either α or (1-α), and $_{x}$ is a subscript indicating the given field, either atmosphere or pool.

$$\varphi^2{}_{yo}\,\frac{4\,G_{total}^2\,f_{yo}}{D\,\rho_y} = \frac{4\,G_x^2\,f_{mod,x}}{\vartheta_x^2\,D\,\rho_x} \tag{5-55}$$

$$f_{mod,x} = \varphi^2{}_{yo} f_{yo}\,\frac{G_{total}^2\,\vartheta_x^2}{G_x^2}\,\frac{\rho_x}{\rho_y} \tag{5-56}$$

The friction factors, $f_{mod,x}$ and/or f_{ϕ}, replace the segment friction factor in Equation (5-42). Note, by allowing a user to specify the friction factor, which is a function of the geometry of the flow channel, multiple segments should not be defined when using option (1) or (2) as each could have unique geometry.

To prevent numerical issues for vanishingly small fields when applying the multiplication factor option, sensitivity coefficients C4424(1) and C4424(2) initiate a linear transition between the modified friction factor and the default single-phase friction factor as the quality approaches a single-phase.

5.5 Interphase Forces

The force (momentum exchange) between pool and atmosphere flows sharing a single flow path is important both in entraining concurrent flows and in limiting countercurrent ones. In the latter case, it is responsible for the phenomenon of flooding or countercurrent flow limitation (CCFL).

A model is required for use in MELCOR but without the complicated flow regime maps and constitutive equations of the type employed in TRAC [6] or RELAP5 [7]. Therefore, a simple form is used that reproduces a flooding curve in the form given by Wallis [13]:

$$\left(j_g^*\right)^{\frac{1}{2}} + \left(j_f^*\right)^{\frac{1}{2}} = 1 \qquad (5\text{-}57)$$

where $j_g^* \equiv \alpha v_g / v_1$ and $j_f^* \equiv (1-\alpha) v_f / v_0$ are scaled (dimensionless) volumetric flows of gas and fluid, respectively. In the following, we adopt conventional MELCOR notation, where the conventional subscripts "*g*" and "*f*" become "A" and "P," respectively. As is shown in Appendix B, such a flooding curve results if the relative velocity is modeled as a function of void fraction defined by

$$\frac{1}{v_r} \equiv \frac{1}{v_A - v_P} = \frac{\alpha}{v_1} + \frac{1-\alpha}{v_0} \qquad (5\text{-}58)$$

Here v_1 and v_0 are the scaling velocities used to scale j_A and j_P, respectively; they also turn out to be the values of v_r for the limiting values of α (which are 1.0 and 0.0).

Appendix B also shows that the steady (time-independent) solution of the two-phase momentum equation agrees with this result if the interphase force in Equation (4-6) is represented as

$$f_2 = g(\rho_P - \rho_A)\left(\frac{\alpha}{v_1} + \frac{1-\alpha}{v_0}\right) \qquad (5\text{-}59)$$

In the interest of simplicity, only the following form of v_0 and v_1 [18],

$$\frac{v_0}{v_1} = \sqrt{\frac{\rho_A}{\rho_P}} \tag{5-60}$$

is used in MELCOR. Application of this relationship allows (5-59) to be rewritten as

$$f_2 = \left\{\frac{g(\rho_P - \rho_A)}{v_1\sqrt{\rho_A}}\right\}\left[\alpha\sqrt{\rho_A} + (1-\alpha)\sqrt{\rho_P}\right] \tag{5-61}$$

The MELCOR model treats the term in bracket as a "constant" which by default has a value of 900 $(kg/m^3)^{1/2}/s$. This can be changed by sensitivity coefficient 4404(17) so that non-water systems can be treated in a consistent manner. The default value of 900 is found based on the following specific conditions of water:

(1) The scaling velocity v_0 , which is the limiting value of v_r as α goes to zero, is chosen as 0.3 m/s for vertical flow of gas and corresponds to the terminal rise velocity of bubbles.

(2) A thermodynamic pressure of approximately 4.3 MPa.

These conditions yield a value of $\left\{\frac{g(\rho_P-\rho_A)}{v_1\sqrt{\rho_A}}\right\} \approx 900$ $(kg/m^3)^{1/2}/s$.

As justification for treating the bracketed term as a constant we note that for water this value varies from 1010 to 673 over a pressure range of 0.1 to 15 MPa respectively.

This equation is applied to all geometries, and the results are usually qualitatively acceptable. The term $f_{2,j}$ in the finite-difference Equation (4-23) is multiplied by the length over which the interphase force acts rather than the inertial length of the flow path. A distinct length is used for momentum exchange. The default is taken as the inertial length for horizontal flow paths and as the difference in the elevation between the lowest point and the highest point in vertical flow paths (including junction openings). Optional user input on record FL_LME is allowed to override these defaults for application to special geometries.

5.6 Pumps and Fans

A pump or fan model provides a functional relationship between the pressure head developed by such a device and the volumetric flow through it, with the operating speed as a parameter. Two models are currently available in MELCOR. One simply uses a control function to define the pressure head; this gives the user great flexibility but requires that the user accepts complete responsibility for the results. An example of how this approach could be used to build a conventional homologous model for a reactor coolant pump is outlined in the CF Package Users' Guide. The second model, referred

to as “FANA,” was originally intended to model a containment fan but has also been used as an approximate representation of a constant-speed coolant pump in many calculations. A homologous pump model is presented in Section 6.12.

5.6.1 The FANA Model

This model was originally constructed to represent a simple fan, intended to move air (atmosphere) from compartment to compartment in containment. It can, however, be used to approximate a constant-speed coolant pump by appropriate choice of input parameters.

In the model, a parabolic relationship is assumed between the head, ΔP, developed by the fan and the volumetric flow, $\dot{V}$ through it. Three parameters define the resulting curve:

(1) maximum pressure head developed, ΔP_M;

(2) corresponding volumetric flow, $\dot{V}_M$; and

(3) volumetric flow , $\dot{V}_0$, at which the head is zero.

For a given volumetric flow, $\dot{V}$, the pressure head is then given by

$$\left(\frac{\Delta P}{\Delta P_M}\right)^2 = \begin{cases} 1 & \dot{V} < \dot{V}_M \\ \dfrac{\dot{V}_0 - \dot{V}}{\dot{V}_0 - \dot{V}_M} & \dot{V}_M \le \dot{V} \le \dot{V}_0 \\ 0 & \dot{V}_0 < \dot{V} \end{cases} \qquad (5\text{-}62)$$

The resulting curve is illustrated in Figure 5.2. Suitable parameters may usually be chosen by comparison of this figure with the constant-speed operating curve for the device in question (in the normal operation quadrant).

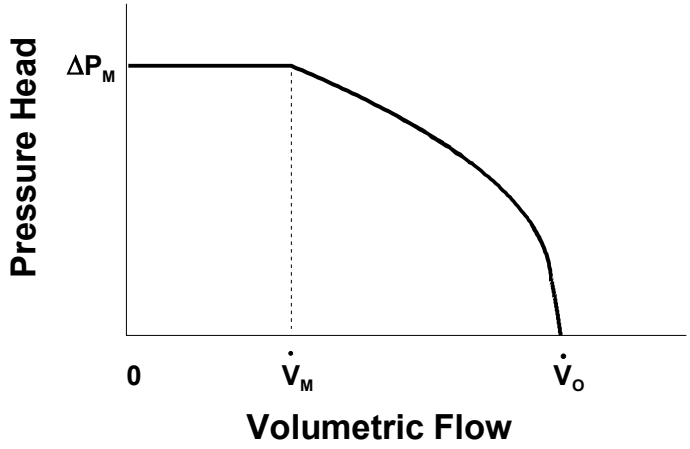


Figure 5.2 Fan model operating characteristics

The “forward” direction for a pump need not correspond to the direction of positive flow in the associated flow path. The necessary sign conventions for treating a reversed pump are described in the FL Package Users’ Guide.

A pump may be specified to be always on, or its operation may be controlled by a tabular function of time or by a control function of other arguments in the MELCOR database. The pump is off if the function is zero and on if it is nonzero. The model is implemented as an explicit momentum source, based on start-of-system timestep velocities. Any functions that control the pump are also evaluated at the start of the MELCOR system timestep and treated as constant over the entire step.

6. Other Models

6.1 Bubble Physics

If a flow of atmospheric materials enters a control volume below the elevation of the surface of the pool in that volume, it must pass through the pool to reach its final destination. This process is visualized as involving rising bubbles in the pool, and the user may specify that an interaction be allowed based on a parametric model of thermal and condensation/evaporation physics. If this option is not selected, no interaction occurs and the transported atmospheric materials are simply added unchanged to the atmosphere in the acceptor volume. A separate pool scrubbing calculation may be done in the RN package using the SPARC90 model [19].

The physics modeled involves breakup of the injected gas stream into a swarm of bubbles, thermal equilibration of the gases with the pool, and saturation of the bubbles with water vapor at local conditions. These bubbles are not considered to reside in the pool and do not contribute to pool swelling. The efficiency of the mass and energy transfer processes is affected by two factors that are treated as independent.

The distance that gases must rise in order to reach the surface of the pool is involved in the breakup of the stream and the corresponding increase in surface area. It is modeled as an efficiency, ε_z, represented as

$$\varepsilon_z = \frac{z_P - z_J - 0.01\ m}{1.0\ h}, 0 \le \varepsilon_z \le 1 \qquad (6\text{-}1)$$

where z_P is the elevation of the pool surface in the acceptor volume; z_J is the junction elevation in the acceptor volume; and h is the height of the junction opening.

No breakup is assumed until the bubbles have risen at least 1 cm, and breakup is assumed to be complete if they rise through the junction opening height plus 1 cm.

The effect of subcooling of the pool is represented as the efficiency

$$\varepsilon_T = \frac{T_{sat}(P) - T_P - 0.1\ K}{5.0\ K}, 0 \le \varepsilon_T \le 1 \qquad (6\text{-}2)$$

This requires local subcooling by at least 0.1 K for any effect and by at least 5.1 K for the maximum possible effect to be predicted. The saturation temperature in Equation (6-2) is evaluated for the pressure at the nominal junction elevation, z_J; this pressure includes the static head corresponding to a depth of $z_P - z_J$.

The overall efficiency is taken as the product of these two efficiencies.

$$\varepsilon = \varepsilon_z \varepsilon_T \qquad (6\text{-}3)$$

If only water vapor and fog are present in the bubbles, it is assumed that a fraction ε of the vapor condenses, and an equal fraction of the fog in the flow path is deposited in the pool, with the remainder passing through to the atmosphere. No modification is made to the specific enthalpy (temperature) of material that passes through. In this case, the entire flow is deposited in the pool if the depth and subcooling are adequate.

If noncondensible gases are present and the depth and subcooling are sufficiently large, it is assumed that bubbles leave the pool at the pool temperature and, further, that the relative humidity in the bubbles is 0.99 (i.e., that the partial pressure of water vapor is 0.99 of the saturation pressure at the pool temperature). If $\varepsilon = 1$, as calculated from Equations (6-1), (6-2), and (6-3), this result is used directly while the trivial result for no interaction is used for $\varepsilon = 0$. For $0 \le \varepsilon \le 1$, a linear interpolation (on the overall ε, Equation (6-3)) is performed between these limits. As in the case of no noncondensibles, a fraction ε of the fog flow is assumed to be deposited in the pool, with the remainder transmitted to the atmosphere.

All constants in this model (those in Equations (6-1) and (6-2) and the limiting relative humidity) are coded as sensitivity coefficients included in array C4405 and may, therefore, be modified by user input. The default values are those discussed here.

The effects of this model are implemented by appropriately modifying the definitions of donor properties. The normal donor properties are used for removal of atmospheric material from the actual donor volume, but a modified set of properties is used for the acceptor volume to which they are added. Specifically, if the volume of atmosphere moved through the flow path is

$$|\Delta V_j| = \alpha_j F_j A_j |\Delta V_{j,A}| \Delta t \qquad (6\text{-}4)$$

then the masses and energies removed from the donor volume, *d*, are

$$\Delta M_{m,d} = -\left|\Delta V_j\right| \frac{M_{m,d}}{V_{A,d}} \tag{6-5}$$

$$\Delta E_{A,d} = -\left|\Delta V_j\right| \frac{H_{A,d}}{V_{A,d}} \tag{6-6}$$

where, of course, the material index *m* in Equation (6-5) is limited to materials in the atmosphere. The masses added to the acceptor volume, *a*, however, have the more general form

$$\Delta M_{m,a} = \rho^*_{m,a}\left|\Delta V_j\right| \tag{6-7}$$

$$\Delta E_{A,a} = (\rho\ h)^*_{A,a}\left|\Delta V_j\right| \tag{6-8}$$

$$\Delta E_{P,a} = (\rho\ h)^*_{P,a}\left|\Delta V_j\right| \tag{6-9}$$

where *m* in Equation (6-7) includes the pool. The bubble physics model gives the masses and energies delivered to the acceptor volume ($\Delta M_{m,a}, \Delta E_{A,a}$ and $\Delta E_{P,a}$) in terms of the entering masses and energies ($\Delta M_{m,d}$ and $\Delta E_{A,d}$). Therefore, Equations (6-7) through (6-9) serve as *definitions* of the quantities $\rho^*_{m,a}$, $(\rho\ h)^*_{A,a}$, and $(\rho\ h)^*_{P,a}$, which are subject to the following constraints.

$$\rho^*_{1,a} + \rho^*_{2,a} + \rho^*_{3,a} = \frac{M_{2,d} + M_{3,d}}{V_{A,d}} \tag{6-10}$$

$$\rho^*_{n,a} = \frac{M_{n,d}}{V_{A,d}}, \quad n \geq 4\ (NCG) \tag{6-11}$$

$$(\rho\ h)^*_{P,a} + (\rho\ h)^*_{A,a} = \frac{H_{A,d}}{V_{A,d}} \tag{6-12}$$

For atmospheric materials, the differences reflect the changes in composition and specific enthalpy described above; the pool terms reflect heat and mass exchange with the pool. If evaporation takes place, $\rho^*_{1,a}$ can be negative. In this case, it is further constrained so that use of Equation (6-7) does not result in a negative pool mass.

6.2 Time-Dependent (Specified) Flow Paths

The velocity in any flow path may be defined by the user, either as a tabular function of time or as a control function of other arguments in the MELCOR database. The resulting

velocity is imposed on both pool and atmosphere (if present), with the void fraction computed using the standard model described in Section 5.2.

6.3 Critical Flow Models

After the solution of the flow (momentum) equation is complete, the computed flow in each flow path is compared with a calculated critical flow to determine if choking should be imposed. The test is bypassed if neither the pool velocity nor the atmosphere velocity is greater than a threshold of 20.0 m/s, coded as a sensitivity coefficient in C4402. If the flow exceeds the critical value, the flow path is added temporarily to a list of specified-flow flow paths, and the entire solution is repeated with the velocity constrained to be the critical value.

If only atmosphere is flowing through the path, the critical mass flux is taken as the sonic flux at the minimum section. For an ideal gas, this may be related to the sonic flux at stagnation conditions through the relation [20]

$$G_{C,A} = \rho_A^d C_{s,A}^d \left(\frac{2}{\gamma+1}\right)^{\left[\frac{\gamma+1}{2(\gamma-1)}\right]} \tag{6-13}$$

where $G \equiv \rho v$ is mass flux; subscript *C* denotes "critical"; C_s is the sonic velocity; and $\gamma \equiv c_p / c_v$ is the ratio of the specific heat at constant pressure to that at constant volume.

The use of the superscript "*d*" reflects the fact that in MELCOR the donor volume is assumed to be at stagnation conditions. The sonic velocity is evaluated in the CVT package. The multiplier is only a very weak function of γ, having a value within 5% of 0.58 for $1.1 \le \gamma \le 1.8$, and is, therefore, evaluated at a nominal value of $\gamma = 1.4$. There are two factors contributing to this function of γ:

(1) reduction in density because of expansion and

(2) reduction in sound speed because of cooling between stagnation conditions and the minimum section.

CONTAIN [3] includes both factors while HECTR [2] only contains the latter.

If only pool is flowing, the RETRAN [21] model (to be discussed in Section 6.3.1) for the critical mass flux is used, based on the pressure and specific enthalpy of the pool,

$$G_{C,P} = G_{C,RETRAN}\left(P^d, h_P^d\right) \tag{6-14}$$

If both phases are flowing, the critical mass flux is taken as a weighted average of that for the two phases

$$\frac{\alpha\ \rho_A^d + (1-\alpha)\rho_P^d}{G_{C,2ph}} = \frac{\alpha\ \rho_A^d}{G_{C,A}} + \frac{(1-\alpha)\rho_P^d}{G_{C,P}} \qquad (6\text{-}15)$$

This rather peculiar averaging scheme was motivated by the observation that it provides an almost exact representation of the Moody choking model if $G_{C,P}$ and $G_{C,A}$ are replaced by $G_{C,Moody}\ (\alpha = 0)$ and $G_{C,Moody}\ (\alpha = 1)$, respectively (see Appendix C).

If the mass flux evaluated using the new velocities calculated by the momentum equation exceeds the appropriate critical value, the velocity imposed (on both phases) is

$$v_{C,J} = \frac{G_C}{\alpha_J \rho_A^d + (1-\alpha_J)\rho_P^d} \qquad (6\text{-}16)$$

Possible improvements in this model are described in Section 7.2.

Discharge coefficients are available (on FL_USL input record) as multipliers for the critical flow values calculated by these models. Different values may be used for forward (positive) and reverse (negative) flows in each flow path; the default value is 1.0. The appropriate discharge coefficient is included both in the test for choking in each flow path and in the velocity imposed if choking is detected. Use of a very large value is the only way to eliminate the possibility of choking in a flow path.

6.3.1 RETRAN Critical Flow Model

The RETRAN critical flow model consists of two 36-parameter, double-polynomial fits to extended Henry-Fauske critical flow for subcooled water (below and above 300 psia) and two 36-parameter fits to Moody critical flow for saturated (two-phase) water (below and above 200 psia), all as functions of stagnation pressure and enthalpy. It also includes a 9-parameter expression for a “transition” enthalpy as a function of pressure. A linear transition is constructed between the Henry-Fauske model at and below this enthalpy and the Moody model at and above saturation. The reader is referred to Reference [21] for a description of the basic models and the fitting procedure employed.

Two modifications to the RETRAN model were made for use in MELCOR. First, the fits are stated in Reference [21] to be valid only above 170 Btu/lbm and were observed to yield unreasonable (sometimes negative) values not far below this value. Therefore, a linear interpolation was introduced between the fit at the lower limit of its applicability and the solution for orifice flow

$$G_O = \sqrt{2P\ \rho_P} \qquad (6\text{-}17)$$

imposed at $h_P = 0$. Second, it was observed that the transition enthalpy that defined the upper bound for application of the Henry-Fauske model was calculated as greater than the enthalpy of saturated liquid at the lower end of the pressure range (below about 21 psia). Therefore, the transition enthalpy was further bounded to be at least 10 Btu/lbm below saturation.

The fits themselves leave something to be desired; they appear to be excessively complicated, include modest discontinuities (several percent) at region boundaries, and have inappropriate extrapolation properties. Plans for improvement are described in Section 7.2.

6.4 Valves

A valve may be included in any flow path in MELCOR. A valve operation can be simply modeled as a change in the fraction of the flow path area that is open or as a change in the flow area as well as loss term.

6.4.1 Flow Path Area Restriction

The original valve model, enabled using either TRIP/NoTRIP on the FL_VLV record, to model a change in the fraction area of the flow path. This fraction may be defined directly as a tabular function of time or as a control function of other arguments in the MELCOR database. Trips may also be used to model irreversible changes in flow areas, such as ruptures of vessels or compartment walls, or to model the hysteresis in the operation of, say, a relief valve. The open fraction is limited to the range $0.0 \le F \le 1.0$, and if the controlling function returns a value outside this range, it is suitably truncated. The upper bound corresponds to a flow area equal to that input for the flow path, and the lower bound corresponds to a closed path in which no flow is permitted.

Additionally, flow paths can be defined to permit only one-way flow, either forward or reverse. Such flow paths provide a simple way to represent idealized check valves. MELCOR also allows the open fractions (and flow resistances) for specified flow paths involving core cells to be automatically adjusted to represent partial or total blockage of the flow by core debris, as calculated by the COR package. See Section 6.7 for a discussion of this model.

6.4.2 Valve Flow Coefficient as a Loss Term

This new method is typically used in characterizing flow properties of valves. The user indicates a valve flow coefficient, C_v, is used by specifying 'NoTRIPCV' on the FL_VLV input record. A control function that specifies the value of C_V for the valve must be identified. Note, that it is possible to convert between a valve flow coefficient and resistance coefficient, K, so it is possible to define a C_v for any fitting.

$$\Delta P = \left(\frac{fL}{D} + \sum K\right)\frac{\rho v^2}{2} \quad (6\text{-}18)$$

By definition, a valve has a C_v of 1 when a pressure of 1 psi causes a flow of 1 US gallon per minute of water at 60°F (i.e. SG = 1) through the valve. Since the pressure drop through a valve is proportional to the square of the flow rate, the relationship between C_v, flow rate and pressure drop can be expressed as:

$$C_v = Q * \sqrt{\frac{SG}{\Delta P}} \quad (6\text{-}19)$$

Q = Flow in gallons per minute (gpm)

C_v = Valve flow coefficient

ΔP = Difference in pressure between inlet and outlet (psi)

SG = specific gravity of liquid relative to water at 60°F.

Similarities between Equations (6-18) and (6-19) can then be used to derive a relationship between K and Cv:

$$C_v = \frac{D^2}{\sqrt{K}} * \left[\frac{\pi}{4}\sqrt{\frac{2}{\rho_0}}\right] \quad (6\text{-}20)$$

ρ_0 = density of water at 60 F.

D = the pipe diameter

Currently, the valve loss coefficient must be defined only for a single segment flow path and MELCOR applies the diameter from this segment. Standard engineering units for this parameter are gpm/sqrt(psi) are expected. The model converts the control function C_v value into a dynamically calculated loss coefficient, K. The user can still specify multiple segments for this flow path. However, if the pipe diameter from the first segment is applied, no wall friction loss is calculated. If a valve loss coefficient is specified for a flow path, neither a wall-friction loss calculation is performed nor any user-supplied loss coefficient on the FL_USL record is used.

6.5 Volume-Averaged Velocities

Volume-averaged (centered) velocities are used in MELCOR only in the calculation of forced-flow heat transfer coefficients (in a number of packages). This is because both

control volume kinetic energies and momentum flux terms are neglected in the governing hydrodynamic equations. The only forced-flow heat transfer coefficients used in the CVH or FL packages are those associated with the pool atmosphere interface in nonequilibrium volumes (Section 5.1.2).

MELCOR is a lumped-parameter code that is often used to model three-dimensional volumes. A rigorously defined volume-averaged velocity would involve multi-dimensional effects, but the essential geometric information is simply not available. The model used in RELAP5 [7], which is also a lumped-parameter code, was considered for use in MELCOR. It may be written in the form

$$J_{V,\varphi} = \alpha_{V,\varphi} v_{V,\varphi} A_V = \frac{1}{2}\left(\sum_{j\ to\ V} J_{j,\varphi} + \sum_{j\ fm\ V} J_{j,\varphi} \right) \quad \text{(RELAP5 [7])} \tag{6-21}$$

$$J_{j,\varphi} = \alpha_{j,\varphi} v_{j,\varphi} F_j A_j \tag{6-22}$$

where J is volumetric flow; $\varphi = P\ or\ A$ and denotes pool or atmosphere; A_V is the flow area associated with volume V; α_V, φ are the area fractions for the volume flows; and all other symbols have been defined before. The sums in Equation (6-21) are over flow paths that connect *to* or *from* volume V.

Volume flows and velocities calculated from Equation (6-21) are strongly dependent on the logical direction of flow paths. For example, reversing both the sign of a velocity and the associated direction of positive flow (so that the actual volume moved from and the volume moved to are unchanged) does *not* preserve the volume flow. In particular, the net flow in a volume, with a flow +*J to* it and +*J from* it, is +J while the net flow in a volume, with +*J to* it and -*J to* it, is zero. This is because it is assumed in the RELAP5 formulation that all *to* connections are on the left of a volume and all *from* connections on the right. In the second case cited above, the flows cancel and there is no resulting flow at the volume center.

We have found that this is often not the desired result in MELCOR nodalizations. Furthermore, the expected results cannot be obtained in any nodalization that connects volumes in a regular grid to approximate a finite-difference representation of a two-dimensional region; the best that can be done is to calculate the velocity *component* along one diagonal of the grid. Therefore, MELCOR uses a simplification of Equation (6-21) that treats all flow paths on an equal footing:

$$J_{v,\varphi} = \alpha_{V,\varphi} V_{v,\varphi} A_V = \frac{1}{2} \sum_j \left| J_{j,\varphi} \right| \quad \text{(MELCOR)} \tag{6-23}$$

where the sum is over all connected flow paths, and the void fraction associated with the volume flow is taken as a simple weighted average over connected flow paths in the form

$$\alpha_V = \frac{\sum_j \alpha_j F_j A_j}{\sum_j F_j A_j} \tag{6-24}$$

This model can be understood qualitatively using the simple argument that, under steady conditions, a flow *through* a volume is counted twice: once where it enters the volume and once where it leaves. It makes no attempt to assign a direction to the volume velocity and would, therefore, be unacceptable if it were necessary to calculate the momentum-flux terms arising from $\nabla \cdot (\rho vv)$. In accord with this simple double-counting argument, a term is added to the sum in Equation (6-23) for the vapor flow to account for vapor generation in boiling in a nonequilibrium volume.

6.6 Special (Time-Specified) Volumes

MELCOR hydrodynamics allows boundary conditions to be defined by specifying the state of one or more volumes as functions of time. This is frequently necessary for simulation of experiments. It is also useful for defining the outside containment environment for a full reactor plant calculation.

In the simplest case, a volume may be specified as time-independent, with properties that do not change as the calculation progresses. Volumes can also be defined whose properties are maintained constant for a specified period, after which they are “freed” to function as normal volumes. This can simplify initialization of an operating steady state in a reactor. An initially time-independent pressurizer enforces a constant pressure boundary condition while initially time-independent steam generators enforce a constant thermal boundary condition during a pre-transient phase of the calculation.

In addition, several options are available for specifying the pressures, temperatures, and compositions of boundary volumes as functions of time, in terms of user-defined tabular functions, external data files, or control functions, as explained in the CVH Users’ Guide.

A time-specified volume can serve any of the functions of a normal volume. It can provide boundary conditions for in- or out-flows or for heat transfer. However, no volume-averaged velocity (Section 6.5) is calculated for a time-specified volume; forced convection heat transfer is, therefore, not considered in the HS package. All phenomena modeled by the RN package are treated, with the sole exception that radionuclides are not allowed to advect *out* of such a volume. (This is intended to prevent radionuclides from reentering a failed containment building from the environment.) A time-specified volume can also be used in conjunction with a time-specified flow path (Section 6.2) to

define a mass source with well-defined properties. This approach is particularly useful for water sources, for which temperature alone is insufficient to define the complete thermodynamic state. It also provides a way for gas sources to be made to participate in the bubble interactions described in Section 6.1.

Any mass or energy transferred to or from a time-specified volume is recorded as "created" in the CVH package for accounting purposes.

6.7 Core Flow Blockage

MELCOR includes a core flow blockage model to account for the changes in flow resistance in the degraded core states that arise during a postulated reactor accident. It treats the entire range of degradation from partially blocked rod geometry to debris bed geometry. The markedly increased resistance to flow in severely degraded geometries is particularly important because it limits the flow available to carry away decay heat and to provide steam for core oxidation. In addition to improving the basic modeling, inclusion of blockage effects has been found to improve code performance, particularly when a detailed CVH nodalization is used in the core region. The neglect of blockage can lead to prediction of non-physical large flows through regions containing very little fluid; the material Courant condition then forces extremely small timesteps, greatly increasing execution times.

At the start of a MELCOR calculation, the core is (usually) in a state for which the representation of friction (in terms of user input for intact geometry) is appropriate. This changes, however, following relocation of core materials. The blockage model, when invoked, modifies flow areas and flow resistances to account for the effects of refreezing of conglomerate debris onto fuel rods and/or other structures or a loss of simple rod geometry through the creation, or relocation of particulate debris.

The current model considers two flow regimes. For severely damaged core geometries, after particulate debris has been formed, it uses correlations developed for flow in porous media. Until this occurs, a simple modification to the flow resistance in intact geometry is used to account for changes in flow area associated with refrozen conglomerate debris. (Clad ballooning, which would have a similar effect, is not modeled.) As currently coded, the switch in regimes is made on a flow path by flow path basis, triggered by the first appearance of particulate debris in any core cell associated with the flow path. When the uncertainty in predicting the actual geometry of core debris is considered, we believe that this simple treatment is adequate for MELCOR use.

6.7.1 Debris Geometry

There are several correlations for the pressure drop for flow in porous media that can all be represented in the general form

$$(\Delta P)_{pourous\ bed} = \frac{1}{2}\rho\, j^2 \frac{L}{D_p}\left(\frac{1-\varepsilon}{\varepsilon^3}\right)\left[C_1 + C_2\left(\frac{1-\varepsilon}{\mathrm{Re}}\right) + C_3\left(\frac{1-\varepsilon}{\mathrm{Re}}\right)^{C_4}\right] \qquad (6\text{-}25)$$

where j is the superficial velocity (volumetric flux), ε is the porosity of the medium, D_p is the effective particle diameter, and Re is the Reynolds number based on these quantities,

$$Re = \frac{\rho\ j\ D_p}{\mu} \qquad (6\text{-}26)$$

The average velocity of fluid in the medium (strictly, the average of the *component* of that velocity that lies in the direction of positive net flow) is given by

$$v \equiv \frac{j}{\varepsilon} \qquad (6\text{-}27)$$

This is further discussed by Dobranich [22], who lists coefficients for four published correlations in a table equivalent to Table 6.1.

Table 6.1 Coefficients in Friction Correlations for Porous Media

Correlation	C_1	C_2	C_3	C_4	Reference
Ergun (original)	3.5	300.	0.0	-	[25]
Modified Ergun (smooth)	3.6	360.	0.0	-	[23]
Modified Ergun (rough)	8.0	360.	0.0	-	[24]
Achenbach	1.75	320.	20.0	0.4	[24]

This correlational form is used to calculate the effects of core blockage on flow resistance once particulate debris has been formed. The coefficients in the correlation were coded as a sensitivity coefficient array, with $C_i = C4413(i)$; default values for I = 1, 2, and 3 are those for the original Ergun Equation [25].

In any flow path for which the blockage model has been invoked, the average porosity, ε, of core cells in the flow path is calculated from the ratio of hydrodynamic volume to total volume in the cells. This accounts for the effects of particulate and refrozen (conglomerate) debris as described in the COR Package Reference Manual. In addition, the open fraction, $F_j(t)$, for that flow path is set equal to the porosity, ε, as an internally defined valve model. As a result, the nominal velocity in the flow path, v_j, calculated by MELCOR is consistent with the velocity in Equation (6-27), so long as the nominal area of the flow path, A_j, is equal to the geometric area, A_{geo}, of the cell(s)

involved. After particulate debris has been formed, the pressure drop in Equation (6-25) can be cast in the form of an effective loss coefficient (to be used with the nominal velocity, v) as

$$K^* = K_{empty} + \left[C_1 + C_2 \frac{1-\varepsilon}{\mathrm{Re}} + C_3 \left(\frac{1-\varepsilon}{\mathrm{Re}} \right)^{C_4} \right] \frac{(1-\varepsilon)L}{\varepsilon D_p} \qquad (6\text{-}28)$$

to replace the "normal" value in Equation (5-46). Here, the Reynolds number expressed in terms of that nominal velocity is

$$Re = \frac{\rho \ \varepsilon \ v_j D_p}{\mu} \qquad (6\text{-}29)$$

and a term K_{empty} has been added to define the flow resistance in the "empty" path that results when no core materials remain. The porosity is 1.0, and the porous medium model—used outside its range of applicability—would predict no friction.

6.7.2 Interpretation of Flow Areas

The nominal area and the open fraction are specified as part of user input to the FL package. In the regular nodalization of a finite difference code, there would be no need to distinguish the nominal area associated with a cell-boundary flow from the geometric area of the associated cell boundary. However, the distinction is essential in a control volume code such as MELCOR, where the definition of control volume geometry is limited and arbitrary interconnection of volumes is allowed. This is because a flow path must be able to represent the connection of a duct or pipe to a room or plenum as well as the boundary surface between two sections of a larger room or volume.

To avoid complications, MELCOR requires that the nominal flow path area be equal to the geometric area of the core cell(s) for all flow paths in which the blockage model is used. In order to eliminate the need for changes to existing decks when flow blockage modeling is added, the user input area is replaced by the geometric area, and the initial open fraction is simultaneously redefined as the porosity associated with core cell(s) in the flow path for all flow paths in which the blockage model is invoked. The redefined values are flagged in MELGEN and MELCOR output as having been modified by the Flow Blockage model.

This may modify the open area, FA, associated with the initial geometry, which results in different values being calculated for the velocity. However, because the advection terms in MELCOR hydrodynamics depend only on the total volumetric flow

$$J_j = j_j A_j(t) = F_j(t) v_j(t) A_j \qquad (6\text{-}30)$$

(see Equations (4-2) and (4-5)), as do the wall-friction terms (see the discussion following Equation (5-42)), only the form loss coefficient used for intact geometry must be adjusted to compensate for the change in open area. (For more discussion, see the final report on the model in Reference [26].)

The input form loss coefficient is replaced by an "equivalent" coefficient, K_{eqv}, that is related to that input by the user through

$$\frac{K_{eqv}}{[F(0)A_{nom}]^2} = \frac{K_{input}}{\left(F_{input}A_{input}\right)^2} \tag{6-31}$$

for which the calculated pressure drop of intact geometry matches that which would be calculated from the user input area, open fraction, and form loss. All such values are flagged in MELGEN and MELCOR output as having been modified by the Flow Blockage model.

6.7.3 Transition Between Intact and Debris Geometries

If there is a period before the first appearance of particulate debris in any core cell associated with a flow path during which there is conglomerate debris frozen onto fuel rods (or other structures), the resulting reduction in flow area is accounted for by modification of the calculation for intact geometry. The presence of such material changes the porosity and, therefore, the open fraction for a flow path. However, the contribution of wall losses, represented by segment data, ordinarily dominates the pressure drop and—as calculated—this contribution is independent of the open fraction of the flow path. Therefore, a multiplier is applied to the friction calculated for intact geometry to account for the actual change in flow area, fluid velocity, and wall friction resulting from the presence of conglomerate debris prior to rod failure. The modified pressure drop is calculated as

$$(\Delta P)_{net,transition} = \left[\frac{\varepsilon(0)}{\varepsilon(t)}\right]^2 (\Delta P)_{int\,act} \tag{6-32}$$

6.8 Flashing of Superheated Water

If superheated liquid water enters a control volume at an elevation above the pool surface, some fraction of it flashes to vapor. Another fraction is dispersed as liquid droplets that are small enough to remain suspended in the atmosphere for a significant time. A new optional model is available to capture some of these effects. If the water is superheated at the pressure of the receiving volume, the model accounts for stagnation and equilibration at that pressure. Although the model does not explicitly account for heat transfer, at least part of the effect is captured when the partitioned water vapor, fog, and pool liquid are equilibrated with the previous contents or the volume.

The partition between liquid and vapor is calculated from the average enthalpy, and a fraction of the liquid is assigned to the “fog” field. By default, this is taken as the fraction of a Rosin-Rammler distribution that lies below a maximum diameter. If the RN1 package is active, the cut-off diameter is taken as the maximum aerosol size treated by MAEROS (DMAX on record RN1_ASP, with a default of 50 μm). If the RN1 package is not active, the maximum size is defined by a sensitivity coefficient with the same default value, 50 μm. The user may also specify the fraction directly through a sensitivity coefficient. By default, the Sauter mean diameter that characterizes the Rosin-Rammler distribution is defined by a sensitivity coefficient with a default value of 65 μm. The user can choose to replace this default, on a case-by-case basis, with either a different constant value or the value of a REAL control function. If, after equilibration, the mass of fog in any volume exceeds the maximum permitted, the excess is added to the pool of that volume.

If the MELCOR RN package is active, water droplets in the atmosphere are considered to behave as aerosols. In this case, the Rosin-Rammler distribution is ordinarily used to distribute the new fog over the water aerosols sections (size bins). A user-option switch is provided to defeat this, in which case the RN1 package calculates the distribution based on the assumption that the new fog resulted from condensation. However, the resulting average droplet diameter is typically far too small, and the option should be used only for purposes of testing and demonstration.

6.8.1 Flashing and Fog Formation

Consider first the case of pool entering a volume through a flow path. Because the MELCOR hydrodynamic equations do not include the volume-average kinetic energy, the control volume enthalpy is best thought of as a stagnation enthalpy. For any adiabatic flow (it need *not* be isentropic), $h + \frac{1}{2}v^2$ is constant along a streamline. Thus, in the MELCOR representation where v^2 is neglected, enthalpy is "conserved" in flows between volumes.

If the flashing model is selected, the process is modeled as a transformation that takes place “within” a flow path. If a flow of liquid water enters a volume at a point where it is able to achieve pressure equilibrium *and* dissipate its kinetic energy before encountering any boundaries, its state before considering heat and mass transfer involving fluids already in the volume is completely determined by its specific enthalpy and the pressure in the receiving volume. (For a two-phase flow or one containing noncondensibles, a precise treatment would include coupling of liquid and vapor, but this is probably "second order" compared to the current total neglect of flashing.) The transformed flow, in general, contains both liquid water and water vapor. The vapor is added to the atmosphere, and some portion of the liquid is retained in the atmosphere as fog. The implementation is parallel to that for the SPARC model described in Section 6.1. The details are controlled by sensitivity coefficient array 4500.

For water sources, subjecting added “water” to the same transformations can capture flashing effects. This requires the definition of paired mass and enthalpy sources for which the enthalpy source is attributable directly to the mass source. (In general, only the total mass and enthalpy added to a volume are currently significant.) It also requires definition of an entrance elevation for use in evaluating the flashing efficiency. The details are controlled by sensitivity coefficient array 4500 in analogy to the flashing flow model.

6.8.2 Relationship between Fog and Water Aerosol in CVH

The MELCOR RN package models aerosol behavior, among other things. It considers a distribution of aerosol sizes for one or more materials; one of those materials is always water. The treatment is based on the MAEROS code and includes agglomeration, gravitational settling, and several deposition mechanisms. If the RN package is active, it treats airborne liquid water as water aerosol.

6.8.2.1 Non-Hygroscopic Aerosol Model

The RN package is ordinarily run without activating the hygroscopic aerosol option. In this case, the water aerosol is identified with the fog field in the hydrodynamics (CVH and FL) packages. The main distinction between fog and water aerosol is that there is no need for CVH to consider a size distribution for fog. However, changes in fog and in water aerosol are calculated separately.

Total masses of the two are consistent at the start of each time step. The change in the total fog/water-aerosol mass in a control volume that results from settling or deposition is calculated by RN1 (MAEROS). Any further change that results from sources or from phase change is then calculated from purely thermodynamic considerations by CVH/FL. Both fog and aerosols are advected through flow paths.

At this point in the time advancement, the total masses of fog and water aerosol in a volume may differ, and a second part of RN (RN2) is used to reconcile them. It assumes that the change in the total airborne water mass is the result of condensation or evaporation and calculates the effect on the size distribution based on the Mason equation.

6.8.2.2 Hygroscopic Model

If the hygroscopic model is activated, this identification is no longer possible because hygroscopic and surface-tension (Kelvin) effects modify the saturation curve for the liquid water in aerosols. As a result, it does not satisfy the same equation of state as the water in pools or films. As implemented, airborne liquid water is considered to be present only as a water aerosol. The RN package models its condensation or evaporation, governed by the Mason equation and defines a corresponding source or sink of water vapor for CVH. In addition, the mass of any water aerosols that are deposited onto a heat structure or pool surface is added to the film or pool mass,

respectively. If fog is produced in a volume, it is subsumed into the water-aerosol field on the next time step, and the effect on the size distribution is calculated based on an assumption of condensation. Appropriate terms are included in various energy equations to account for all of these processes.

6.8.3 Consequences of Separate Modeling of Fog and Water Aerosol

There are significant differences for calculations with and without fog sources. It should be noted that the distinction between sources of fog (material 2) and of water vapor (material 3) is largely one of labeling. Both are added to the atmosphere, and the associated added enthalpy should determine whether the actual state is liquid, vapor, or mixed-phase.

6.8.3.1 Case of No Fog Sources

In the absence of fog sources, the previous modeling in MELCOR should be reasonably consistent, even though there is no size distribution associated with fog. Changes in the aerosol size distribution result from agglomeration, deposition, settling, and from condensation or evaporation. Within each volume, the former processes are directly modeled by the MAEROS equations while the effect of the latter on the aerosol size distribution is modeled by the Mason equation.

If the hygroscopic model is not activated, RN simply uses the Mason equation to calculate the change in size distribution consistent with the total change in airborne water mass that was calculated from thermodynamics by CVH. When fog enters a volume through a flow path, it is accompanied by water aerosol, with fractional inventories of each moved in proportion to the fraction of the atmosphere that is relocated. Thus—up to condensation or evaporation effects—fog entering a volume effectively brings with it the size distribution of water aerosol in the donor control volume.

If the hygroscopic model is activated, RN calculates condensation or evaporation directly. Because there could be no fog in the control volume at the start of a step, any that was formed in CVH must have resulted from condensation and is so interpreted by RN. Any initial fog mass is transferred from CVH to RN. Within RN, it is treated as water vapor in the Mason equation, so that the pre-condensation state is effectively supersaturated.

6.8.3.2 Case of Fog Sources

If a calculation includes a fog source—either explicitly as material 2 or implicit in a water vapor (material 3) source that is less than saturated—there is no size distribution associated with that source. By default, the effective size distribution is calculated in RN under the assumption that the increase in airborne water mass corresponded to condensation on existing aerosols. This may not represent the actual process, and the flashing model provides a far superior representation.

6.8.3.3 Case of Flashing Source or Flow

If the flashing model has not been selected, the size distribution of water droplets from flashing is calculated under the assumption that the increase in airborne water mass corresponded to condensation on existing aerosols and is almost certainly incorrect. In contrast, the models for flashing flows and water sources calculate a distribution of droplet sizes and directly increment the inventory of water-class aerosol appropriately. This is done before the advection of aerosols is calculated, so that the appropriate size information is propagated through flow paths. Because the droplets are created at saturation, other change in fog mass is dominated by condensation or evaporation and is represented reasonably well by the existing modeling.

6.9 Droplet Size Distribution

In a recent summary report [27], Witlox and Bowen suggest the use of either a log-normal or a Rosin-Rammler distribution [28] for droplets formed by flashing of liquid jets. Brown and Wohletz [29] have shown that the Rosin-Rammler distribution is equivalent to the Weibull distribution often used for sequential fragmentation, which suggests that it may be more appropriate of the two.

The Rosin Rammler distribution is a two-parameter distribution. It may be written as

$$\frac{M(d > d_P)}{M_{Total}} = exp\left[-\left(\frac{d_p}{\sigma}\right)^k\right] \tag{6-33}$$

where $M(d>d_P)$ is the cumulative mass of all particles with diameter greater than d_P, M_{Total} is the total mass, σ is a characteristic diameter, and k is a fitting parameter. The two parameters are the exponent, k, and the characteristic diameter, σ.

The conventional measure of size for a droplet distribution is the Sauter mean diameter, $\bar{d}_S$, defined by

$$\left\langle\frac{Area}{Volume}\right\rangle = \frac{6}{\bar{d}_s} \tag{6-34}$$

where $\langle ... \rangle$ denotes an average value; this diameter matches the surface-to-volume ratio for the distribution as a whole. If we assume a constant density, σ can be eliminated from Equation (6-33) in favor of $\bar{d}_S$ to obtain

$$\frac{M(d > d_P)}{M_{Total}} = exp\left\{-\left[\Gamma\left(\frac{k-1}{k}\right)\right]^{-k}\left(\frac{d_p}{\bar{d}_S}\right)^k\right\} \tag{6-35}$$

where Γ is the gamma function. We note that, under the interpretation of Reference 29, the exponent k can be related to a "fractal dimension," D_f, where $0 \leq D_f < 3$, by $k = 3 - D_f$.

Although this suggests that k should not exceed 3, larger values are often obtained when fitting experimental data.

Eltkobt [30] has proposed a droplet size distribution equivalent to

$$\frac{M(d > d_p)}{M_{Total}} = exp\left[-0.44\left(\frac{d_p}{\bar{d}_s}\right)^{5.32}\right] \qquad (6\text{-}36)$$

Equation (6-36) is not precisely consistent with the general form of the Rosin-Rammler distribution, Equation(6-35): if the exponent is 5.32; the coefficient in the exponential should be $\left[\Gamma\left(\frac{4.32}{5.32}\right)\right]^{-5.32} = 0.473$, rather than 0.44. However, this changes the Sauter mean diameter for the actual distribution by a factor of $(0.473/0.44)^{1/5.32} = 1.014$, which is quite insignificant. We have chosen to implement the general form, Equation (6-35), in MELCOR with a default ELtkobt's value of k = 5.32.

For a Rosin-Rammler distribution in the form of Equation (6-35) or Equation (6-36), the second parameter is the Sauter mean diameter. The authors of Reference 27 conclude that current understanding is insufficient to justify detailed modeling of the average size of droplets produced by flashing flows. They state that, while there are some clear trends (for example, increasing superheat tends to decrease the average droplet size), many proposed correlations all lack adequate validation.

Witlox and Bowen state in the "Conclusions and Recommendations" section of Reference 27 that average diameters "under flash break-up conditions for low pressure (<20 bar) … range between the limits of 20 μm and 80 μm," and suggest use of values of 70 μm for low superheat (< 40°C) and 30 μm for high superheat (> 40°C).

Razzaghi [31] has performed numerical simulations based on a specific model of the flashing and droplet formation processes and presented plots of the calculated size distributions. In his Figure 6, the diameter at the peak of the *number* distribution ranges from about 6 μm for flashing from 10 MPa and 550 K to 13 μm for flashing from 5 MPa and 475 K. While these should not be directly interpreted as Sauter mean diameters for use in Equation 3, they seem significantly smaller than those recommended in Reference 27.

However, Razzaghi notes that the calculated diameters are strongly dependent on the values used for modeling parameters, particularly on the assumed "duration of the

inertial growth period of bubble nuclei." His paper also cites a calculated droplet size of 25 μm for flashing of a 590-K jet, and compares it to values of 16 to 76 μm reported for such a jet by Koestel, Gido, and Lamkin [32]. The latter range is close to that recommended by Witlox and Bowen [27], which tends to support their position.

The current default in MELCOR is to use a constant value of 65 μm. Until and unless better guidance and/or an appropriate model can be developed, we recommend the use of sensitivity studies guided by engineering judgment.

6.10 Integral Heat Exchanger Model

MELCOR now contains an integral model for heat exchangers. In previous versions, the user was required to construct a heat exchanger using control volumes, flow paths, and heat structures. A rather detailed nodalization was often required to capture the temperature profiles within heat exchanger tubes and/or shells. The necessarily small size of the volumes often forced the code to take very short time steps.

In contrast, the integral model simply connects two flow paths and implicitly accounts for the temperature profiles within the heat exchanger primary and secondary. Heat is transferred *within* the flow paths using quasi-steady relationships familiar from the "Number of Transfer Units" (NTU) formulation of heat exchanger performance. The heat removal or addition is effectively imposed on the actual downstream volume in each flow path.

The model is implemented as part of the FL package.

6.10.1 Representation of a Heat Exchanger

There are two flow paths in the MELCOR representation of an integral heat exchanger, each connecting two volumes as shown in Figure 6.1. For example, flow path 1, connecting volume 1A to volume 1B, might represent the tubes, and flow path 2, connecting volume 2A to volume 2B might represent the tubes and the shell, respectively, in a tube and shell heat exchanger. If both flows are positive and volume 1A is hotter than volume 2A, the effect is to reduce the enthalpy of the fluid added to volume 1B and increase that of the fluid added to volume 2B. The MELCOR representation is also valid for off-design conditions where one or both of the flows may be reversed with respect to "normal" operation.

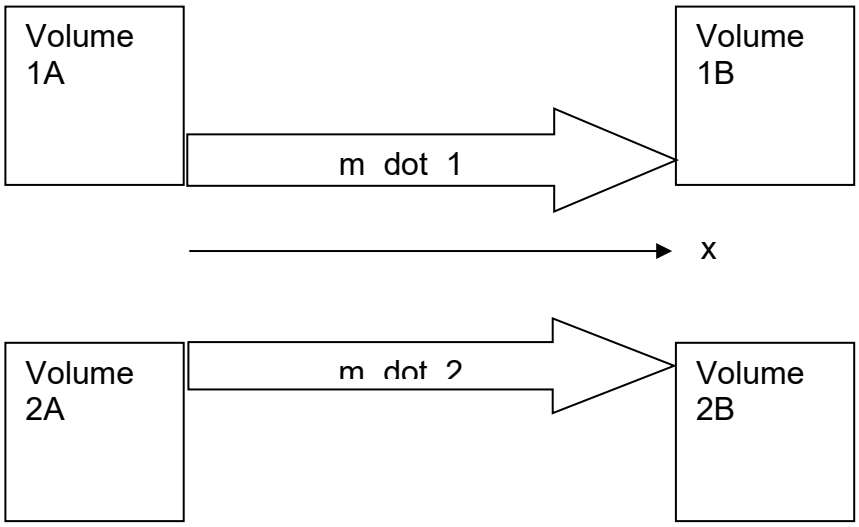


Figure 6.1 MELCOR Representation of a Heat Exchanger

6.10.2 Quasi-Steady Heat Exchanger Performance

There are several designs for heat exchangers. Each has an implicit temperature profile $T_1(x,t)$, $T_2(x,t)$, but we consider only quasi-steady operation and, therefore, won't include time in the equations. This is consistent with the MELCOR interpretation that flow paths have no fluid inventory or residence time.

6.10.3 Parallel Flow

Consider the geometry in Figure 6.1 with

$$\dot{m}_1 > 0, \quad \dot{m}_2 > 0 \tag{6-37}$$

The energy equation is

$$\dot{m}_1 C_{P1} \frac{dT_1}{dx} = -\dot{m}_2 C_{P2} \frac{dT_2}{dx} = U \frac{A}{L} (T_2 - T_1) \tag{6-38}$$

Here U is an effective heat transfer coefficient that accounts for convective heat transfer between each fluid and the wall of the tube (or whatever it may be) that separates the fluids as well as for conduction through that wall.

It is convenient to define the dimensionless variable H that represents the ratio of convective heat transfer to advective heat transport

$$H \equiv \frac{\langle UA \rangle}{|\dot{m}| C_P} \tag{6-39}$$

The angle brackets in $\langle UA \rangle$ reflect the fact that it represents an average because, in general, *U* is a function of position. This group appears throughout the analysis and is a measure of the relative importance of heat transfer to flow heat capacity in limiting the performance of the heat exchanger. It has been defined in terms of the absolute value of the mass flow in the interest of reducing confusion over sign conventions.

In terms of these variables, the energy equation becomes

$$\frac{1}{H_1}\frac{dT_1}{dx} = -\frac{1}{H_2}\frac{dT_2}{dx} = \frac{T_2 - T_1}{L} \tag{6-40}$$

Integrating the first equality from *x*=0 (the inlet) to x=L (the outlet) leads to

$$\frac{T_{1,out} - T_{1,in}}{H_1} = -\frac{T_{2,out} - T_{2,in}}{H_2} \tag{6-41}$$

a general result that simply expresses conservation of energy. In addition, the temperature difference must satisfy the equation

$$\frac{1}{T_2 - T_1}\frac{d}{dx}(T_2 - T_1) = -\frac{H_1 + H_2}{L} \tag{6-42}$$

The result of integrating this equation from x=0 to x=l is

$$H_1 + H_2 = ln\left(\frac{T_{2,in} - T_{1,in}}{T_{2,out} - T_{1,out}}\right) \tag{6-43}$$

The total heat transfer may be written in terms of inlet and outlet temperatures as

$$\begin{aligned} \dot{Q}_{2\to1} &= \dot{m}_1 C_{P1}\left(T_{1,out} - T_{1,in}\right) = \langle UA \rangle \frac{T_{1,out} - T_{1,in}}{H_1} \\ &= -\dot{m}_2 C_{P2}\left(T_{2,out} - T_{2,in}\right) = -\langle UA \rangle \frac{T_{2,out} - T_{2,in}}{H_2} \end{aligned} \tag{6-44}$$

6.10.3.1 Design

For design purposes, it is conventional to assume steady state operation with all temperatures known. The variables H_1 and H_2 can then be eliminated to obtain

$$\dot{Q}_{2\to1} = \langle UA \rangle \frac{\left(T_{1,out} - T_{1,in}\right) - \left(T_{2,out} - T_{2,in}\right)}{H_1 + H_2}$$
$$= \langle UA \rangle \frac{\left(T_{2,in} - T_{1,in}\right) - \left(T_{2,out} - T_{1,out}\right)}{ln\left(\frac{T_{2,in} - T_{1,in}}{T_{2,out} - T_{1,out}}\right)} \quad (6\text{-}45)$$

This is normally written in terms of the so-called “log mean temperature difference” (LMTD) as

$$\dot{Q}_{2\to1} = \langle UA \rangle \; LMTD \quad (6\text{-}46)$$

where

$$LMTD_{PARALLEL} = \frac{\left(T_{2,in} - T_{1,in}\right) - \left(T_{2,out} - T_{1,out}\right)}{ln\left(\frac{T_{2,in} - T_{1,in}}{T_{2,out} - T_{1,out}}\right)} \quad (6\text{-}47)$$

6.10.3.2 Transient

It is common to assume quasi-steady operation under transient conditions, and the MELCOR model is formulated in this way. However, because only the inlet temperatures are known, Equation (6-46) is not useful, and the equations must be recast in to define the heat transfer in terms of the difference in *inlet* temperatures.

To this end, T_{1out} and T_{2out} can be eliminated by using Equation (6-41) in the form

$$H_1 T_{2out} + H_2 T_{1out} = H_1 T_{2in} + H_2 T_{1in} \quad (6\text{-}48)$$

and Equation (6-42) in the form

$$T_{2out} - T_{1out} = (T_{2in} - T_{1in})\, exp(-Z) \quad (6\text{-}49)$$

where

$$Z \equiv H_2 + H_1 \equiv \frac{\langle UA \rangle}{|\dot{m}_2| C_{P2}} + \frac{\langle UA \rangle}{|\dot{m}_1| C_{P1}} > 0 \quad (6\text{-}50)$$

The result is

$$T_{1out} = \frac{H_2 T_{1in} + H_1 T_{2in}[1 - exp(-Z)]}{H_1 + H_2} \quad (6\text{-}51)$$

and the net heat transfer can then be written as

$$\dot{Q}_{2\to1} = \dot{m}_1 C_{P1}(T_{1out} - T_{1in}) = \langle UA\rangle \frac{T_{1out} - T_{1in}}{H_1} = \langle UA\rangle \frac{1 - exp(-Z)}{Z}\left(T_{2,in} - T_{1,in}\right) \tag{6-52}$$

6.10.3.3 Relation to the NTU Formulation

This result can be obtained from the so-called "Number of Transfer Units" (NTU) formulation, expressed (with the original notation changed to that of this report) as in Reference [33],

$$\dot{Q}_{2\to1} = \varepsilon \dot{Q}_{2\to1,max} = \varepsilon C_{min}\left(T_{2,in} - T_{1,in}\right) \tag{6-53}$$

where

$$C_{min} = \left(|\dot{m}_1| C_{P,1}, |\dot{m}_2| C_{P,2}\right) \tag{6-54}$$

is the lesser of the two flow heat capacities. The effectiveness, ε, is given by

$$\varepsilon = \frac{1 - \exp\left[-N_{tu}\left(1 + R_C\right)\right]}{1 + R_C}$$

$$N_{tu} \equiv \frac{\langle UA\rangle}{C_{\min}}$$

$$R_C \equiv \frac{C_{\min}}{C_{\max}}$$

$$C_{\max} \equiv \max\left(|\dot{m}_1| C_{P,1}, |\dot{m}_2| C_{P,2}\right)$$

This can be cast in a more symmetric form as

$$\varepsilon C_{\min} = \frac{1 - \exp\left[-\frac{\langle UA\rangle}{C_{\min}} - \frac{\langle UA\rangle}{C_{\max}}\right]}{\frac{1}{C_{\min}} + \frac{1}{C_{\max}}} = \langle UA\rangle \frac{1 - \exp\left[-\left(\frac{\langle UA\rangle}{C_{\min}} + \frac{\langle UA\rangle}{C_{\max}}\right)\right]}{\frac{\langle UA\rangle}{C_{\min}} + \frac{\langle UA\rangle}{C_{\max}}} \tag{6-55}$$

after which identification of

$$\frac{\langle UA\rangle}{C_{\min}} + \frac{\langle UA\rangle}{C_{\max}} \equiv \frac{\langle UA\rangle}{|\dot{m}_1| C_{P,1}} + \frac{\langle UA\rangle}{|\dot{m}_2| C_{P,2}} \equiv H_1 + H_2 \equiv Z \tag{6-56}$$

results in the form derived above, just derived as Equation (6-52).

$$\dot{Q}_{2\to1} = \varepsilon C_{min}\left(T_{2,in} - T_{1,in}\right) = \langle UA\rangle \frac{1 - exp(-Z)}{Z}\left(T_{2,in} - T_{1,in}\right) \qquad (6\text{-}57)$$

6.10.3.4 Limits and Numerical Issues

In the limit of large heat transfer coefficient and/or small flow, Z becomes very large, and

$$\frac{1 - exp(-Z)}{Z} \xrightarrow[Z\to0]{} \frac{1}{Z} \qquad (6\text{-}58)$$

$$\dot{Q}_{2\to1} \xrightarrow[Z\to0]{} \frac{\left(T_{2,in} - T_{1,in}\right)}{\frac{1}{|\dot{m}_2|C_{P2}} + \frac{1}{|\dot{m}_1|C_{P1}}} \qquad (6\text{-}59)$$

This reflects the fact that the heat transfer is limited by the heat carrying capacity of the fluid. Further, if one heat capacity is much less than the other

$$\dot{Q}_{2\to1} \xrightarrow[Z\to0,|\dot{m}_1|C_{P1}<<|\dot{m}_2|C_{P2}]{} |\dot{m}_1|C_{P1}\left(T_{2,in} - T_{1,in}\right) \qquad (6\text{-}60)$$

In the limit of small heat transfer coefficient and/or large flows, Z goes to zero and

$$\frac{1 - exp(-Z)}{Z} \xrightarrow[Z\to0]{} 1 \qquad (6\text{-}61)$$

$$\dot{Q}_{2\to1} \xrightarrow[Z\to0]{} \langle UA\rangle\left(T_{2,in} - T_{1,in}\right) \qquad (6\text{-}62)$$

This reflects the fact that there is there is little temperature change through either side of the heat exchanger. For small Z, $[1-\exp(-Z)]/Z$ becomes indeterminate, but we use the expansion

$$\frac{1 - exp(-Z)}{Z} \approx 1 - \frac{1}{2}Z + \frac{1}{6}Z^2 - \frac{1}{24}Z^3 \qquad (6\text{-}63)$$

which has a relative error less than 10^{-6} for $|Z|<0.10$. For large Z, we can also use the approximation

$$1 - exp(-Z) \approx 1 \qquad (6\text{-}64)$$

which has a relative error less than 10^{-6} for $Z>14$.

6.10.4 Counter flow

Consider the geometry in Figure 6.1 with

$$\dot{m}_1 > 0, \quad \dot{m}_2 < 0 \tag{6-65}$$

The energy equation is

$$\dot{m}_1 C_{P1} \frac{dT_1}{dx} = -\dot{m}_2 C_{P2} \frac{dT_2}{dx} = U \frac{A}{L} (T_2 - T_1) \tag{6-66}$$

In terms of the *H* variables defined earlier, this is

$$\frac{1}{H_1} \frac{dT_1}{dx} = \frac{1}{H_2} \frac{dT_2}{dx} = \frac{T_2 - T_1}{L} \tag{6-67}$$

Integrating the first equality from *x*=0 (the inlet for 1 and outlet for 2) to x=L (the outlet for 1 and inlet for 2), we find

$$\frac{T_{1,out} - T_{1,in}}{H_1} = \frac{T_{2,in} - T_{2,out}}{H_2} \tag{6-68}$$

In addition, the temperature difference then satisfies the equation

$$\frac{1}{T_2 - T_1} \frac{d}{dx} (T_2 - T_1) = \frac{H_2 - H_1}{L} \tag{6-69}$$

Integrating from x=0 to x=l results in the equation

$$H_2 - H_1 = ln \left(\frac{T_{2,in} - T_{1,out}}{T_{2,out} - T_{1,in}} \right) \tag{6-70}$$

and the net heat transfer can be written as

$$\begin{aligned} \dot{Q}_{2\to1} &= \dot{m}_1 C_{P1} \left(T_{1,out} - T_{1,in} \right) = \langle UA \rangle \frac{T_{1,out} - T_{1,in}}{H_1} \\ &= \dot{m}_2 C_{P2} \left(T_{2,out} - T_{2,in} \right) = -\langle UA \rangle \frac{T_{2,out} - T_{2,in}}{H_2} \end{aligned} \tag{6-71}$$

6.10.4.1 Design

For design purposes, where all temperatures are known, this may be cast as

$$\dot{Q}_{2\to1} = -\langle UA\rangle \frac{\left(T_{2,out} - T_{2,in}\right) + \left(T_{1,out} - T_{1,in}\right)}{H_2 - H_1}$$
$$= \langle UA\rangle \frac{\left(T_{2,in} - T_{1,out}\right) - \left(T_{2,out} - T_{1,in}\right)}{ln\left(\frac{T_{2,in} - T_{1,out}}{T_{2,out} - T_{1,in}}\right)} \quad (6\text{-}72)$$

As with the parallel flow case, this is normally written in terms of the so-called “log mean temperature difference” (LMTD) as

$$\dot{Q}_{2\to1} = \langle UA\rangle \; LMTD \quad (6\text{-}73)$$

where

$$LMTD_{Countercurrent} = \frac{\left(T_{2,in} - T_{1,out}\right) - \left(T_{2,out} - T_{1,in}\right)}{ln\left(\frac{T_{2,in} - T_{1,out}}{T_{2,out} - T_{1,in}}\right)} \quad (6\text{-}74)$$

6.10.4.2 Transient

As in Section 6.10.3.2, and for the same reasons, the unknown outlet temperatures must be eliminated. Using Equation (6-68) in the form

$$H_1 T_{2,out} + H_2 T_{1,out} = H_1 T_{2,in} + H_2 T_{1,in} \quad (6\text{-}75)$$

and Equation (6-70) in the form

$$exp(2Y)\, T_{2,out} + T_{1,out} = T_{2,in} + exp(2Y)\, T_{1,in} \quad (6\text{-}76)$$

or, more symmetrically

$$exp(Y)\, T_{2,out} + exp(-Y)\, T_{1,out}$$
$$= exp(-Y)\, T_{2,in} + exp(Y)\, T_{1,in} \quad (6\text{-}77)$$

where

$$Y \equiv \frac{H_2 - H_1}{2} \quad (6\text{-}78)$$

In terms of Y, $T_{1,out}$ is given by

$$T_{1,out} = \frac{\left[H_1\left(T_{2,in} - T_{1,in}\right) + H_2 T_{1,in}\right] exp(Y) - H_1 T_{2,in}\, exp(-Y)}{H_2\, exp(Y) - H_1\, exp(-Y)} \tag{6-79}$$

and the net heat transfer can be written

$$\begin{aligned} \dot{Q}_{2\to1} &= \dot{m}_1 C_{P1}\left(T_{1,out} - T_{1,in}\right) \\ &= \langle UA \rangle \left(T_{2,in} - T_{1,in}\right) \frac{exp(Y) - exp(-Y)}{H_2\, exp(Y) - H_1\, exp(-Y)} \end{aligned} \tag{6-80}$$

By use of

$$\begin{aligned} H_2 &= \frac{H_2 + H_1}{2} + \frac{H_2 - H_1}{2} = \frac{H_2 + H_1}{2} + Y \\ H_1 &= \frac{H_2 + H_1}{2} - \frac{H_2 - H_1}{2} = \frac{H_2 + H_1}{2} - Y \end{aligned} \tag{6-81}$$

This may be cast in the form

$$\begin{aligned} &\dot{Q}_{2\to1} \\ &= \langle UA \rangle \left(T_{2,in} \right. \\ &\left. - \ T_{1,in}\right) \ \frac{exp(Y) - exp(-Y)}{\frac{H_2 + H_1}{2}[exp(Y) - exp(-Y)] + Y[exp(Y) + exp(-Y)]} \\ &= \langle UA \rangle \left(T_{2,in} - T_{1,in}\right) \frac{1}{\frac{H_2 + H_1}{2} + Y\, coth(Y)} \end{aligned} \tag{6-82}$$

6.10.4.3 Relation to the NTU Formulation

This result is again equivalent to the NTU formulation. The result from Reference [33] is

$$\varepsilon = \frac{1 - exp[-N_{tu}(1 - R_C)]}{1 - R_C\, exp[-N_{tu}(1 - R_C)]} \tag{6-83}$$

with N_{tu} and R_C defined as before. This can be cast in a more symmetric form through a series of manipulations. First,

$$\varepsilon C_{\min} = \langle UA \rangle \frac{1 - \exp\left[-\frac{\langle UA \rangle}{C_{\min}} + \frac{\langle UA \rangle}{C_{\max}} \right]}{\frac{\langle UA \rangle}{C_{\min}} - \left(\frac{\langle UA \rangle}{C_{\min}} + \frac{\langle UA \rangle}{C_{\max}} \right) \exp\left[-\frac{\langle UA \rangle}{C_{\min}} + \frac{\langle UA \rangle}{C_{\max}} \right]} \tag{6-84}$$

With the identification that

$$\frac{\langle UA \rangle}{C_{min}} = \frac{1}{2}\left(\frac{\langle UA \rangle}{C_{min}} + \frac{\langle UA \rangle}{C_{max}} \right) - \frac{1}{2}\left(\frac{\langle UA \rangle}{C_{min}} - \frac{\langle UA \rangle}{C_{max}} \right) = \frac{H_{min} + H_{max}}{2} + \frac{H_{min} - H_{max}}{2} \tag{6-85}$$

$$\frac{\langle UA \rangle}{C_{max}} = \frac{1}{2}\left(\frac{\langle UA \rangle}{C_{min}} + \frac{\langle UA \rangle}{C_{max}} \right) + \frac{1}{2}\left(\frac{\langle UA \rangle}{C_{min}} - \frac{\langle UA \rangle}{C_{max}} \right) = \frac{H_{min} + H_{max}}{2} - \frac{H_{min} - H_{max}}{2} \tag{6-86}$$

Equation (6-82) becomes

$$\begin{aligned} \varepsilon C_{\min} &= \langle UA \rangle \frac{1 - \exp(-Y')}{\frac{H_{\min} + H_{\max}}{2}[1 - \exp(-Y')] + Y'[1 + \exp(-Y')]} \\ &= \langle UA \rangle \frac{1}{\frac{H_{\min} + H_{\max}}{2} + Y' \frac{1 + \exp(-Y')}{1 - \exp(-Y')}} \\ &= \langle UA \rangle \frac{1}{\frac{H_{\min} + H_{\max}}{2} + Y' \coth(Y')} \end{aligned} \tag{6-87}$$

where

$$Y' \equiv \frac{H_{min} - H_{max}}{2} \tag{6-88}$$

Interchange of “min” and “max” simply changes the sign of Y’, and Equation (6-87) is even in Y’. Therefore, the subscripts “min” and “max” may be replaced by “1” and “2”, yielding the symmetric result derived in the preceding section:

$$\begin{aligned} \dot{Q}_{2\to1} &= \langle UA \rangle \left(T_{2,in} - T_{1,in} \right) \\ &= \langle UA \rangle \left(T_{2,in} - T_{1,in} \right) \frac{1}{\frac{H_2 + H_1}{2} + Y \, coth(Y)} \end{aligned} \tag{6-89}$$

6.10.4.4 Limits and Numerical Issues

In the limit of large heat transfer coefficient and/or small flow, H_2, H_1, and Y are all small, and

$$Y\,coth(Y) \xrightarrow[Y\to 0]{} 1$$
$$\dot{Q}_{2\to 1} \xrightarrow[Y\to 0]{} \langle UA\rangle \left(T_{2,in} - T_{1,in}\right) \tag{6-90}$$

As with the parallel case, this reflects the fact that there is little temperature change through either side of the heat exchanger. For small Y, $Y\coth(Y)$ becomes indeterminate, but we can use the expansion

$$Y\,coth(Y) \approx 1 + \frac{1}{3}Y^2 - \frac{1}{45}Y^4 \tag{6-91}$$

which has a relative error less than 10^{-6} for $|Y|<0.28$. We can also approximate

$$coth(Y) \approx 1 \tag{6-92}$$

with a relative error less than 10^{-6} for $|Y|>7.3$

6.10.4.5 Off-Normal Operation in MELCOR

The user must specify, as part of MELGEN input, whether the heat exchanger is parallel or counter-flow for positive flow in both paths (normal operation) on user record FL_IHX. On each time step, MELCOR determines the actual regime based on current flows and applies either Equation (6-50) for parallel flow or Equation (6-81) for counter flow, as appropriate.

6.11 Stratified Flow Model

When two connected volumes contain fluids of different densities in a gravitationally unstable configuration (i.e., with a denser fluid above a lighter fluid), there is a tendency to evolve towards a stable configuration with the denser fluid on the bottom. This often occurs by establishment of separated counter-current flow of denser fluid downwards and lighter fluid upwards.

MELCOR models this naturally for cases where one fluid is pool (dominated by liquid) and the other atmosphere (dominated by gas), as described in Section 5.5, Interphase Forces, because these are modeled by different fields (phases) that can share a flow path. The difference in the pressure plus gravitational head that drives the two flows is included in the flow equations as is the momentum exchange (drag) between them that limits their relative velocity. The expression for the momentum exchange is chosen such that the calculated flows satisfy the Wallis flooding relation in the quasi-steady limit. Note

that this is the result of a dynamic model for relative velocity and is not simply the imposition of a counter-current flow limit.

The same situation can occur when both fluids are represented by pool or both by atmosphere, but have different densities as a result of different temperatures and/or compositions). The latter (atmosphere) case is important in a number of situations including air ingress into a high temperature gas reactor (HTGR), natural circulation in the hot leg of a pressurized water reactor (PWR), and natural circulation in containment. MELCOR can still calculate a counter-current flow, but two calculational flow paths are required. If there is only physical connection between the volumes, it may be split into MELCOR flow paths, but the momentum exchange between the two flow paths must be modeled if reasonable flows are to be calculated.

This split-flow-path approach has often been used to model natural circulation in the hot leg of a PWR, with a user-defined control function "pump" in one or both of the paths to model the momentum exchange between the flows. The pump pressure is calculated as a function of the relative velocity. In early calculations, the value was based on an analogy to wall drag in simple pipe flow, which led to a form

$$\Delta P = -\frac{2 \cdot f \cdot L}{D_h} \bar{\rho} \cdot (v_1 - v_2) \cdot |v_1 - v_2| = -C \cdot \bar{\rho} \cdot (v_1 - v_2) \cdot |v_1 - v_2| \tag{6-93}$$

Here the variables are:

- ΔP = differential pressure [Pa]
- v_1 = velocity in the upper flow path [m/s]
- v_2 = velocity in the lower flow path [m/s]
- f = Fanning friction factor [-]
- D_h = hydraulic diameter [m]
- $\bar{\rho}$ = Average density of the fluids in the two flow paths [kg/m^3]
- L = length over which the drag force acts [m].

And

$$C \equiv \frac{2fL}{D_h} \tag{6-94}$$

is a dimensionless coefficient.

In recent calculations, the definition of ΔP has been modified. A suite of control functions uses a PID (Proportional, Integral, Differential) controller to drive the flows

towards values given by a Froude number correlation of the type described in following subsections.

We have added a general internal model to the MELCOR fluid dynamics package to deal with the momentum exchange in separated atmosphere flow. In simplest terms, it couples two flow paths through an exchange of momentum essentially similar to the control-function-pump model. Instead of requiring user definition of a "pump", the momentum exchange is chosen such that quasi-steady solutions match the published correlations of Epstein and Kenton [34]. Thus, like the pool/atmosphere flooding model, it is a dynamic model for relative velocity, rather than the simple imposition of a counter-current flow limit.

6.11.1 Epstein-Kenton Correlation for Stratified Flow

The model described in Epstein and Kenton is based on volumetric flows. It defines correlations for the volumetric flow in pure natural circulation, Qcc, where the net flow, Qnet, is zero, and for the net flow at "flooding", Qflood, where the reverse flow is zero in terms of the Froude number, Fr, defined by

$$Fr = \frac{Q}{\sqrt{\ell^5 g \Delta\rho / \bar{\rho}}} \tag{6-95}$$

Here the variables are

- Q = volumetric flow [m^3/s]
- ℓ = characteristic dimension [m]
- g = acceleration of gravity [m^2/s]
- $\Delta\rho$ = density difference [kg/m^3]
- $\bar{\rho}$ = average density [kg/m^3]

(In Epstein and Kenton, Q_{net} and Q_{flood} are denoted by Q_u and q, respectively.) In dimensional form, these take the form

$$Q_{flood} = X_{flood} C_D \sqrt{\frac{\ell^5 g \Delta\rho}{\bar{\rho}}} \tag{6-96}$$

$$Q_{cc} = X_{cc} C_D \sqrt{\frac{\ell^5 g \Delta\rho}{\bar{\rho}}} \tag{6-97}$$

The assumption is that the X coefficients are only functions of the orientation and geometry of the opening.

Within the range of countercurrent flow, there is a forward flow, Q_+, and a reverse flow, Q_-, and a further relation defines the reverse flow as

$$Q_- = Q_{cc}\left(1 - \frac{Q_{net}}{Q_{flood}}\right)^n \tag{6-98}$$

for $0 \le Q_{net} \le Q_{flood}$, where the net flow is given by

$$Q_{net} = Q_+ - Q_- \tag{6-99}$$

Given the actual geometry of the openings, these correlations can be recast in terms of flow per unit area

$$j \equiv Q/A \tag{6-100}$$

As

$$j_{flood} = x_{flood}\ C_D \sqrt{\frac{\ell g \Delta\rho}{\bar{\rho}}} \tag{6-101}$$

$$j_{cc} = x_{cc}\ C_D \sqrt{\frac{\ell g \Delta\rho}{\bar{\rho}}} \tag{6-102}$$

$$j_- = j_{cc}\left(1 - \frac{j_{net}}{j_{flood}}\right)^n \tag{6-103}$$

For horizontal flow through a rectangular opening, the characteristic length, ℓ, is the height of the opening, H, $x_{flood} = 0.9428$, $x_{cc} = 0.3333$, and $n = 1.5$.

For horizontal flow through a circular opening, the characteristic length, ℓ, is the diameter, D (which is also the height of the opening), $x_{flood} = 0.9549$, $x_{cc} = 0.3038$, and $n = 1.6$.

Reference 36 also contains a correlation of the same form for vertical flow through circular and rectangular conduits, for which the x coefficients are also functions of L/D where L is the length of the conduit.

More detail, including a discussion of the vertical flow correlation, can be found in Reference [35]

6.11.2 MELCOR Stratified Flow Model

This section describes the new model as implemented in MELCOR. It is based on the approach of coupling flows in two MELCOR flow paths that led to existing modeling of counter-current stratified flow in PWR hot legs. However, we have tried to generalize the approach and to eliminate unnecessary assumptions such as equality of flow areas. The implementation couples only gas ("atmosphere") flows. The coding is not limited to cases of pure gas flow, but is relatively untested for mixed-phase flow and should be used with caution for such flows.

6.11.2.1 Quasi-Steady Behavior of the Hydrodynamic Equations

Consider a case with gas flowing through two flow paths of area A_1 and A_2, and coupled by momentum exchange. Under quasi-steady conditions, the pressure differential driving circulation is balanced by the momentum exchange between the two flows, so that

$$\Delta P_{circ} = (\rho_{out} - \rho_{in}) g (z_1 - z_2) = 2\Delta P_{momex} = 2C(j_1, j_2)\bar{\rho}(j_1 - j_2)|j_1 - j_2| \quad (6\text{-}104)$$

Here, as a generalization of the model used for PWR hot leg circulation, we have expressed the momentum exchange in terms of volumetric fluxes based on the *combined* area of the two paths. (We later confirm empirically that this eliminates the need to require equal flow areas for the two paths.)

$$j_i \equiv \frac{A_i v_i}{A_{TOT}} \quad (6\text{-}105)$$

$$A_{TOT} \equiv A_1 + A_2 \quad (6\text{-}106)$$

We have also assumed that the coefficient C has an explicit dependence on volumetric fluxes (velocities). The magnitude of the relative volumetric flux is then given by

$$|j_1 - j_2| = \sqrt{\frac{\Delta z g \Delta\rho}{2C(j_1, j_2)\bar{\rho}}} \quad (6\text{-}107)$$

From this, at the flooding point (corresponding to $j_i = 0$ or $j_2 = 0$)

$$j_{flood} = \sqrt{\frac{\Delta z g \Delta\rho}{2C_{flood}\bar{\rho}}} \quad (6\text{-}108)$$

Where

$$C_{flood} \equiv C(j_f, 0) \equiv C(0, -j_f) \tag{6-109}$$

Consider the case of horizontal flow through a rectangular opening of height H, and assume that the two paths model the upper and lower portions of the opening so that $\Delta z = H/2$ (regardless of their relative areas). If Equation (6-109) is to match the corresponding Epstein-Kenton correlation (Equation (6-101) with the appropriate coefficient), we must have

$$j_{flood} = \sqrt{\frac{\Delta z g \Delta\rho}{2 C_{flood} \bar{\rho}}} = 0.9428\ C_D \sqrt{\frac{H g \Delta\rho}{\bar{\rho}}} \tag{6-110}$$

which requires that

$$C_{flood} = \frac{1}{4(0.9428\ C_D)^2} \tag{6-111}$$

It is easily seen from Equation (6-107) that assuming a constant C would lead to a constant relative volumetric flux in the counter-current flow regime. This would lead to too much circulation, which would result in a too rapid mixing in the case of air ingression in an HTGR. The problem is clear: the use of two independent flow paths, each of constant area, cannot capture the change in partition of the total area between the outflow and the inflow as the net flow changes. Physical intuition and the results of computational fluid dynamics calculations both show that as the net flow increases to the flooding point within a single conduit, the fraction of the total flow area occupied by the forward flow, Q'_+ actually increases and that occupied by the reverse flow Q'_- decreases.

However, we can compensate by adjusting the dependence of the momentum exchange coefficient. The analytic solution of the MELCOR equation for quasi static flow, Equation (6-104), takes the form

$$(j'_1 - j'_2)|j'_1 - j'_2| = \frac{C_{flood}}{C(j_1, j_2)} \tag{6-112}$$

where the prime denotes normalization of the volumetric flux to the flooding flux, j_{flood}. Through numerical experimentation, we have found that the form

$$C(j_1, j_2) = \frac{C_{flood}}{1 + E[(j'_1)^2 + (j'_2)^2 - 1]} \tag{6-113}$$

leads to good agreement with the Epstein-Kenton Correlations. If *C* is defined by Equation (6-113), Equation (6-112) takes the form of a quadratic for the relative volumetric flux in terms of the net volumetric flux with the solution

$$|j_1' - j_2'| = \sqrt{\frac{2 - 2E + E(j_1' + j_2')^2}{2 - E}} \tag{6-114}$$

If the net volumetric flux is known, this defines the individual volumetric fluxes in the counter-current region. The coefficient E can be chosen to match exactly one point in addition to the flooding point in the Epstein-Kenton correlation. If we chose to match the point of pure natural circulation, it is easy to show that

$$E = \frac{1 - 4F^2}{1 - 2F^2} \tag{6-115}$$

Where

$$F \equiv \frac{Q_{cc}}{Q_{flood}} = \frac{j_{cc}}{j_{flood}} = \frac{x_{cc}}{x_{flood}} \tag{6-116}$$

For horizontal flow through a rectangular opening, *F* = 0.3333 / 0.9428 = 0.3535, while for horizontal flow through a circular opening, *F* = 0.3038 / 0.9549 = 0.3181. Figure 6.2 shows the comparison of the Epstein-Kenton correlations for the rectangular and circular openings (solid lines) to the analytic solution of the MELCOR model for these openings (dashed lines). As shown in Figure 6.2, the agreement with the Epstein-Kenton correlations is quite reasonable. It could probably be improved by modifying the expression for the velocity dependence of the momentum exchange coefficient *C* in Equation (6-113), but we do not believe that the effort would be justified, at least for current applications.

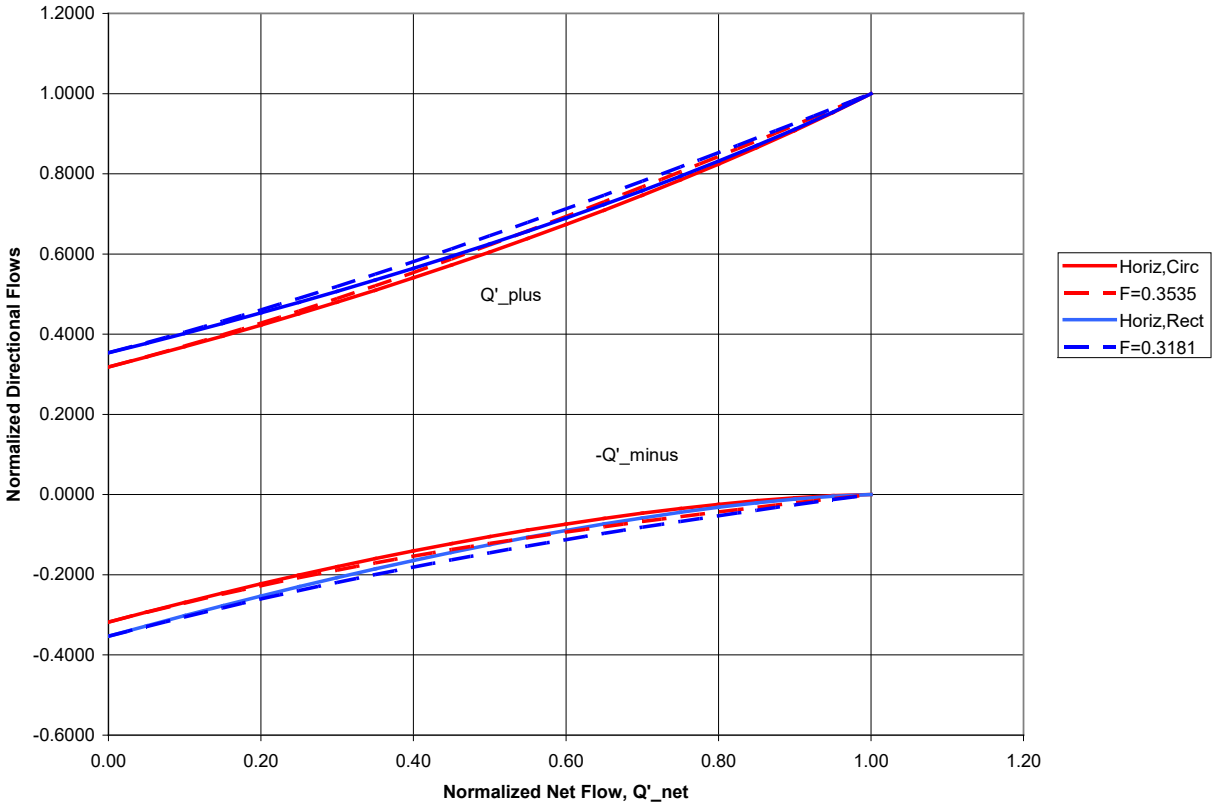


Figure 6.2 Normalized Flow versus Normalized Net Flow for the CCF Model

6.11.2.2 Numerical Implementation of Model

On examining the modeling approach described above, we see that the essential input consists of identifying the two paths to be coupled and providing the value of the coefficient in the definition of ΔP. If these are known, the exchange terms can be calculated and included in the flow equations when they are set up and solved. All that is necessary is to add a term to each of matrix elements that couple the two flow paths in the equation set. At least for now, we limit the coupling to gas ("atmosphere") flow, so that only four matrix elements are involved. If Equation (6-90) were used, the result would be

$$\begin{bmatrix} A_{11}+\dfrac{\Delta t}{\rho_1 L_1}\dfrac{A_1 C}{A_{tot}}\overline{\rho}\left|j_1-j_2\right| & A_{12}-\dfrac{\Delta t}{\rho_1 L_1}\dfrac{A_2 C}{A_{tot}}\overline{\rho}\left|j_1-j_2\right| \\ A_{21}-\dfrac{\Delta t}{\rho_2 L_2}\dfrac{A_1 C}{A_{tot}}\overline{\rho}\left|j_1-j_2\right| & A_{22}+\dfrac{\Delta t}{\rho_2 L_2}\dfrac{A_2 C}{A_{tot}}\overline{\rho}\left|j_1-j_2\right| \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}=\begin{bmatrix} RHS_1 \\ RHS_2 \end{bmatrix}$$

Here A_{ij} and RHS_i are the matrix and right-hand-side vector elements representing all *other* phenomena, and only the part of the equation set involving the coupled flows is

shown. Solution of the flow equations is iterative and, as with the friction modeling, previous-iterate values of velocities are used in calculating velocity-dependent terms in the matrix elements, in this case, $C(j_1 - j_2)$ and $|j_1 - j_2|$.

The control-function-pump model is numerically explicit in the sense that it is based on start-of-step velocities, and typically requires short timesteps and smoothing (relaxation) of ΔP values for stability. In contrast, the coupling in the new model is numerically implicit, and there should be no such stability problems.

6.11.2.3 Extension to Two-Phase Flow

As mentioned earlier, although only the atmosphere fields are coupled by the model, the implementation is not limited to cases of pure atmosphere flow. As coded, the areas used in the model are the areas available to gas ("atmosphere") flow. That is the areas used are defined by

$$A_i \Rightarrow F_i \alpha_i A_i$$

where the variables are those defined in other sections of this manual.

F_i = open fraction [-]
α_i = atmosphere fraction [-]
A_i = nominal area [m^2]

6.11.2.4 Discussions

The CCF model described in this reference section has been implemented into MELCOR. See CVH package Users' Guide for the input format and description to use this model. We recommend that users may want to adjust the input parameters via the CFD code comparison or other calculation methods in order to represent the countercurrent gas flow realistically.

6.12 Homologous Pump Model

The new MELCOR homologous pump model is similar to that of RELAP [36] but with some distinguishing features including a polar homologous pump curve representation and a "universal correlation" as discussed in a later section. Several new MELGEN input records have been added that, in general, allow the user to fully specify:

- Rated pump conditions,
- Single/two-phase pump performance via homologous curve input,
- Pump friction torque as a polynomial in pump speed ω, or rather the non-dimensional ratio $|\omega/\omega\_R|$ with ω_R the rated pump speed,
- Pump inertia as a polynomial in the quantity $|\omega/\omega\_R|$,
- Pump speed and motor torque controls,

- Pump trips,
- Pump numerical treatment options

Additionally, pump data from both the Semiscale [37] and Loft [38] experiments are available as “built-in” performance modeling options. A “universal correlation” [39] is included. These features are described below.

6.12.1 The Homologous Pump Representation

The essential goal of the homologous pump model is to characterize centrifugal pump performance by predicting the pressure head (ΔP) and the hydraulic torque (τ_H) given inputs of impeller speed (i.e. pump speed) ω and pump capacity (volumetric flow rate) Q. Empirical homologous curves are one way of compactly summarizing pump performance in response to given conditions ω, Q. In this representation, non-dimensional ratios α and v are formed as:

$$\alpha = \frac{\omega}{\omega_R} \; and \; \nu = \frac{Q}{Q_R}, for \; rated \; speed \; and \; capacity \; \omega_R, Q_R \qquad (6\text{-}117)$$

Then, homologous ratios α/v and v/α are formed from the non-dimensional ratios. The entire domain of pump operation (a total of 4 modes, described later) is covered by allowing:

$$\left|\frac{\nu}{\alpha}\right| \leq 1, called \; the \; \text{"α range" and} \; \left|\frac{\alpha}{\nu}\right| < 1, \text{called the "}\nu \; range\text{"} \qquad (6\text{-}118)$$

The idea is that these homologous ratios become the independent variables for the homologous-curve-dependent variables that include single-phase pump head, single-phase hydraulic torque, fully-degraded pump head, and fully-degraded hydraulic torque. For each dependent variable, a plot can be created that traces out 4 possible operating mode curves, each with 2 pieces (called octants): (1) α range and (2) v range. Thus, a homologous curve for a given dependent variable consists of 8 octants (connecting line segments) that each represent different combinations of pump speed (>= 0 or < 0) and capacity (>=0 or <0). When ω and Q are such that $|\nu/\alpha|$ is less than or equal to 1, this indicates pump operation in the v range of the current mode, and as soon as v grows large enough that $|\nu/\alpha|$ would exceed 1, the independent variable is changed to $|\alpha/\nu|$ such that pump operation enters the α range of the current mode. Thus the magnitude of the independent variable is never greater than 1 and is bounded by [-1,1]. Note that α and v ranges deal exclusively with non-dimensional pump speed/capacity magnitudes and their ratios. The pump modes account for the 4 possible combinations of negative/positive speed and negative/positive capacity. The modes are summarized in Table 6.2:

Table 6.2 Possible pump operating modes

Plot Quadrant	Nondim. Speed/Capacity	Mode Identifier
1st (Upper Right)	α>= 0 and v >= 0	Normal (N)
2nd (Upper Left)	α> 0 and v < 0	Dissipation (D)
3rd (Lower Left)	α<= 0 and v <= 0	Turbine (T)
4th (Lower Right)	α< 0 and v > 0	Reversal (R)

Note that the possibility of negative speed and negative capacity requires the independent variable to cross the zero value. The dependent variable also changes depending on the range (α or v) in question. In the homologous formulation, non-dimensional values for pump head and hydraulic torque are:

$$h = \frac{H}{H_R}\ and$$
$$t = \frac{\tau_H}{\tau_{H,R}}, for\ rated\ head\ H_R\ and\ hydraulic\ torque\ \tau_{H,R} \quad (6\text{-}119)$$

Furthermore, these quantities are divided by either α^2 or v^2 depending upon the range (α or v). The final form for a homologous curve is shown for Figure 6.3 through Figure 6.7 below. Each was adapted from a source ([36], [37], [38], [40). Note that a system of three-letter octant identifiers is used.

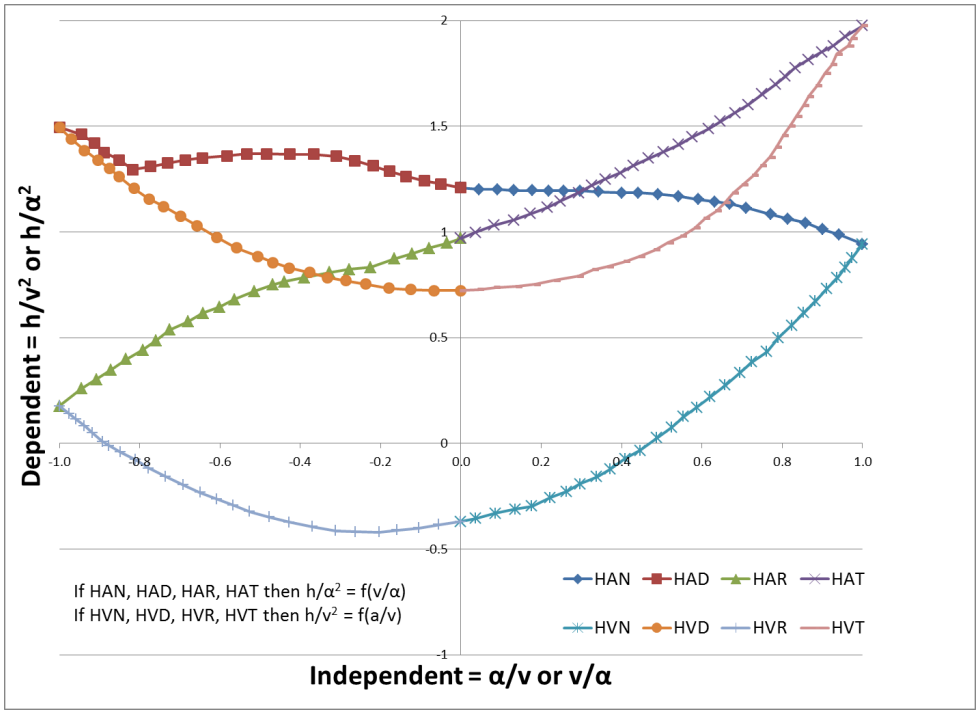


Figure 6.3 Semiscale single-phase head curve

Each octant identifier indicates at once the dependent variable, the independent variable, and the pump mode. The general identifier is [H/B][A/V][N/D/T/R], where the first square bracket pair denotes head (H)/torque (B), the second square bracket pair denotes α range (A) or v range (V), and the third square bracket pair denotes normal (N)/dissipation (D)/turbine (T)/reversal (R).

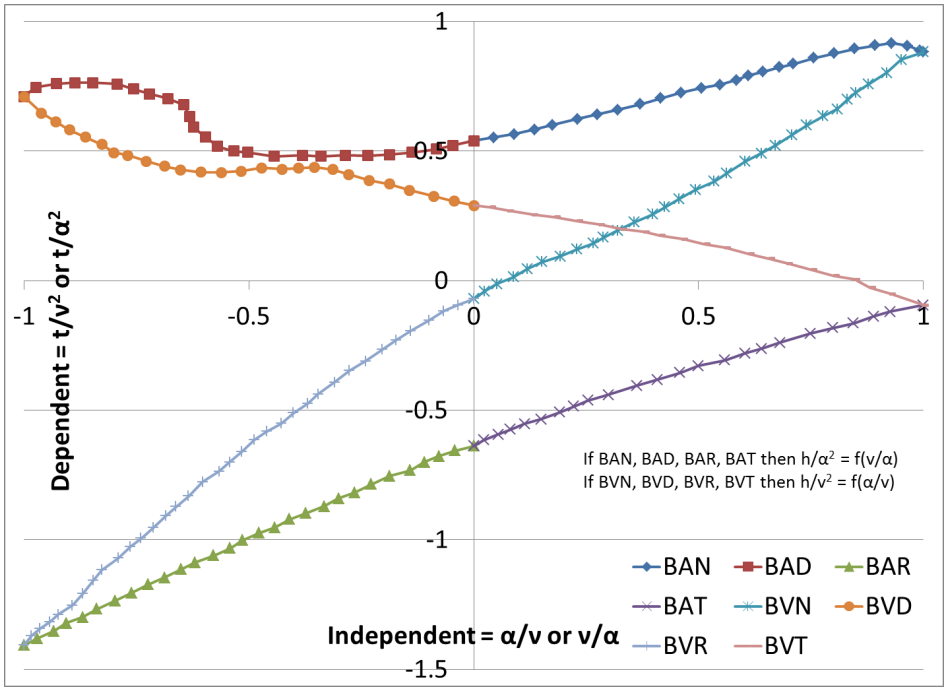


Figure 6.4 Semiscale single-phase hydraulic torque curve

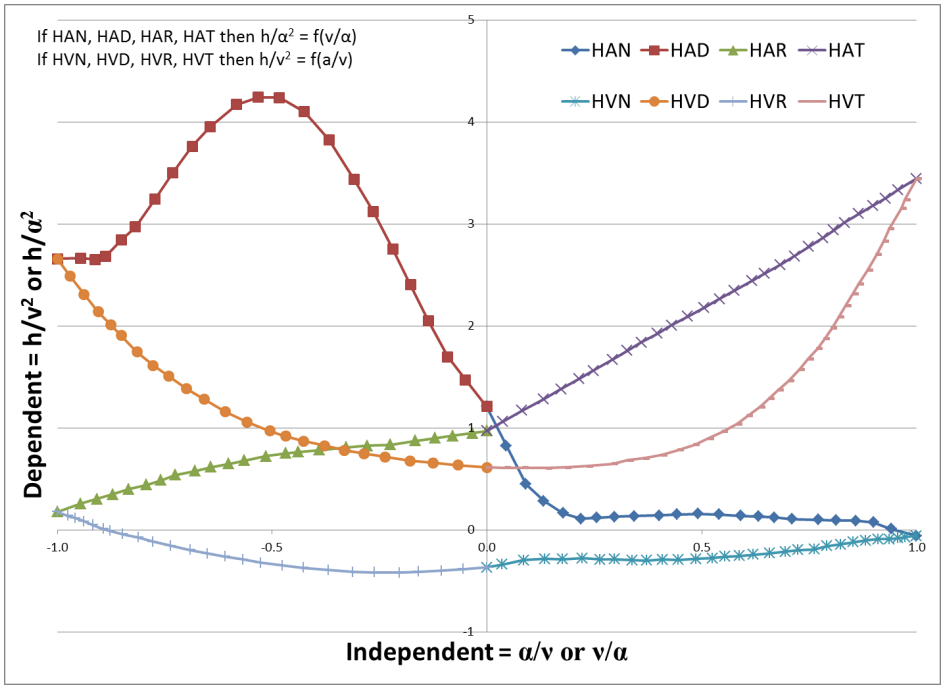


Figure 6.5 Semiscale fully-degraded head curve

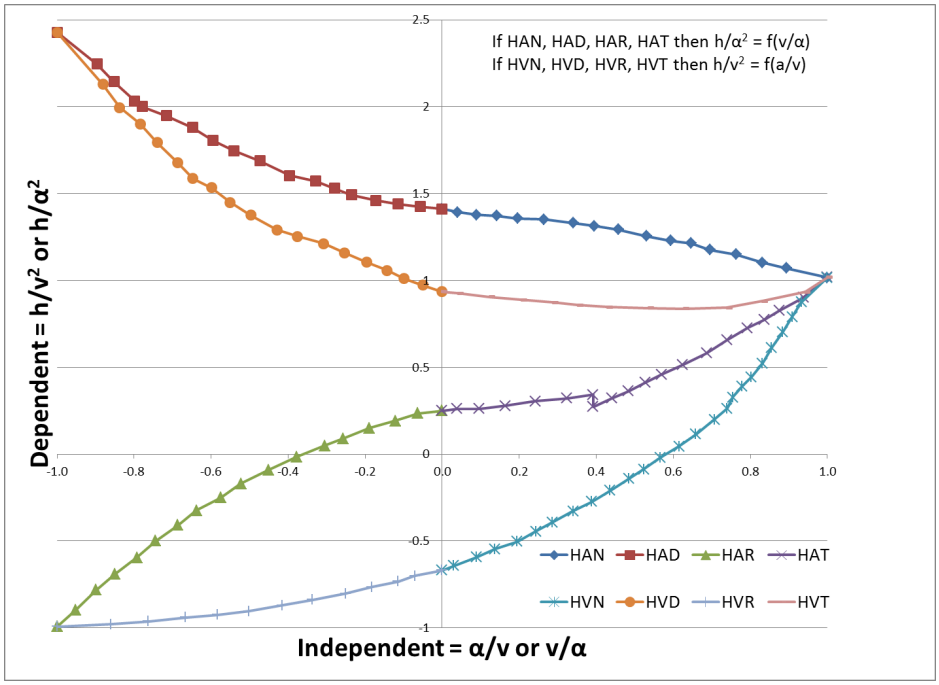


Figure 6.6 LOFT single-phase head curve

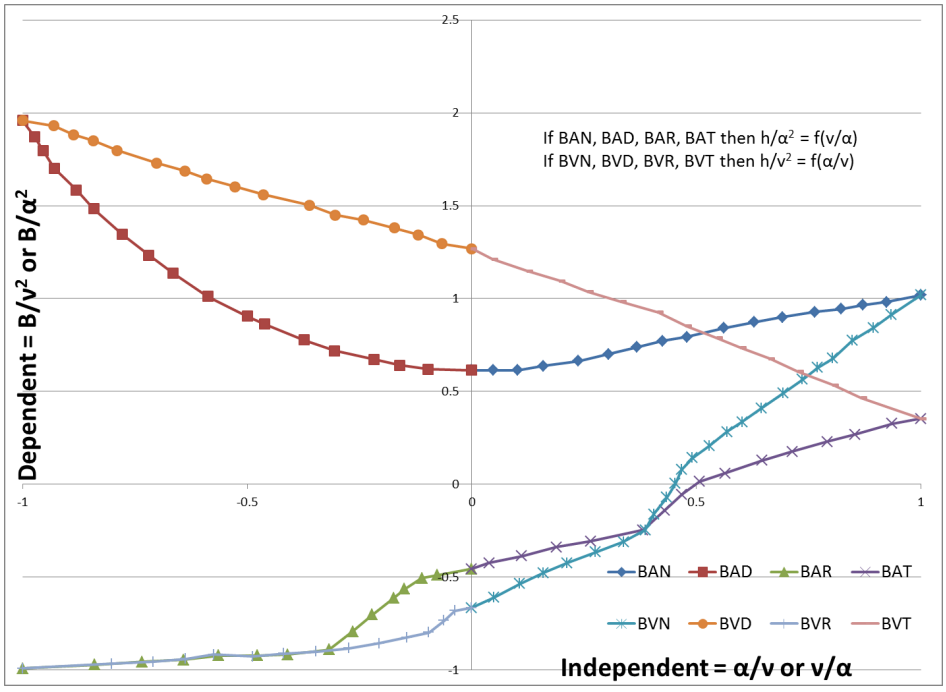


Figure 6.7 LOFT single-phase hydraulic torque curve

6.12.2 The Polar Homologous Representation

For several reasons related to programming convenience, MELCOR internally employs the polar homologous representation [41] of pump performance. It bears some resemblance to the conventional homologous method, but uses different variable definitions such that the independent variable is always positive and bounded on [0,2π]. As an added bonus, the variable transformation allows all octants to be ordered in monotonically-increasing fashion with respect to a single independent variable. This obviously simplifies data interpolation logic, since with the homologous data representation a total of 8 separate data tables (one per octant) are used. Essentially, the polar homologous variable transformation results in one independent variable and one dependent variable (either a head and torque function) that truly is a mathematical "function" because there is no more than 1 value of the dependent variable for a given value of the independent variable.

The polar homologous representation is identical to the homologous representation until the point where independent and dependent variables are chosen. The new independent variable is:

$$x = C + \tan^{-1}\left(\frac{\alpha}{\nu}\right), for\ \alpha\ and\ \nu\ as\ defined\ before \tag{6-120}$$

The constant C assumes different values depending upon the mode of pump operation. Regardless of the negative/positive sign of speed or capacity, the argument of the inverse tangent function does not change. Some useful properties of the inverse tangent function are exploited: it is defined for an argument equal to zero and it is bounded as an argument goes to infinity on either side of zero, approaching a limit of +/- π/2. One could foresee a problem if capacity and hence v equals zero, but in fact the division by zero can simply be treated as a case of the argument α/v approaching +/- ∞. The homologous octants, under this definition of x, are arranged in a predictable way on [0,2π] as summarized in Table 6.3 below.

Table 6.3 Octant arrangement under the polar homologous representation

Octant Identifier (C value)	Portion of domain on [0,2π]
[H/B]VN (0)	[0,π/4]
[H/B]AN (0)	[π/4,π/2]
[H/B]AD (π)	[π/2,3π/4]
[H/B]VD (π)	[3π/4,π]
[H/B]VT (π)	[π,5π/4]
[H/B]AT (π)	[5π/4,3π/2]
[H/B]AR (2π)	[3π/2,7π/4]
[H/B]VR (2π)	[7π/4,2π]

The dependent variables (head and torque functions, WH and WT) are:

$$WH = \frac{\left(\frac{h}{\alpha^2} \mid \frac{h}{\nu^2}\right)}{1 + \left(\left(\frac{\nu}{\alpha}\right)^2 \mid \left(\frac{\alpha}{\nu}\right)^2\right)} \quad and \quad WT = \frac{\left(\frac{t}{\alpha^2} \mid \frac{t}{\nu^2}\right)}{1 + \left(\left(\frac{\nu}{\alpha}\right)^2 \mid \left(\frac{\alpha}{\nu}\right)^2\right)} \qquad (6\text{-}121)$$

Where the (|) denotes a decision where the left hand side of | (in all parenthetical terms) is taken for the α range whereas the right hand side is taken for the v range. One observes that WH, for example, is readily obtained from data pairs corresponding to any given octant, e.g. HAN where the homologous data pairs are (v/α, h/α^2). Thus, there is no need for extra programming logic as the data pairs of each octant can be substituted into the WH and WT equations directly. After converting independent and dependent variables into polar homologous form, new plots are recovered as shown in Figure 6.8 through Figure 6.13 below. These figures contain all the same information as Figure 6.3 through Figure 6.7 but are more convenient for lookups and interpolation. Note the user still inputs homologous data in conventional form such that, in general, 32 tabular functions (TF) are required to fully characterize pump performance (8 octants for single-phase head, 8 for single-phase torque, 8 for fully-degraded head, 8 for fully-degraded torque). The conversion of user-specified pump data to polar homologous form is transparent to the user (handled internally by MELCOR).

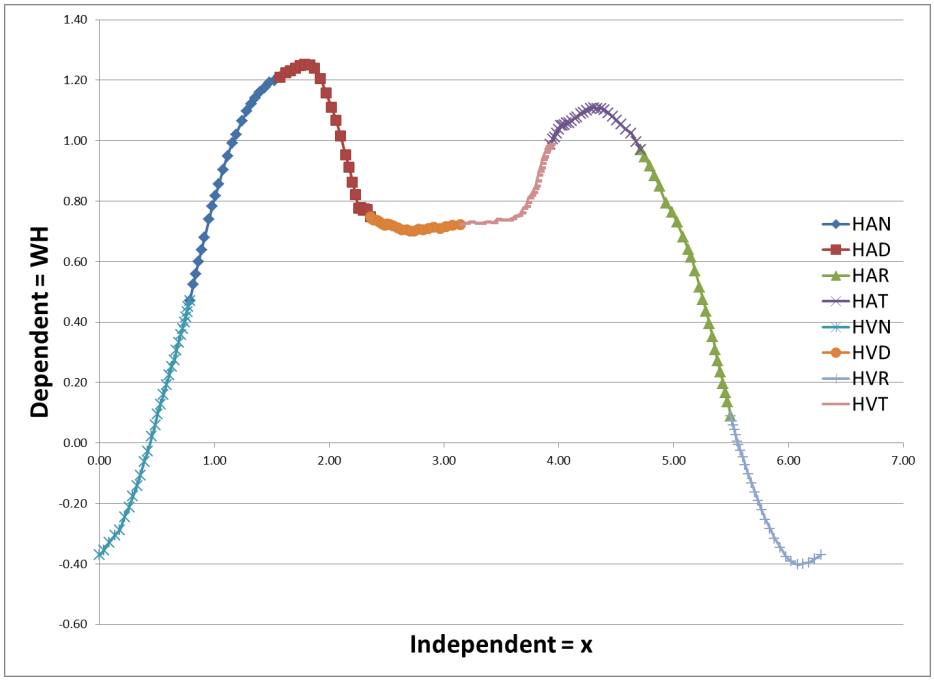


Figure 6.8 Polar homologous single-phase head curve, Semiscale

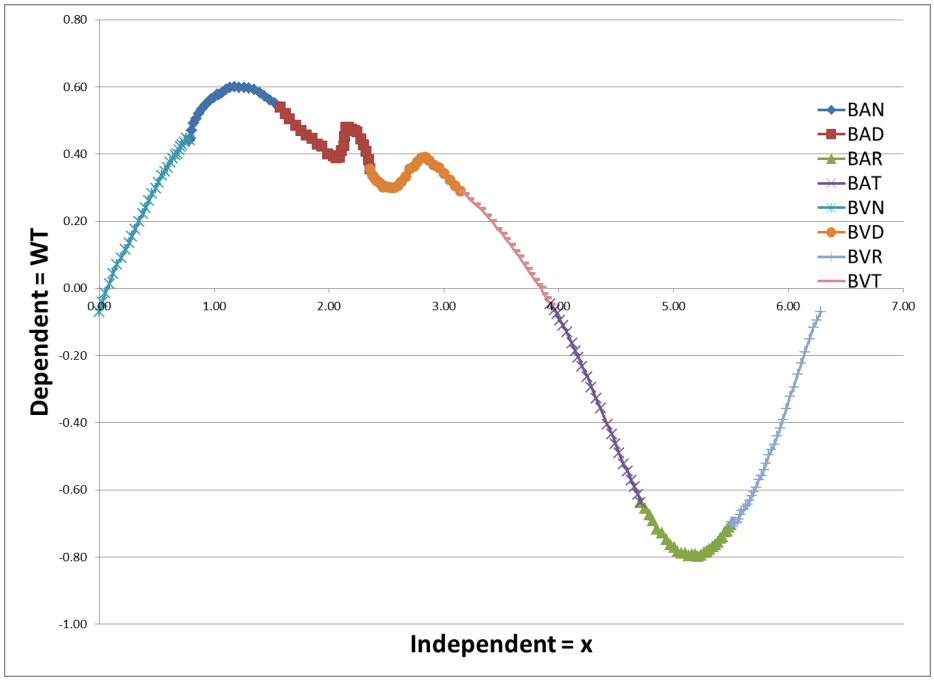


Figure 6.9 Polar homologous single-phase hydraulic torque curve, Semiscale

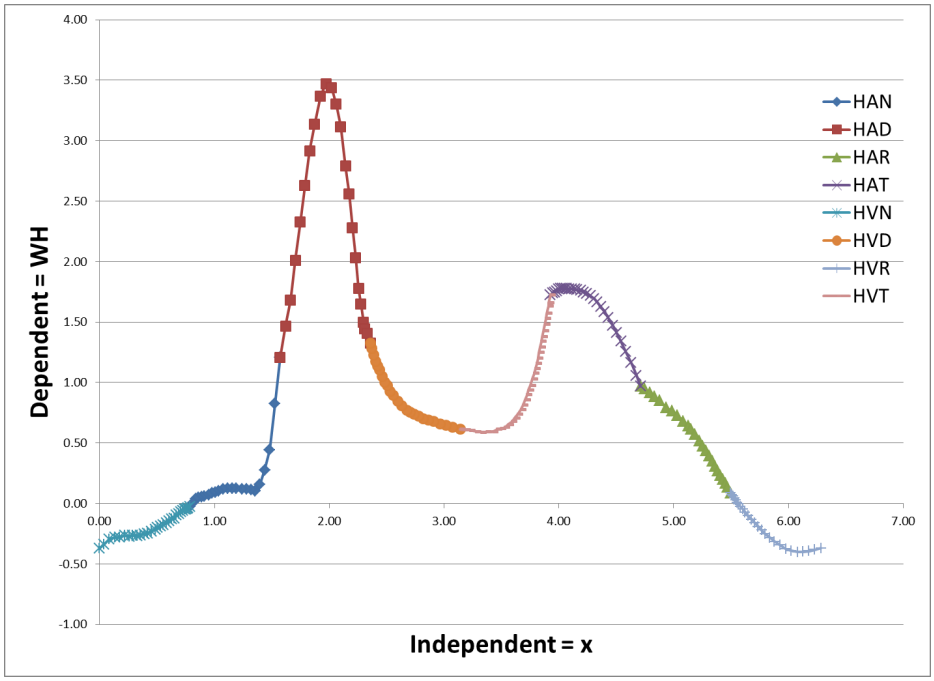


Figure 6.10 Polar homologous fully-degraded head curvew, Semiscale

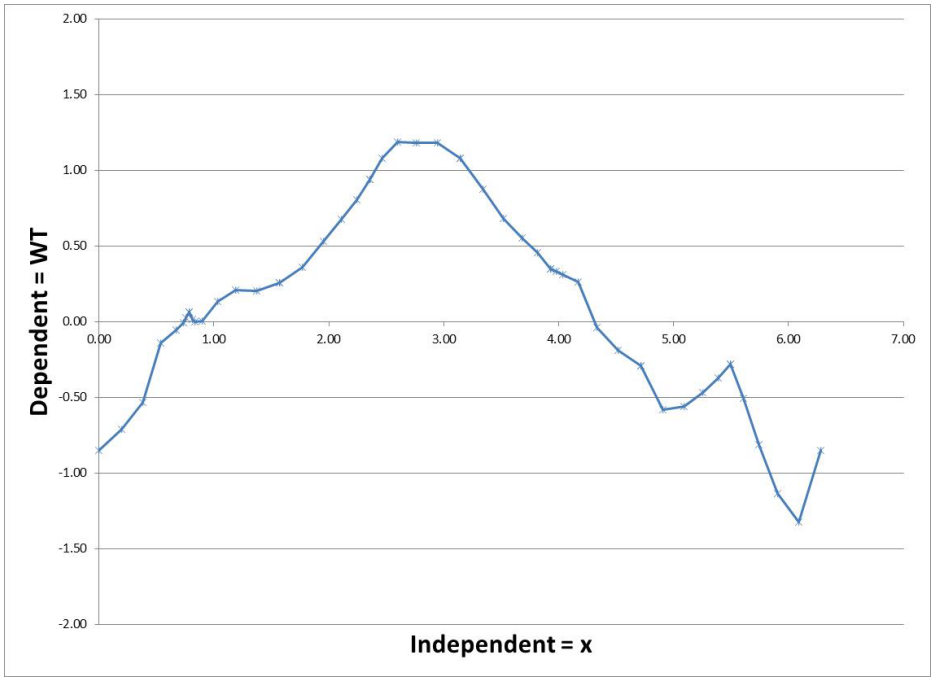


Figure 6.11 Polar homologous fully-degraded hydraulic torque curve

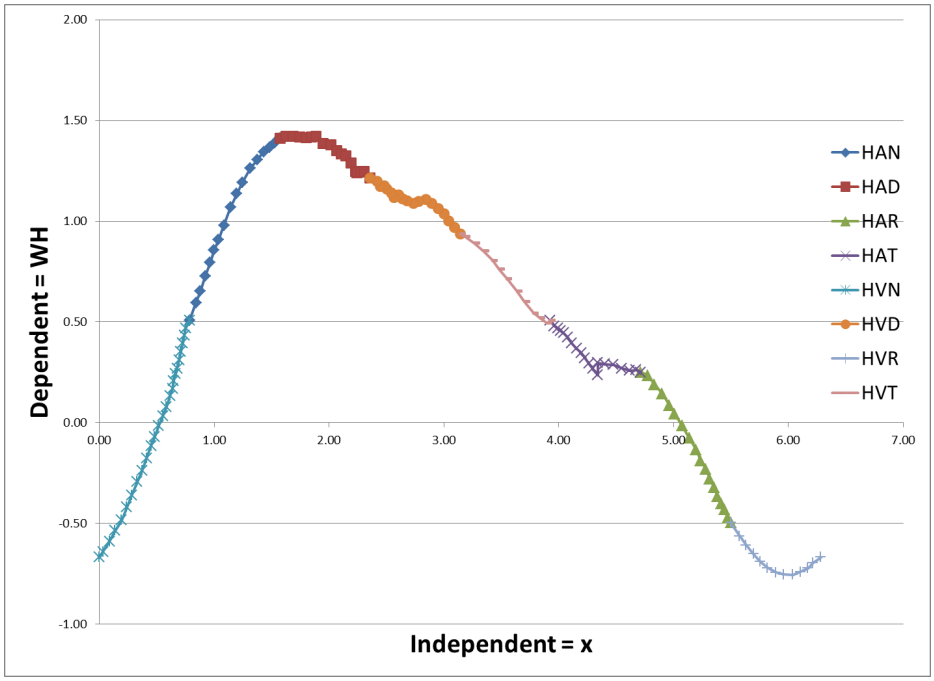


Figure 6.12 Polar homologous single-phase head curve, LOFT

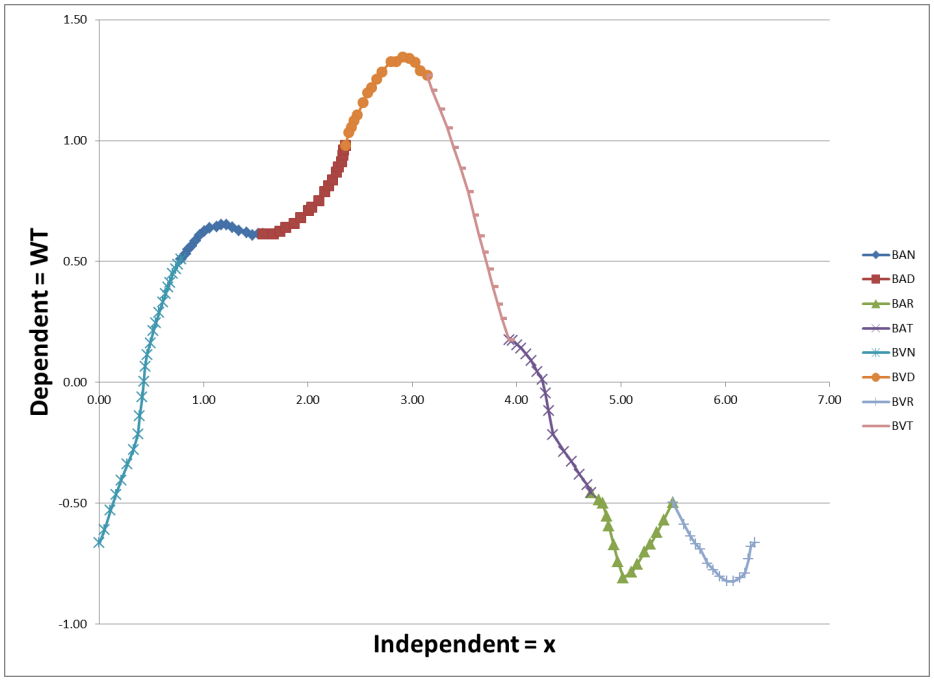


Figure 6.13 Polar homologous single-phase hydraulic torque curve, LOFT

6.12.3 Homologous Curve Consistency

Considering both homologous and polar homologous plots from figures above, there are obviously some requirements as to what octant curves must meet where. Notice that neither the homologous nor the polar homologous curves are disjointed or discontinuous where octants meet at certain values of the independent variable. These features reflect the physical operation of the pump, since octant/mode transitions correspond to changes in speed and capacity. Data input by the user must be checked for consistency, and Table 6.4 summarizes the eight separate conditions that must be satisfied.

Table 6.4 Consistency conditions on homologous pump data

Check Number	Homologous Form (ind = indvar)	Polar Homologous Form
1	[H/B]AN = [H/B]AD at ind = 0.0	Octants match at $\pi/2$
2	[H/B]AN = [H/B]VN at ind = 1.0	Octants match at $\pi/4$
3	[H/B]VR = [H/B]VN at ind = 0.0	Octants match at 0 and 2π
4	[H/B]AR = [H/B]VR at ind = -1.0	Octants match at $7\pi/4$
5	[H/B]AT = [H/B]AR at ind = 0.0	Octants match at $3\pi/2$
6	[H/B]VT = [H/B]AT at ind =1.0	Octants match at $5\pi/4$
7	[H/B]VT = [H/B]VD at ind = 0.0	Octants match at π
8	[H/B]VD = [H/B]AD at ind = -1.0	Octants match at $3\pi/4$

6.12.4 Universal Correlation

Based upon several data sets, head and hydraulic torque functions were derived as a function of pump specific speed for a given set of pump conditions x(ω,Q) [39]. These functions are only valid in the normal operating mode (x on [0,π/2]). Pump specific speed is defined, in this context, as a dimensional quantity:

$$N_s = \frac{\omega_R \sqrt{Q_R}}{{H_R}^{3/4}} \qquad (6\text{-}122)$$

Given specific speed and given pump conditions ω and Q, a double interpolation may be performed on independent variable x and dependent variable WH (WT) to recover an estimate of WH (WT) from which a dimensional head (hydraulic torque) can be recovered. Outside the normal mode and/or for two-phase considerations, pump performance modeling must default to built-in or user-supplied data. The actual code

implementation follows the method in [39] so that each curve of Figure 6.14 and Figure 6.15 below is fitted with a 3rd order polynomial f(Ns). The interpolation is accomplished with these curve fits rather than with raw tabular pump data. Arrays of curve-fit data are included in Table 6.5 and Table 6.6 below. The constants A through D represent polynomial coefficients as in:

$$(WH|WT) = A + BN_s + CN_s^2 + DN_s^3 \qquad (6\text{-}123)$$

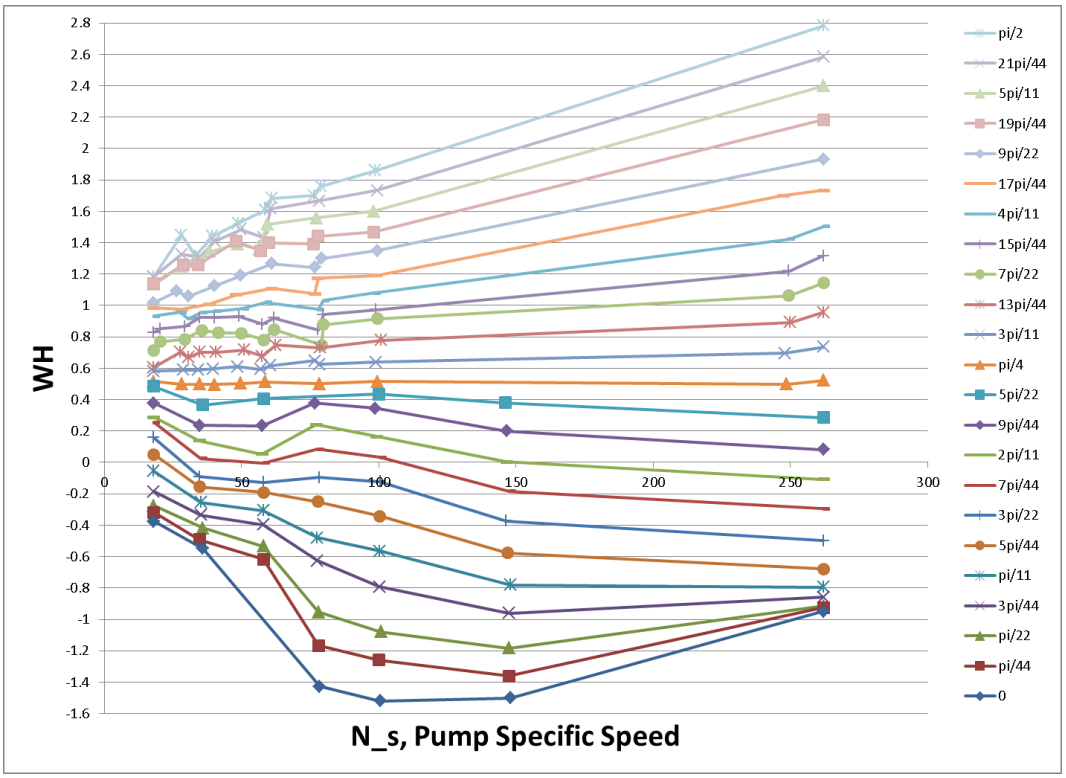


Figure 6.14 Universal correlation head function, each curve a different x value on [0,π/2] (mode N)

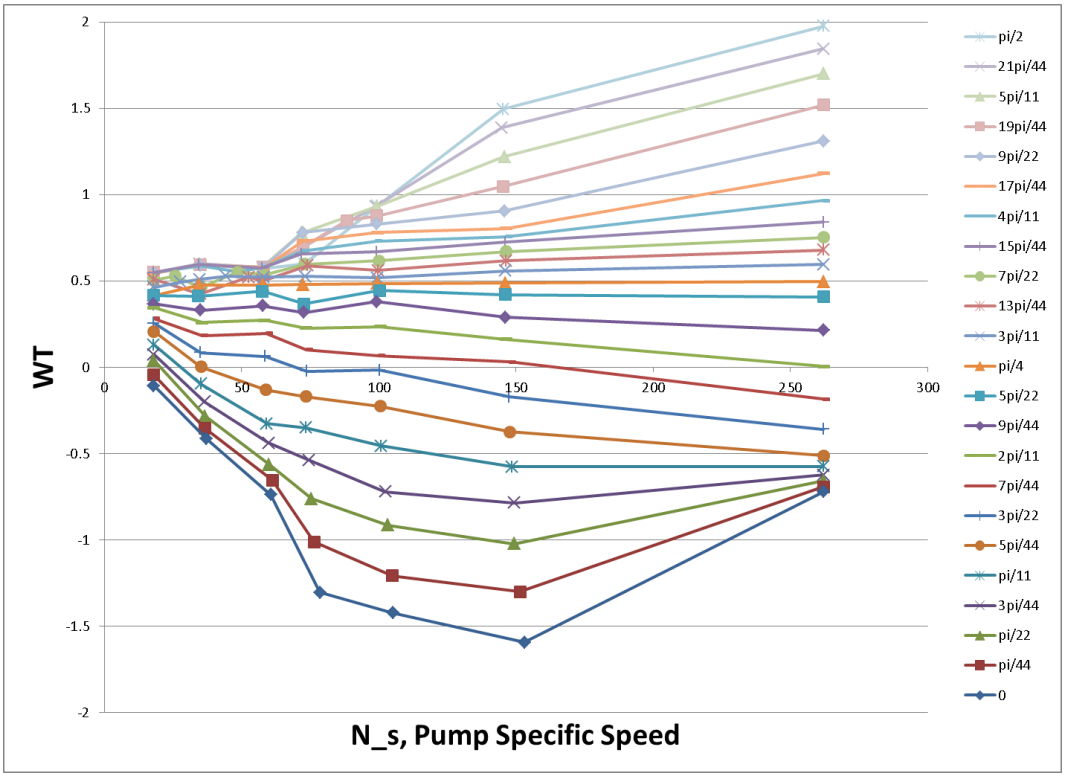


Figure 6.15 Universal correlation torque function, each curve a different x value on [0,π/2] (mode N)

Table 6.5 Array of polynomial-fit data to Figure 6.14 curves

x	A	B	C	D
pi/2	1.0946	0.0088	-9.5600E-06	0.00
21pi/44	1.0855	0.0077	-8.2443E-06	0.00
5pi/11	1.0813	0.0063	-5.3857E-06	0.00
19pi/44	1.0486	0.0049	-2.6536E-06	0.00
9pi/22	0.9948	0.0038	-1.0063E-06	0.00
17pi/44	0.9508	0.0021	2.7905E-06	0.00
4pi/11	0.9048	0.0011	4.1812E-06	0.00
15pi/44	0.8508	0.0007	3.4325E-06	0.00
7pi/22	0.7454	0.0012	7.6247E-07	0.00
13pi/44	0.6240	0.0017	-2.1960E-06	0.00
3pi/11	0.5494	0.0012	-1.9922E-06	0.00
pi/4	0.5000	0.0000	0.0000E+00	0.00

x	A	B	C	D
5pi/22	0.4515	-0.0062	0.0000E+00	0.0000E+00
9pi/44	0.3630	-0.0010	0.0000E+00	0.0000E+00
2pi/11	0.2595	-0.0021	7.1163E-06	-1.7401E-08
7pi/44	0.2138	-0.0025	-5.8728E-07	1.0494E-08
3pi/22	0.1375	-0.0038	2.6036E-06	9.2164E-09
5pi/44	0.1218	-0.0063	1.2390E-05	0.0000E+00
pi/11	0.0200	-0.0065	-2.8421E-07	4.9910E-08
3pi/44	-0.1975	0.0022	-1.7558E-04	1.1621E-06
pi/22	0.0130	-0.0143	3.8180E-05	1.0995E-08
pi/44	0.0541	-0.0184	5.7827E-05	-8.7150E-09
0	0.2704	-0.0346	2.0041E-04	-3.2951E-07

Table 6.6 Array of polynomial-fit data to Figure 6.15 curves

x	A	B	C	D
pi/2	0.3014	0.0068	-1.00E-06	0.00E+00
21pi/44	0.3385	0.0065	-3.00E-06	0.00E+00
5pi/11	0.3909	0.0056	-2.00E-06	0.00E+00
19pi/44	0.4370	0.0041	2.00E-07	0.00E+00
9pi/22	0.4532	0.0044	-1.00E-05	3.00E-08
17pi/44	0.4577	0.0044	-2.00E-05	5.00E-08
4pi/11	0.4992	0.0021	-1.00E-06	0.00E+00
15pi/44	0.5235	0.0016	-1.00E-06	0.00E+00
7pi/22	0.4573	0.0018	-3.00E-06	0.00E+00
13pi/44	0.4382	0.0016	-3.00E-06	0.00E+00
3pi/11	0.4750	0.0007	-1.00E-06	0.00E+00
pi/4	0.4294	0.0008	-2.00E-06	0.00E+00

x	A	B	C	D
5pi/22	0.4089	0.0002	-6.00E-07	0.00E+00
9pi/44	0.3540	0.0000	-2.00E-06	0.00E+00
2pi/11	0.3371	-0.0012	-3.00E-07	0.00E+00
7pi/44	0.3522	-0.0048	3.00E-05	-6.00E-08
3pi/22	0.2704	-0.0039	6.00E-06	0.00E+00
5pi/44	0.3547	-0.0108	6.00E-05	-1.00E-07
pi/11	0.3614	-0.0153	8.00E-05	-1.00E-07
3pi/44	0.3814	-0.0190	1.00E-04	-1.00E-07
pi/22	0.4123	-0.0227	1.00E-04	-1.00E-07
pi/44	0.3134	-0.0210	7.00E-05	0.00E+00
0	0.3756	-0.0257	8.00E-05	0.00E+00

Thus a user with knowledge only of rated conditions (implying N_s) for a given pump may predict its performance for the normal operating mode without defaulting to Semiscale or LOFT data. Semiscale and LOFT data are included in the universal correlation, so a calculation using the universal method with a specific pump speed equal to that of the Semiscale/LOFT pump yields results similar (in the N regime) to a calculation with built-in Semiscale/LOFT data.

6.12.5 Head and Hydraulic Torque Computation, Two-Phase Effects

The speed and flow conditions are used to deduce a value of the polar homologous independent variable x that can then be used to interpolate the polar homologous functions WH and WT. In general, two-phase effects must also be accounted for and x may be used to interpolate WH2 and WT2. Note WH2 and WT2 are just the two-phase (fully-degraded) head and torque functions obtained, for example, by interpolating the data of Figure 6.10 and Figure 6.11, respectively. The void fraction (or in the case of a MELCOR flow path, the atmosphere area fraction) can be used to interpolate the so-called head and torque degradation multipliers MH and MT. These functions of void fraction are plotted for Semiscale in Figure 6.16 and Figure 6.17 below. Note that the LOFT report provides degradation multipliers as well. This information is used to compute overall head and torque functions:

$$WH_{overall} = WH - MH * (WH - WH2) \quad (6\text{-}124)$$

$$WT_{overall} = WT - MT * (WT - WT2) \quad (6\text{-}125)$$

Using definitions already presented, the non-dimensional head and hydraulic torque are obtained from these overall functions. The dimensional head and hydraulic torque is then:

$$H = hH_R \text{ , and } \tau_H = t\tau_{H,R} \quad \text{(6-126)}$$

$$for\ h\ and\ t\ the\ nondimensional\ head\ and\ hydraulic\ torque$$

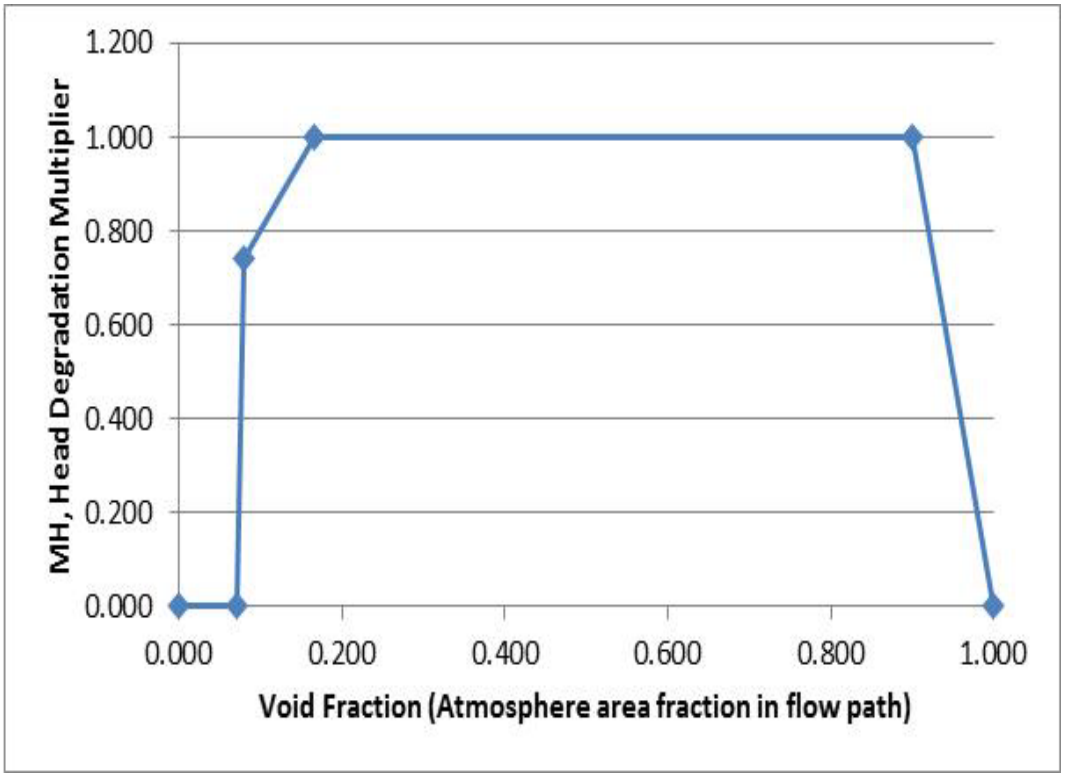


Figure 6.16 Semiscale head degradation multiplier

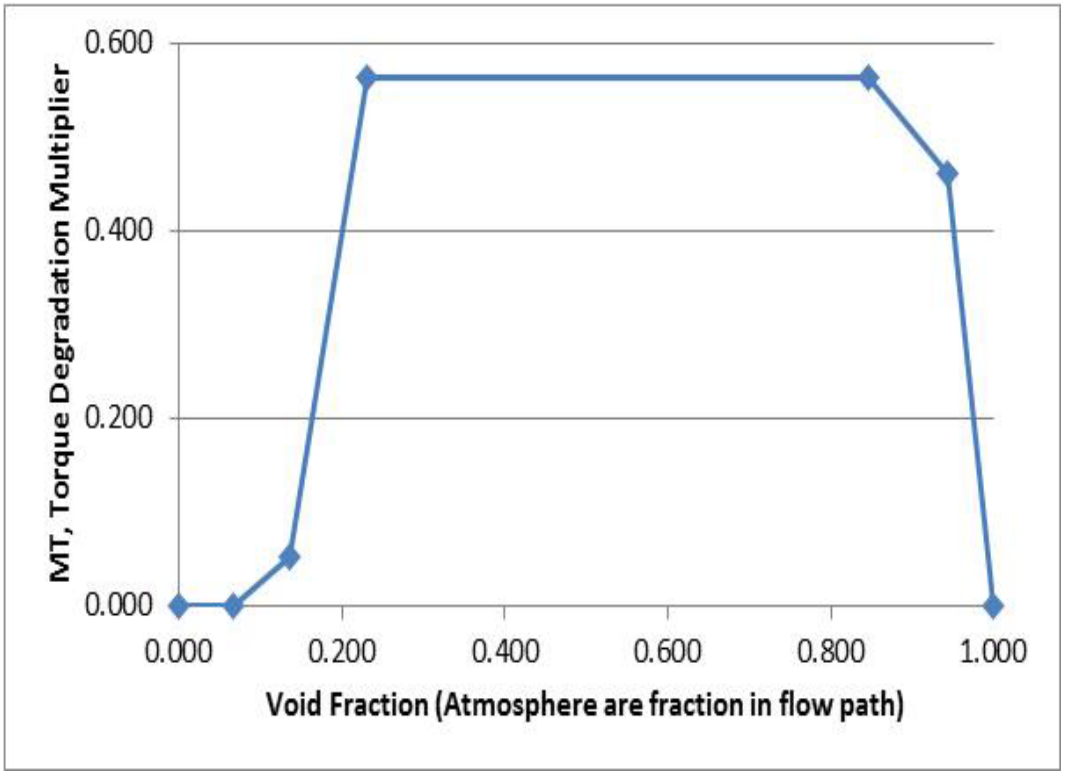


Figure 6.17 Semiscale torque degradation multiplier

If the user chooses an explicit formulation, the pump pressure head is included as a momentum source term in the flow path phasic velocity equations (same behavior as FANA and QUICK-CF pump types in the FL package). The user may also request a semi-implicit treatment for which pump head is expanded into implicit and explicit terms such that new-time phasic velocities do in part depend on new-time pump head terms. The details of this implementation are included in Appendix D.

6.12.6 Pump Friction Torque

The frictional torque associated with a pump is modeled as in RELAP [36] with a polynomial in the quantity $|\omega/\omega\_R|$, according to:

$$\tau_{fr} = \pm \tau_{frn} \, , for \left|\frac{\omega}{\omega_R}\right| < S_{PF} \qquad (6\text{-}127)$$

$$\tau_{fr} = \pm \left(\tau_{fr0} + \tau_{fr1} \left| \frac{\omega}{\omega_R} \right|^{x1} + \tau_{fr2} \left| \frac{\omega}{\omega_R} \right|^{x2} + \tau_{fr3} \left| \frac{\omega}{\omega_R} \right|^{x3} \right), for \left| \frac{\omega}{\omega_R} \right| \geq S_{PF} \quad (6\text{-}128)$$

The user specifies each of the constants and exponents in the above equations as well as the critical speed ratio S_{PF} below which the frictional torque is constant. The sign convention is that frictional torque is negative for a positive pump speed but positive for a negative pump speed. The exponents x1, x2, and x3 cannot equal 0 but there are no restrictions on the torque coefficients.

6.12.7 Pump Inertia

The pump inertia may be variable in certain situations, so a model similar to that of frictional torque [36] is applied. A 3rd order polynomial is written as:

$$I_p = I_{pn}, for \left| \frac{\omega}{\omega_R} \right| < S_{PI} \quad (6\text{-}129)$$

$$I_P = \left(I_{P0} + I_{P1} \left| \frac{\omega}{\omega_R} \right| + I_{P2} \left| \frac{\omega}{\omega_R} \right|^2 + I_{P3} \left| \frac{\omega}{\omega_R} \right|^3 \right), for \left| \frac{\omega}{\omega_R} \right| \geq S_{PI} \quad (6\text{-}130)$$

The user may specify each polynomial coefficient but not the exponents. The pump inertia is a positive nonzero quantity and, as with friction torque, may be held constant for pump speed below a threshold.

6.12.8 Pump Speed, Motor Torque, Trips

Pump speed is controlled either by the user via control function (CF) or TF or by the code via solution of a torque-inertia equation. The user has three options for speed control: (1) the pump speed can always be specified by CF/TF, (2) the pump speed can always be obtained by torque-inertia equation solution, or (3) the pump speed can be under CF/TF control until a trip (e.g. TRIP-type CF) indicates switchover to torque-inertia equation solution. Option 1 would be appropriate if, for example, a turbine-driven pump with speed determined by a turbine equation (presumably given in separate CFs) was being modeled. Option 3 would be appropriate for modeling a pump coast-down during off-normal operating conditions after some time of normal pump operation. Option 2 allows the net torque (motor torque less the sum of hydraulic and friction torque) to determine whether pump speed increases (net positive torque), decreases (net negative torque), or stays constant (net zero torque, motor balances friction and hydraulic torques). To model a scenario in which the pump speed can ramp up and coast down repeatedly, option 2 would allow for finer control over motor torque and hence net torque so as to drive pump speed in the desired direction. The convention for a pump motor

under trip control is: if TRIP CF is ON-FORWARD, motor torque is zero and if TRIP CF is ON-REVERSE motor torque is given by CF/TF. Given the motor torque/trip convention, virtually any pump shut off and restart sequence can be modeled. The torque-inertia equation is:

$$I_p \frac{d\omega}{dt} = \tau_{net} = \tau_{motor} - \left(\tau_H + \tau_{fr}\right) \tag{6-131}$$

In general, the net torque has the three components shown above (motor, hydraulic, friction), but the motor torque is only nonzero if A) the pump trip has not occurred or B) a pump trip CF is in an ON-REVERSE state. The assumptions are that an ON-FORWARD trip CF state signifies a pump disconnection from its driving motor while an ON-REVERSE trip CF state allows pump connection to its driving motor. Equation (6-131) above is treated either (1) explicitly in time via a forward Euler technique with the pump inertia, motor torque, hydraulic torque, and friction torque being functions of old-time pump speed, or (2) implicitly with a backward Euler technique and fixed point iteration. For the forward Euler method, the difference equation solved in the pump speed update subroutine is:

$$\begin{aligned} &\omega^n = \omega^{n-1} + \frac{\tau_{net}(\omega^{n-1})}{I_p(\omega^{n-1})}\Delta t \\ &n = current\ time, \qquad n-1 = old\ time, \\ &\Delta t = time\ step \end{aligned} \tag{6-132}$$

For the backward Euler method, the fixed-point iteration scheme is:

$$\omega^n{}_{[0]} = \omega^{n-1} \tag{6-133}$$

$$\omega^n{}_{[i+1]} = \omega^{n-1} + \frac{\tau_{net}\left(\omega^n{}_{[i]}\right)}{I_p\left(\omega^n{}_{[i]}\right)}\Delta t\ , for\ i = 1 \ldots i_{lim} \tag{6-134}$$

The fixed point iteration scheme starts with the guess that pump speed at time level n is that of time level n-1. Then, an iterative solution proceeds wherein the net torque and pump inertia are evaluated at the latest-iterate pump speed to approximate the next-iterate value. Eventually, the next-iterate pump speed matches the last-iterate pump speed within some convergence tolerance. This solution method incorporates n-level information into the solution of the n-level pump speed and ought to improve code performance (in terms of stability and run-time) when the torque-inertia equation is being solved.

6.12.9 Pump Energy Dissipation, Efficiency

The total power imparted to a pumped fluid by an impeller rotating at a given speed is equal to the product of that speed and the delivered hydraulic torque. This quantity is also known as “brake power” or “brake horsepower”. In reality, not all the power delivered to the fluid is manifest as a pressure (head) increase (and therefore as “hydraulic power”) because some fraction is lost to dissipation, appearing as thermal energy added to the fluid. If the brake power is known and the hydraulic power can be calculated, the energy dissipation DISS (W) is:

$$\begin{aligned} DISS &= (brake\ power) - (hydraulic\ power) \\ &= \tau_H \omega \frac{2\pi}{60} - gH\left(\left(1-\alpha_g\right)\rho_f V_f + \alpha_g \rho_g V_g\right) Af \end{aligned} \tag{6-135}$$

This suggests one measure of pump efficiency EFF:

$$EFF = \frac{gH\left(\left(1-\alpha_g\right)\rho_f V_f + \alpha_g \rho_g V_g\right) Af}{\tau_H \omega \frac{2\pi}{60}} \tag{6-136}$$

Where, by this definition, efficiency is the quotient of hydraulic power (i.e. used for pumping) and brake power (the sum of useful and lost power).

The dissipated energy heats up the pumped fluid and cannot, in general, be neglected. A simple way to account for dissipation energy addition is to assume that both the pool and the atmosphere (if both are present) receive amounts of thermal energy that would lead to identical increases in their respective phasic temperatures. Therefore, if the pool and atmosphere were in thermal equilibrium before the dissipation energy addition, they remain so afterwards. Accordingly, a phasic split fraction $f_{P,DISS}$ can be computed as:

$$f_{P,DISS} = \frac{\left(\left(1-\alpha_g\right)\rho_f C_{p,f}\right)}{\left(\left(1-\alpha_g\right)\rho_f C_{p,f} + \alpha_g \rho_g C_{p,g}\right)} \tag{6-137}$$

Then, the rate of dissipation energy addition to the pool, $DISS_P$, is:

$$DISS_P = DISS * f_{P,DISS} = \left(\tau_H \omega \frac{2\pi}{60} - gH\left((1-\alpha_g)\rho_f V_f + \alpha_g \rho_g V_g\right) Af\right)\left(\frac{\left((1-\alpha_g)\rho_f C_{p,f}\right)}{\left((1-\alpha_g)\rho_f C_{p,f} + \alpha_g \rho_g C_{p,g}\right)}\right) \quad (6\text{-}138)$$

While the rate of dissipation energy addition to the atmosphere, $DISS_A$, is:

$$DISS_A = DISS * \left(1 - f_{P,DISS}\right) = \left(\tau_H \omega \frac{2\pi}{60} - gH\left((1-\alpha_g)\rho_f V_f + \alpha_g \rho_g V_g\right) Af\right)\left(\frac{\left(\alpha_g \rho_g C_{p,g}\right)}{\left((1-\alpha_g)\rho_f C_{p,f} + \alpha_g \rho_g C_{p,g}\right)}\right) \quad (6\text{-}139)$$

Note that the dissipation energy source is essentially equivalent, in MELCOR terms, to two external sources (one AE-type and one PE-type) with a RATE interpretation. The dissipation energy is effectively treated as such when the user chooses to model dissipation energy addition.

6.13 Steam Generator Heat Transfer and Flow Morphology Model

The control volume flow regime is an imposed gravitational separation of the pool and atmosphere fields. The resultant pool elevation is used in computing the surface area and void fractions for heat structure heat transfer and flow path advection, respectively. However, this imposed morphology may not be appropriate when modeling certain reactor components, such as the steam generator, where prevailing conditions may differ. Therefore, an optional one-parameter model providing an alternative approach is presented. Although the model provides additional flexibility, the approach remains a limited engineering model that requires thoughtful user input and may require parameter adjustment to achieve satisfactory performance.

The steam generator heat transfer and flow morphology model is activated by adding an optional record, CV_SGM, to the input description for a given CVH control volume (see the CVH Package Users' Manual for details). When both pool and atmosphere fields exist within the control volume, a user-specified target void fraction (SG_VF) is imposed. The model computes a corresponding pool elevation based on the target void fraction. This "virtual" pool elevation is used for

1) Energy transfer between heat structures and coolant, and

2) Computing void fraction values at flow path junctions.

Activating this model will also automatically overwrite the following user input settings:

1) The control volume thermodynamics switch is set to the equilibrium condition (i.e., the same as setting ICVTHR = EQUIL on the CV_THR input record), and

2) Critical pool fraction values for CPFPL and CPFAL (set on record HS_LBP) are effectively set to C4071(1) and C4071(2), defaults are 0.02 and 0.98, respectively. This means there will typically be heat transfer to both pool and atmosphere fields for the affected heat structures.

6.13.1 Model Description

Based on the value of SG_VF, the model computes a “virtual” pool elevation for the control volume. This virtual pool elevation is computed using the SG_VF value as the void fraction of the pool. If this elevation is greater than the upper altitude of the control volume, then the “virtual” pool elevation is set to the top of the control volume and the effective SG_VF value is reduced to a consistent, physical value bounded by 1.0.

Consider a control volume of relative height 1.0, a physical pool volume fraction of 0.1, and relative pool elevation of 0.1. The control volume contains three uniformly sized vertically stacked heat structures. -a shows the pool/atmosphere interface divides the surface area of heat structure A into 30% and 70% for the given fields. Heat structures B and C are unsubmerged and therefore can only exchange energy with the atmosphere field.

Figure 6-18-b illustrates the case where CV_SGM is active and SG_VF is specified as 0.8. The virtual pool elevation is calculated to be 0.5; therefore, heat structure B is subdivided equally between the pool and atmosphere fields. -c illustrates the case when SG_VF = 0.92. The relative height of the virtual pool elevation is initially calculated to be 1.25. But because this exceeds the altitude of the control volume, the virtual pool elevation is reduced to 1.0 and all heat structures are fully submerged. Given the virtual pool elevation was reduced, an effective pool void fraction of 0.90 is computed and applied.

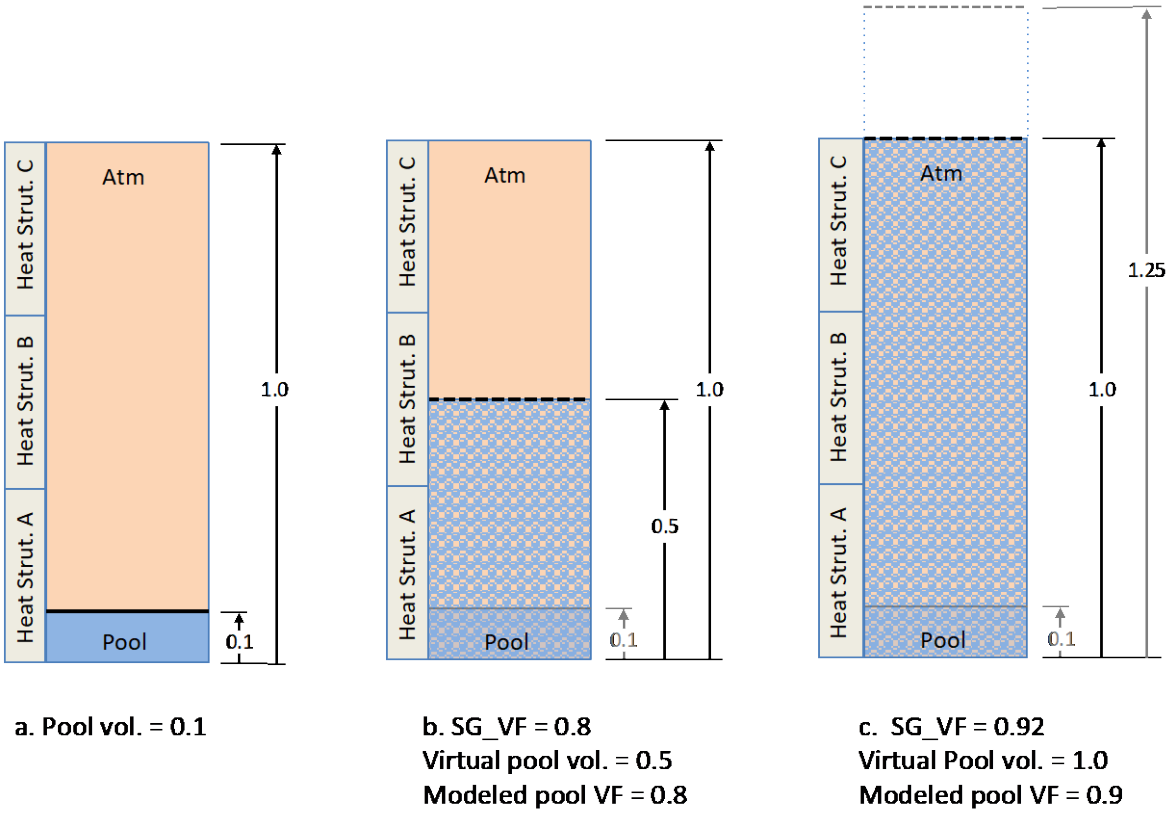


Figure 6-18 Example illustration of how the SG heat transfer model adjusts the pool height based on the input value of SG_VF. (a) no SG model, (b) SG_VF = 0.8, (c) SG_VF = 0.92.

For simple geometries where the volume is monotonically increasing with altitude, the equations for calculating the virtual pool elevation in control volume, i, can be expressed as follows

$$V_{i,pSG} = min\left[V_{i,tot}, V_{i,p}\left(\frac{1}{1-\bar{\alpha}_{SG}}\right)\right] \tag{6-140}$$

$$Z_{i,pSG} = Z_{i,bot} + \left(Z_{i,top} - Z_{i,bot}\right)\frac{V_{i,pSG}}{V_{i,tot}} \tag{6-141}$$

where

$\bar{\alpha}_{SG}$ target CV_SGM void fraction specified though input (SG_VF)

$V_{i,tot}$ total CVH Volume in CV i

$V_{i,p}$ pool volume in CV i

$V_{i,pSG}$ adjusted CV_SGM pool volume in CV i

$Z_{i,bot}$ altitude at the bottom of CV i

$Z_{i,top}$ altitude at the top of CV i

$Z_{i,pSG}$ CV_SGM based virtual pool height of CV i

The effective void fraction is computed as follows:

$$\alpha_{i,pSG} = \frac{V_{i,pSG} - V_{i,p}}{V_{i,pSG}} \tag{6-142}$$

In addition to the heat transfer characteristics in a control volume, the flow path modeling is also affected when CV_SGM is active. The code adjusts the value of the junction void fraction based on two things: (a) the virtual pool elevation and (b) the effective void fraction of the pool.

In part (a) the virtual pool elevation is used to compute the junction void fraction. This requires no change to the geometric algorithm normally used in MELCOR.

Part (b) adjusts the volume fractions at the junction to be consistent with the effective void fraction of the donor pool.

$$\alpha_{jSG} = \alpha_j + (1 - \alpha_j)\alpha_{i,pSG} \tag{6-143}$$

where

α_j volume fraction at flow path junction j computed in the standard way (but using the virtual pool elevation)

$\alpha_{i,pSG}$ CV_SGM based void fraction in pool of CV i (associated with flow path junction j)

α_{jSG} CV_SGM based volume fraction at flow path junction j

Note that due to the CV_SGM model setting the control volume thermodynamics switch to equilibrium, subcooling of the pool can only occur when no atmosphere is present and superheating of the atmosphere vapor can only occur if no pool exists in the control volume.

6.13.2 Bounding Constraints

Under certain bounding conditions specifying an effective void fraction would violate physical constraints, e.g., (1) the maximum heat structure surface temperature within a control volume is at or below T_{sat}, and (2) the maximum void fraction of any coolant entering the control volume is less than the effective pool void fraction, $\alpha_{i,pSG}$. Under these conditions there is no physical mechanism available for an effective void fraction within the pool. For this reason, bounding constraints would be a useful addition to the model. However, without adding a real flow regime map, we can only account for one of these two factors, i.e., the max heat structure surface temperature versus T_{sat}.

To account for this bounding condition, a model feature has been added where the pool swelling model is linearly reduced to zero using a simple function to smooth the transition. The temperature range above T_{sat} over which the model is (de)activated is specified by CVH sensitivity coefficient C4423(1), which by default has a value of 1.0.

Defining

$T_{i,HSmax}$ maximum temperature of any heat structure surface in control volume i,

then we can write

$$\alpha_{i,pSG} = \alpha_{i,pSG} * F_i\left(T_{i,HSmax}\right) \tag{6-144}$$

where

$$F_i\left(T_{i,HSmax}\right) = max\left[min\left[\frac{T_{i,HSmax} - T_{i,sat}}{C4423(1)}, 1.0\right], 0.0\right] \tag{6-145}$$

Note that all CV_SGM control volumes can be deactivated for all conditions by setting the value of sensitivity coefficient C4423(1) to a very large number.

A new CVH plot variable CVH-VOIDSG has been added to enable a user to plot $\alpha_{i,pSG}$.

6.13.3 Example Results

A comparison of the control volume void fractions and gas temperatures in the steam generator section of an example test problem are presented in Figure 6-19 and Figure 6-20. In these figures, “Default” means that the CV_SGM is not active, no heat transfer enhancement factors are input, but the control volumes in the steam generator are set to equilibrium thermodynamics. The “Example Adj.” results are for the modified user input file that represents the best-case results when creatively adjusting user input, but without CV_SGM active. All other results are for when CV_SGM is activated and shows

the effect of changing the SG_VF input parameter over the range of 0.4, 0.6, 0.9, 0.96, 0.98, and 0.99.

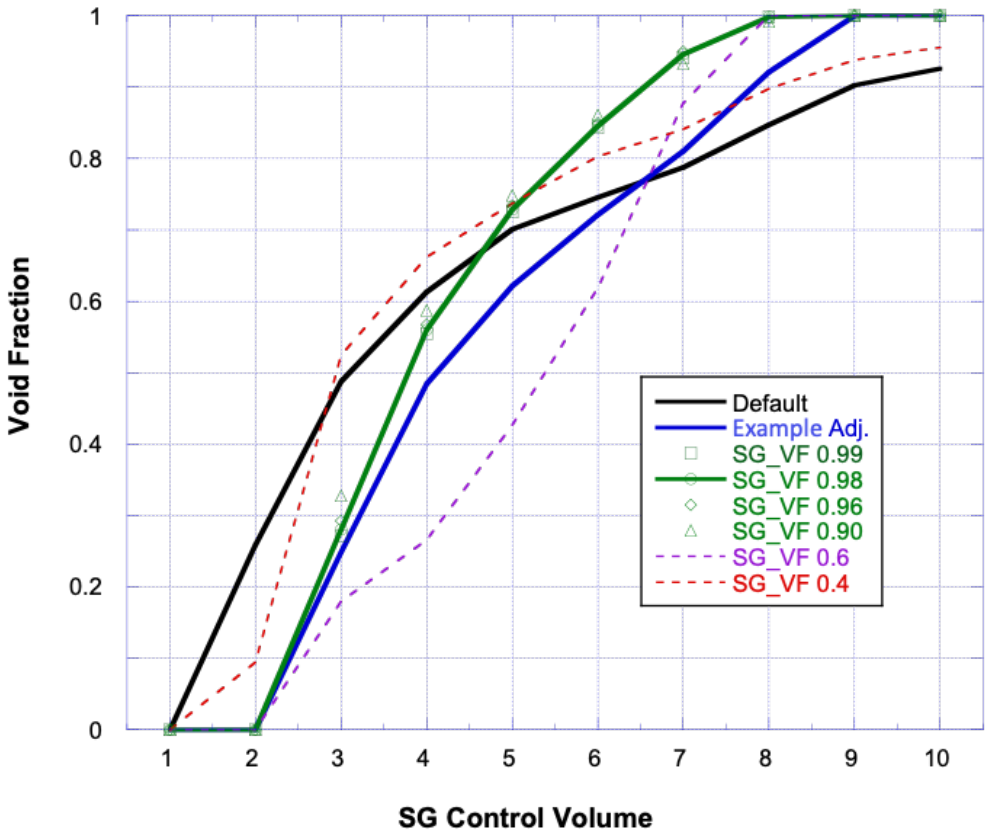


Figure 6-19 Vertical distribution of the void fraction in steam generator control volumes as an example problem

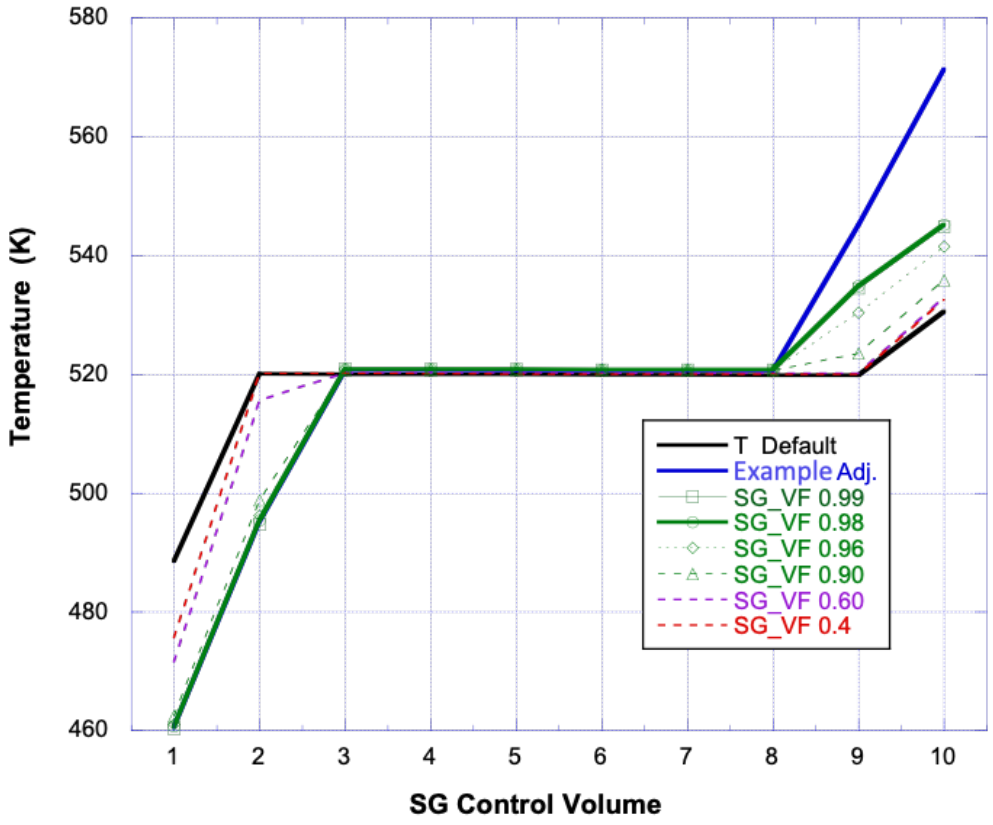


Figure 6-20 Vertical distribution of the atmosphere temperature in steam generator control volumes as an example problem

7. Discussion and Development Plans

7.1 Interphase Forces

An assessment of the simple model for interphase forces (described in Section 5.5) appears to have eliminated more obvious limitations of the previous implementation. Calculations need to be done and compared with data (as represented by more general slip correlations) to assess the overall adequacy of the revised model.

7.2 Critical Flow Modeling

Atmosphere velocities that are significantly supersonic have been observed in some calculations, despite the presence of the critical flow model. This can arise if the phase velocities calculated by the momentum equation are very different. (Because of its greater inertia, the velocity of the pool is sometimes much less than that of the atmosphere before choking is considered.) The problem is that the net mass flux,

calculated with the disparate velocities, may be subcritical (according to the current calculational model) even though one velocity is supersonic.

The entire concept of choking in a two-velocity model may need further examination. In the short term, however, the introduction of the interfacial momentum-exchange term, by reducing the differences between the calculated phase velocities, has gone a long way toward eliminating this problem.

The relatively complicated fits [21] used for Moody and Henry critical flow are not particularly good (a few percent). They are each constructed for two pressure ranges and exhibit discontinuities of several percent at the matching line. The extrapolation properties are poor; the extrapolation often goes negative just outside the fit region. We have found (see Appendix C) that there are simpler representations, with comparable or better accuracy and good extrapolation properties.

The single phase critical model for the vapor field assumes dispersed fog to be negligible in the derivation of the sonic velocity (i.e., the sonic velocity is only a function of the gas properties). Given the assumption of negligible fog mass in the derivation, as the fog mass increases within the flow field for a set sonic velocity, the critical mass flux increases as the fog density increases, which may greatly overestimate the two-phase flow rate. Should users desire to relax the fog density limit specified on SC, the two-phase critical mass flux can be significantly overestimated.

Two simplified two-phase models are commonly employed for lumped parameter codes, namely the homogeneous frozen models (HFM), and the homogeneous equilibrium models (HEM). An HFM approximation has been implemented to permit consideration of the impact of fog on the critical flow rate for the atmospheric field. Common in sub-compartment analyses involving small line breaks into small containment volumes, two-phase critical flow through available flow paths determines the resulting maximum differential pressure across the sub-compartment boundaries. In design basis analyses, where conservative approaches and assumptions are applied, the HFM predicts a lower critical flow rate than the HEM resulting in prediction of greater differential pressures; therefore, the HFM was implemented to support users performing sub-compartment analyses. The implemented model is similarly found in the CONTAIN code and its application facilitates comparison studies between the two codes. The critical two-phase mass flow rate for the atmospheric field with dispersed fog is given by the following equation:

$$G_{C,A} = \rho_A^d C_{s,A}^d \left(\frac{\rho_{gases}^d}{\rho_A^d}\right)^{\frac{1}{2}} \left(\frac{2}{\gamma+1}\right)^{\left[\frac{\gamma+1}{2(\gamma-1)}\right]} \quad (7\text{-}1)$$

A new input card, considered as a global input, called “CVH_ATMCS,” with two keywords to choose from “DEFAULT,” which is derived assuming the fog mass is negligible, and “FMOD” that is the HFM approximation. This record is optional for both

MELGEN and MELCOR inputs. Without this card, MELCOR uses the current DEFAULT model. See CVH package Users' Guide for the input requirement and print option for the sound speed table. It is the intent to eventually revise the default modeling to use HFM.

APPENDIX A: Sensitivity Coefficients

A number of sensitivity coefficients are available in the hydrodynamics (CVH and FL) packages. Their use is described in the CVH Package Users' Guide and most are mentioned at appropriate places in this Reference Manual. This appendix is intended to aid the user in finding those places.

Coefficient	Default Value	Units	Usage, Reference
C4400			**Timestep Control**
(1)	0.5	--	Equation (4-47)
(2)	0.9	--	Equation (4-48)
(3)	0.15	--	Not discussed in this manual. Used only for a calculation involving no flow paths.
(4)	0.05	--	Equation (4-45)
(5)	0.0	Pa	Equation (4-45)
(6)	0.1	--	Executive fallback, Section 4.3 and second paragraph after Equation (4-44)
(7)	0.0	Pa	Executive fallback, Section 4.3 and second paragraph after Equation (4-44)
(8)	0.1	--	Equation (4-46)
(9)	1.0	K	Equation (4-46)
(10)	0.2	--	Executive fallback, Section 4.3 and second paragraph after Equation (4-44)
(11)	1.0	K	Executive fallback, Section 4.3 and second paragraph after Equation (4-44)

Coefficient	Default Value	Units	Usage, Reference
C4401			**Velocity Convergence Criteria**
(1)	0.09	--	Section 4.3, following outline of strategy
(2)	0.0	m/s	Section 4.3, following outline of strategy,
(3)	0.0	--	Implies iteration limit. See discussion in Users' Guide.
(4)	0.0	--	Allows relaxed convergence tolerance. See discussion in Users' Guide.

Coefficient	Default Value	Units	Usage, Reference
C4402			**Minimum Velocity to be Considered for Choking**
(1)	20.0	m/s	First paragraph, Section 6.3

Coefficient	Default Value	Units	Usage, Reference
C4404			**Friction Factor Parameters**
(1)	3.48	--	Colebrook-White, Equation (5-51)
(2)	4.0	--	Colebrook-White, Equation (5-51)
(3)	2.0	--	Colebrook-White, Equation (5-51)
(4)	9.35	--	Colebrook-White, Equation (5-51)
(5)	1/ln(10)	--	Used in solution of Colebrook-White, should not be modified
(6)	1.0	--	Two-phase viscosity, Equation (5-48)
(7)	14.14	--	Used in solution of Colebrook-White, should not be modified
(8)	0.0005	--	Used in solution of Colebrook-White, should not be modified
(9)	0.0	--	Used in solution of Colebrook-White, should not be modified
(10)	1.0	--	Two-phase viscosity, Equation (5-48)
(11)	2.5	--	Two-phase viscosity, Equation (5-48)
(12)	0.9	--	Bound for atmosphere friction, text following Equation (5-49)
(13)	16.0	--	Laminar friction, Equation (5-50)
(14)	2000.0	--	Limiting Reynolds Number, text following Equation (5-51)
(15)	5000.0	--	Limiting Reynolds Number, text following Equation (5-51)
(17)	900.0	--	Interphase friction coefficient, see text following Equation (5-61)

Coefficient	Default Value	Units	Usage, Reference
C4405			**SPARC Bubble Physics Parameters**
(1)	0.01	m	Minimum rise distance, Equation (6-1)
(2)	1.0	--	Rise scale, Equation (6-1)
(3)	0.1	K	Minimum subcooling, Equation (6-2)
(4)	5.0	K	Subcooling scale, Equation (6-2)
(5)	0.99	--	Exit relative humidity, text following Equation (6-3)

Coefficient	Default Value	Units	Usage, Reference
C4406			**Maximum Allowed Fog Density**
(1)	0.1	kg/m^3	Text of Section 5.1.4

Coefficient	Default Value	Units	Usage, Reference
C4407			**Pool/Atmos Heat/Mass Transfer Parameters**
(1)	0.3	m/s	Bubble rise velocity, second paragraph, Section 5.1.3
(2)	0.02	--	Forced convection, Equation (5-21)
(3)	0.14	--	Turbulent free convection in atmosphere, Equation (5-23)
(4)	1/3	--	Turbulent free convection in atmosphere, Equation (5-23)
(5)	0.54	--	Laminar free convection in atmosphere, Equation (5-23)
(6)	1/4	--	Laminar free convection in atmosphere, Equation (5-23)
(7)	0.27	--	Turbulent free convection in pool, Equation (5-22)
(8)	1/4	--	Turbulent free convection in pool, Equation (5-22)
(9)	0.27	--	Laminar free convection in pool, Equation (5-22)
(10)	1/4	--	Laminar free convection in pool, Equation (5-22)
(11)	0.4	--	Maximum pool void, text following Equation (5-31)
(12)	0.9	--	Maximum condensation fraction, text following Equation (5-15)

Coefficient	Default Value	Units	Usage, Reference
C4408			**Pressure Iteration Parameters**
(1)	1001000.	--	Seven packed decimal digits used to disable(1) a corresponding model (for debugging)
(2)	0.005	--	Subcycle step increase, pressure convergence, Equation (4-44)

Coefficient	Usage, Reference
C4409	**Limits and Tolerances for Time-Specified Volumes**
(1-6)	These coefficients are used to test the acceptability and consistency of user input for time-specified volumes. They are not discussed in this reference manual; the description in the users' guide is complete and self-contained.

Coefficient	Default Value	Units	Usage, Reference
C4410			**Vapor Velocity Enhancement during Direct Containment Heating**
(1)	1.0	--	Multiplier on volume-averaged velocity
(2)	1500.0	K	Minimum temperature of airborne debris for application

Coefficient	Default Value	Units	Usage, Reference
C4407			**Pool/Atmos Heat/Mass Transfer Parameters**
			These coefficients can be used to increase heat transfer from the atmosphere of a volume in which direct containment heating is occurring by parametrically increasing the atmosphere velocity that is used in heat transfer correlations.

Coefficient	Usage, Reference
C4411	**Limits and Tolerances for Iterations in the CVT Package**
(1-5)	These coefficients are used to control iterative calculations in the CVT package. They are not discussed in this reference manual; the description in the users' guide is complete and self-contained.

Coefficient	Default Value	Units	Usage, Reference
C4412			**Limits and Tolerances for Iterations in the CVH Package**
(1)	0.01	--	Void fraction convergence, discussion in Section 4.3

Coefficient	Default Value	Units	Usage, Reference
C4413		**--**	**Flow Blockage Friction Parameters**
(1)	3.5	--	Equations (6-22) and (6-25)
(2)	300.0	--	Equations (6-22) and (6-25)
(3)	0.0	--	Equations (6-22)and (6-25)
(4)	0.4	--	Equations (6-22) and (6-25)
(5)	1.0E-6	--	Minimum porosity to be used in Equations (6-22) and (6-25)

Coefficient	Default Value	Units	Usage, Reference
C4414			**Hydrodynamic Volume Fraction**
(1)	1.0E-3	--	Minimum fraction of the initial volume in each segment of the volume/altitude table of a control volume that is always available to hydrodynamic materials, regardless of relocation of virtual volume.

Coefficient	Default Value	Units	Usage, Reference
C4415	**Criteria for solving the flow equations in sparse form**		
(1-4)	These coefficients are used to control iterative calculations in the CVT package. They are not discussed in this reference manual; the description in the users' guide is complete and self-contained.		

Coefficient	Default Value	Units	Usage, Reference
C4500			**Parameters in flashing model for sources and flows**
(1)	1.0	--	Efficiency of flashing, used as a multiplier on the transformation.
(2)	-1.0	--	Fog fraction. If >0.0, it is used to override the value calculated from the Rosin-Rammler distribution.
(3)	65.0E-6	m	Sauter mean droplet diameter.
(4)	50.0E-6	m	Maximum diameter for fog if the RN1 package is inactive
(5)	5.32	--	Power in Rosin-Rammler size distribution, Equation (5-36).

APPENDIX B: The Interphase Force and the Flooding Curve

The interphase force results from exchange of momentum ("drag") between the two fields, pool and atmosphere in MELCOR, when they share a flow path. Many codes such as TRAC [6] and RELAP5 [7] contain detailed models for this force. These models are typically based on specific microscopic pictures of the state of the fluid and, therefore, must contain a number of submodels for different flow regimes. There are at least two practical difficulties in constructing and validating such a model:

(1) The force is not directly measurable; all observable quantities result from delicate balances among this force, wall forces, and gravitational forces. Inertial forces are sometimes involved.

(2) Discontinuities between the submodels or even a lack of smoothness in the transitions between them can result in numerical problems so severe as to prevent calculation of acceptable solutions in any but the simplest cases.

Much of the complexity can be avoided—at the expense of accuracy in some cases—by considering only a single momentum equation, defining an average (mixture) velocity for the two fields, and modeling the relative velocity between them as a constitutive relation. In this approach, referred to as the "drift flux" model, the relative velocity is a function of the local conditions but *not* of their history. RELAP4 [1] is typical of codes employing the drift flux model.

The drift flux model is conventionally cast in terms of the volumetric fluxes defined by

$$j_g \equiv \alpha v_g = \alpha j + \alpha \varepsilon v_r \tag{B-1}$$

$$j_g \equiv \varepsilon v_\ell = \varepsilon j - \alpha \varepsilon v_r \tag{B-2}$$

where

$$\varepsilon \equiv 1 - \alpha \tag{B-3}$$

$$j \equiv j_g + j_\ell \tag{B-4}$$

$$v_r \equiv v_g - v_\ell \tag{B-5}$$

and the fields are identified as ℓ and *g*, denoting "liquid" and "gas", respectively. (Note that the natural dimensions of the volumetric fluxes, $m^3/m^2 \cdot s$, are the same as those of the velocities.) In these relations, v_r or, more usually,

$$j_{g,j} \equiv \alpha \varepsilon v_r \tag{B-6}$$

is considered to be defined by a constitutive equation as a function of α, densities, and geometry.

For a given value of α, the locus of possible values of j_g and j_ℓ as functions of j form a straight line, referred to as a *drift flux line*, as shown in Figure B.1.

The upper left-hand quadrant of Figure B.1 represents a region of countercurrent flow where no quasi-steady solutions are possible. The boundary of this region, formed by the envelope of the drift-flux lines and shown as a dashed curve in the figure, is called the *flooding curve* and defines the limit of (quasi-steady) countercurrent flow. The curve may be parameterized by α and represents the locus of points where

$$\left(\frac{\partial j_g}{\partial \alpha}\right)_j = 0. \tag{B-7}$$

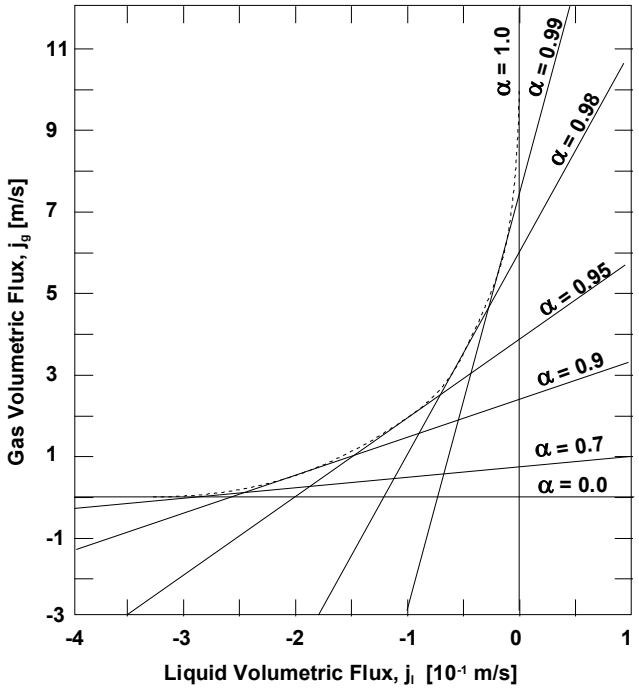


Figure B.1 Drift flux lines and the flooding curve

One empirical correlation that defines the flooding curve, as discussed by Wallis in Section 11.4 of Reference [13], has the form

$$\left(\frac{j_{g,F}}{v_1}\right)^{\frac{1}{2}} + \left(\frac{j_{\ell,F}}{v_0}\right)^{\frac{1}{2}} = 1. \tag{B-8}$$

Here $j_{g,F}$ and $j_{\ell,F}$ define a point on the flooding curve, and v_0 and v_1 are scaling velocities independent of α. Note that this equation is often written with a constant other than 1 on the right-hand side and/or with a coefficient multiplying either or both terms on the left-hand side; these can be absorbed into the scaling velocities without loss of generality.

It is a straightforward exercise to show that if

$$v_r(\alpha) = \frac{1}{\alpha/v_1 + \varepsilon/v_0} \tag{B-9}$$

the flooding curve defined by Equation (B-7) is given by

$$j_{g,F} = \frac{\alpha^2/v_1}{(\alpha/v_1 + \varepsilon/v_0)^2} \tag{B-10}$$

$$j_{\ell,F} = -\frac{\varepsilon^2/v_0}{(\alpha/v_1+\varepsilon/v_0)^2} \tag{B-11}$$

Equations (B-10) and (B-11) clearly satisfy the Wallis flooding relation given by Equation (B-8). In addition, they give a parameterization of that curve by the void fraction α. MELCOR uses velocities rather than volumetric fluxes as the basic variable. In terms of velocities, the parameterization is

$$\alpha_F = \frac{|v_{g,F}|/v_0}{|v_{g,F}|/v_0+|v_{\ell,F}|/v_1} \tag{B-12}$$

The drift flux model is most often used for quasi-steady, nearly incompressible flow. It is relatively simple to ensure that a two-fluid model gives similar results in the corresponding regime. In this limit, where $\partial/\partial t \to 0$ and derivatives of density may be neglected, the momentum equations for the two fields—neglecting momentum flux $\left(v\partial v/\partial x\right)$ terms—may be written as

$$\alpha\frac{\partial P}{\partial x} = \alpha\ \rho_g g_x - \alpha\ F_g v_g - \alpha\ \varepsilon\ F_{\ell g}\left(v_g - v_\ell\right) \tag{B-13}$$

$$\varepsilon\frac{\partial P}{\partial x} = \varepsilon\ \rho_\ell g_x - \varepsilon\ F_\ell v_\ell - \alpha\ \varepsilon\ F_{\ell g}\left(v_\ell - v_g\right) \tag{B-14}$$

The coefficients F_g, F_ℓ, and $F_{\ell g}$ in the various momentum exchange terms are abbreviations for the usual $2f\rho|v|/D$ terms, in the form most commonly employed in simulation codes for two-phase flow. In these equations, g_x is the component of the gravitational acceleration in the x direction; in particular, it is $-g$ if x is measured positive in the upward vertical direction.

If the pressure gradient is eliminated between Equations (B-13) and (B-14), the result can be cast in the form

$$j_g = \frac{\alpha\left(F_\ell + F_{\ell g}\right)}{\alpha\ F_\ell + \varepsilon\ F_g + F_{\ell g}} j - \frac{\alpha\ \varepsilon\ \left(\rho_\ell - \rho_g\right) g_x}{\alpha\ F_\ell + \varepsilon\ F_g + F_{\ell g}} \tag{B-15}$$

Comparison of this equation with Equation (B-1) shows that the quasi-steady solutions of the two-fluid equations have a relative velocity given by

$$v_r = -\frac{\left(\rho_\ell - \rho_g\right) g_x}{\alpha\ F_\ell + \varepsilon\ F_g + F_{\ell g}} \tag{B-16}$$

and comparison of this result with Equation (B-9) suggests that the interphase force be defined by

$$\alpha\ F_\ell + \varepsilon\ F_g + F_{\ell g} = \left(\rho_\ell - \rho_g\right) g_x (\alpha / v_1 + \varepsilon / v_0) \tag{B-17}$$

In MELCOR, we are most concerned with the flooding curve that defines the limit of countercurrent flow. In most cases of interest, the net wall force, $F_\ell + F_g$, is small compared to the interphase force when flooding occurs. Therefore, wall forces are neglected in Equation (B-17), and the interphase force term, $F_{\ell g}$, is set directly equal to the right-hand side of this equation.

Finally, when the differential form of the momentum equation is integrated from volume center to volume center, the integral of $g_x dx$ becomes $-g\Delta z$.

APPENDIX C: Moody Critical Flow

During evaluation of critical flow models for incorporation into MELCOR, the Moody critical flow tables in RELAP4 [1] were compared with the analytic fits in RETRAN [21] for atmospheric and higher pressures. The two representations agree within a few percent in general and within a few tenths of 1% at reactor operating pressures.

The data for each pressure were found to be fit extremely well by the simple expression

$$\frac{\rho_m}{G_c(\alpha)} = \frac{\alpha\,\rho_g}{G_c(1)} + \frac{(1-\alpha)\rho_\ell}{G_c(0)} \tag{C-1}$$

where

$$\rho_m \equiv \alpha\,\rho_g + (1-\alpha)\rho_\ell \tag{C-2}$$

is the mixture density. Equation C-1 states simply that the inverse of the mass-averaged velocity in critical flow is a linear function of the void fraction based on the critical flows at qualities of 1.0 and 0.0. We know of no theoretical basis for this, but the fit is quite good. Figure C.1 shows a typical example. The data are from the RETRAN fits for a pressure of 400 psia; the dashed line shows an approximate linear representation.

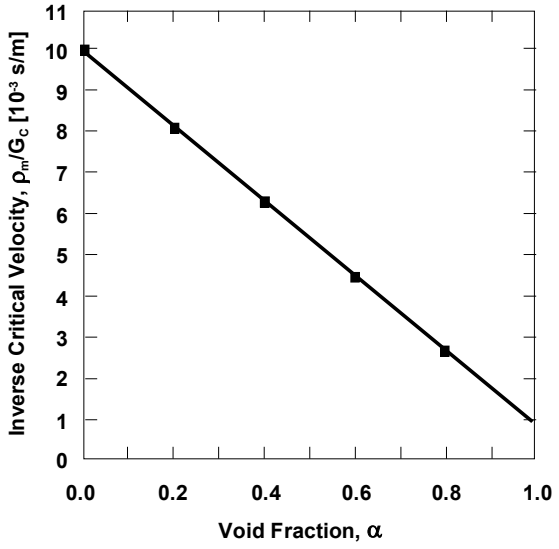


Figure C.1 Moody critical flow data and approximate fit

APPENDIX D: Semi-Implicit Homologous Pump Pressure Head Treatment

D.1 Semi-Implicit Homologous Pump Pressure Head Treatment

Recall the general set of linear equations MELCOR solves during inner velocity iterations:

$$\left(1+\frac{K^{*}_{j,\varphi}\Delta t}{2L_j}\left|v^{n-}_{j,\varphi}+v'_{j,\varphi}\right|+\frac{\alpha_{j,-\varphi}f_{2,j}L_{2,j}\Delta t}{\rho_{j,\varphi}L_j}\right)v^{n}_{j,\varphi} \;-\; \frac{\alpha_{j,-\varphi}f_{2,j}L_{2,j}\Delta t}{\rho_{j,\varphi}L_j}v^{n}_{j,-\varphi}$$
$$+\sum_{s,\psi}C(j,\varphi:s,\psi)v^{n}_{s,\psi}$$

$$= v^{o+}_{j,\varphi}+\frac{K^{*}_{j,\varphi}\Delta t}{2L_j}\left|v'_{j,\varphi}\right|v^{n-}_{j,\varphi}+\frac{\Delta t}{\rho_{j,\varphi}L_j}\left(\hat{P}_i+\Delta P_j-\hat{P}_k\right) \qquad \text{(D-1)}$$
$$+\left(\rho\, g\Delta z\right)^{o}_{j,\varphi}+\frac{\partial\left(\rho\, g\Delta z\right)_{j,\varphi}}{\partial M_{i,P}}\left(\hat{M}^{o}_{i,P}-M^{o+}_{i,P}\right)$$
$$+\frac{\partial\left(\rho\, g\Delta z\right)_{j,\varphi}}{\partial M_{k,P}}\left(\hat{M}^{o}_{k,P}-M^{o+}_{k,P}\right)\Bigg]$$

The left-hand side includes new-time phasic velocities for the j[th] flow path ($V^{n}_{j,\varphi}$ and $V^{n}_{j,-\varphi}$) as well as terms for other flow paths interfaced to the same from/to CVs with which j communicates. The right- hand side includes old-time information, i.e. velocity equation terms that are treated explicitly. Currently, the term ΔP_j appears on the right-hand side of the j[th] flow path velocity equation and is treated as an explicit source term.

It is through ΔP_j that any type of pump, specified on FL_PMP, delivers a pressure head to its assigned flow path. If the newly-implemented homologous pump model is to be semi-implicit such that phasic velocity calculations do not totally treat pump head as an "old-time" quantity, then (1) ΔP_j must be expanded into implicit and explicit terms, and (2) Equation (D-1) above must be rearranged such that (A) the terms multiplying $V^{n}_{j,\varphi}$ and $V^{n}_{j,-\varphi}$ on the left-hand side account for the implicit part of the expanded ΔP_j and (B) the explicit part of the expanded ΔP_j is retained on the right-hand side. Such manipulations of ΔP_j are described below.

D.2 Expansion of ΔP_j

The homologous pump model predicts, among other things, the pump performance in terms of delivered pressure head H as a result of pumping the fluid. The pressure head

is computed as a function of pump speed, ω, and pump capacity (volumetric flow rate), Q. The pump capacity is a function of flow path phasic velocities since it is computed, for flow path j, as:

$$Q_j = Af\left(V_{j,P} + V_{j,A}\right) \tag{D-2}$$

Note Af is the product of flow path area and flow path open fraction. Thus Q_j depends on phasic velocities so that pump head can be expressed as a function of phasic velocities. Neglecting the dependence of head on pump speed (i.e. assuming pump speed is constant) and expanding via a two-term Taylor series about Q_j^{n-} for n- the latest-iterate value (computed with n- velocities during the inner velocity iteration) one recovers:

$$H_j^n = H_j^{n-1} + \left(\frac{dH}{dQ}\right)^{n-1}\left(Q_j^n - Q_j^{n-}\right) \tag{D-3}$$

Alternatively:

$$\Delta P_j = \Delta P_j^{n-1} + \left(\frac{d\Delta P}{dQ}\right)^{n-1}\left(Q_j^n - Q_j^{n-}\right) \tag{D-4}$$

Where,

$$\Delta P_j^{n-1} = \rho_m g H_j^{n-1} \quad ; \quad \left(\frac{d\Delta P}{dQ}\right)^{n-1} = \rho_m g \left(\frac{dH}{dQ}\right)^{n-1}$$

Now, the definition of capacity from Equation (D.3) may be substituted, yielding:

$$\begin{aligned}\Delta P_j = {} & \rho_m g H_j^{n-1} + \rho_m g A f \left(\frac{dH}{dQ}\right)^{n-1}\left(V_{j,P}^n + V_{j,A}^n\right) \\ & - \rho_m g A f \left(\frac{dH}{dQ}\right)^{n-1}\left(V_{j,P}^{n-} + V_{j,A}^{n-}\right)\end{aligned} \tag{D-5}$$

Separating out the terms further:

$$\begin{aligned}\Delta P_j = {} & \rho_m g H_j^{n-1} - \rho_m g A f \left(\frac{dH}{dQ}\right)^{n-1}\left(V_{j,P}^{n-}\right) - \rho_m g A f \left(\frac{dH}{dQ}\right)^{n-1}\left(V_{j,A}^{n-}\right) \\ & + \rho_m g A f \left(\frac{dH}{dQ}\right)^{n-1}\left(V_{j,\varphi}^n\right) + \rho_m g A f \left(\frac{dH}{dQ}\right)^{n-1}\left(V_{j,-\varphi}^n\right)\end{aligned} \tag{D-6}$$

D.3 Modified Velocity Equation

This expression for ΔP_j can be substituted back into Equation (D.1). Also, the new-time information (RHS terms 4 and 5 in Equation (D.6)) can be moved to the LHS of Equation (D.1) while the old-time information (RHS terms 1, 2, and 3 in Equation (D.6)) can be kept on the RHS of Equation (D.1). This results in:

$$
\begin{aligned}
V_{j,\varphi}^{n}\left(1+\left(\frac{K_{j,\varphi}^{*}\Delta t}{2L_j}\right)\left|V_{j,\varphi}^{n-}+V_{j,\varphi}'\right|+\left(\frac{\alpha_{j,-\varphi}f_{2,j}L_{2,j}\Delta t}{\rho_{j,\varphi}L_j}\right)\right.\\
\left.-\zeta\left(\frac{\Delta t}{\rho_{j,\varphi}L_j}\right)\rho_m g A f\left(\frac{dH}{dQ}\right)^{n-1}\right)\\
-V_{j,-\varphi}^{n}\left(\frac{\alpha_{j,-\varphi}f_{2,j}L_{2,j}\Delta t}{\rho_{j,\varphi}L_j}\right.\\
\left.+\zeta\left(\frac{\Delta t}{\rho_{j,\varphi}L_j}\right)\rho_m g A f\left(\frac{dH}{dQ}\right)^{n-1}\right)\\
+\sum_{s,\Psi}\left[C(j,\varphi{:}\,s,\Psi)V_{s,\Psi}^{n}\right]\\
=\ V_{j,\varphi}^{o+}\ +\left(\frac{K_{j,\varphi}^{*}\Delta t}{2L_j}\right)\left(\left|V_{j,\varphi}'\right|V_{j,\varphi}^{n-}\right)\\
+\left(\frac{\Delta t}{\rho_{j,\varphi}L_j}\right)\left(\hat{P}_i\right.\\
\left.+\left(\rho_m g H_j^{n-}-\ \zeta\rho_m g A f\left(\frac{dH}{dQ}\right)^{n-1}\left(V_{j,\varphi}^{n-}+V_{j,-\varphi}^{n-}\right)\right)\right.\\
\left.-\hat{P}_k\right)\ +(\rho g\Delta z)_{j,\varphi}^{o}+\frac{\partial(\rho g\Delta z)_{j,\varphi}}{\partial M_{i,P}}\left(\widehat{M}_{i,P}^{o}-M_{i,P}^{o+}\right)\\
+\frac{\partial(\rho g\Delta z)_{j,\varphi}}{\partial M_{k,P}}\left(\widehat{M}_{k,P}^{o}-M_{k,P}^{o+}\right)
\end{aligned}
\qquad \text{(D-7)}
$$

Thus linearizing ΔP_j leads to (1) new terms in the coefficient multipliers of $V_{j,\varphi}^{n}$ and $V_{j,-\varphi}^{n}$ that capture the implicit part of ΔP_j , and (2) a modified explicit term for ΔP_j. The new terms are highlighted in yellow. Note the factor ζ is positioned so that if set to 1 the semi-implicit formulation can be used while if set to 0, the solution reverts to a fully explicit treatment for ΔP_j ($\Delta P_j = \rho_m g H_j^{n-}$). Note the “old-time” information is denoted (n-1) while the “latest-iterate” information is denoted n-, the distinction being that n-1 quantities are set before the level n velocity iteration loop is entered while n- quantities change with each level n velocity iteration.

D.4 Derivative Computation, $\left(\frac{dH}{dQ}\right)^{n-1}$

The change in pump head with respect to flow (at constant speed) must be evaluated at level n-1 conditions (speed and flow). This derivative expresses how the pump head (which is in general a function of single and two-phase pump performance) varies with capacity in the neighborhood of Q_j^{n-1}. One may produce an H^{n-1} for Q_j^{n-1} as well as an $H^{n-\Delta n}$ for flow $Q_j^{n-\Delta n} = Q_j^{n-1} - f_L Q_j^{n-1}$ and also an $H^{n+\Delta n}$ for flow $Q_j^{n+\Delta n} = Q_j^{n-1} + f_R Q_j^{n-1}$. Note these head values account for single and two-phase effects as appropriate. This produces three data points for head in the neighborhood of Q_j^{n-1} (again the pump speed is assumed constant). These three points can be fitted to a Lagrange polynomial, the derivative of which (when evaluated at n-1) yields the desired $\left(\frac{dH}{dQ}\right)^{n-1}$. Note there is no requirement for even point spacing, i.e. for $f_L = f_R$. In general, according to the formula for first derivatives of quadratic Lagrange polynomials:

$$\left(\frac{dH}{dQ}\right)^{n-1} = \left[\frac{H_a\big((2b)-(b+c)\big)}{(a-b)(a-c)}\right] + \left[\frac{H_b\big((2b)-(a+c)\big)}{(b-a)(b-c)}\right] + \left[\frac{H_c\big((2b)-(a+b)\big)}{(c-a)(c-b)}\right] \tag{D-8}$$

Where,

$$a = Q_j^{n-\Delta n} = Q_j^{n-1} - f_L Q_j^{n-1} \; and \; H_a = H(a, \omega^{n-1})$$

$$b = Q_j^{n-1} \; and \; H_b = H(b, \omega^{n-1})$$

$$c = Q_j^{n+\Delta n} = Q_j^{n-1} + f_R Q_j^{n-1} \; and \; H_c = H(c, \omega^{n-1})$$

References

1. RELAP4/MOD5 A Computer Program for Transient Thermal-Hydraulic Analysis of Nuclear Reactors and Related Systems User's Manual, Volume 1, RELAP4/MOD5 Description, ANCR-NUREG-1335, Idaho Nuclear Engineering Laboratory, Idaho Falls, ID (September 1976).

2. S. E. Dingman, et al., HECTR Version 1.5 User's Manual, NUREG/CR-4507, SAND86-0101, Sandia National Laboratories, Albuquerque, NM (April 1986).

3. K. K. Murata, et al., "Code Manual for CONTAIN 2.0: A Computer Code for Nuclear Reactor Containment Analysis," NUREG/CR-6533, SAND97-1735, 1997.

4. "MAAP4, Modular Accident Analysis Program User's Manual," Volumes 1 and 2, EPRI (1994).

5. JANAF Thermochemical Tables, Dow Chemical Company, Thermal Research Laboratory, Midland, MI (1965).

6. TRAC-PF1, An Advanced Best-Estimate Computer Program for Pressurized Water Reactor Analysis, NUREG/CR-3567, LA-9944-MS, Los Alamos National Laboratory, Los Alamos, NM (February 1984).

7. V. H. Ransom, et al, RELAP5/MOD1 Code Manual Volume 1. System Model and Numerical Methods; Volume 2: *Users Guide and Input Requirements*, NUREG/CR-1826, EGG-2070, Idaho National Engineering Laboratory (March 1982).

8. L. N. Kmetyk, MELCOR 1.8.1 Assessment: FLECHT SEASET Natural Circulation Experiments, SAND91-2218, Sandia National Laboratories, Albuquerque, NM (December 1991).

9. J. G. Collier, Convective Boiling and Condensation, 2nd ed., McGraw-Hill, New York (1981), p. 326.

10. R. B. Bird, W. E. Stewart, and E. N. Lightfoot, Transport Phenomena, John Wiley & Sons, New York (1960), Equation 18.4-25.

11. See, for example, W. H. McAdams Heat Transmission, McGraw Hill, New York (1959). The correlation for the atmosphere is for an unstable temperature gradient; that for the pool is for a stable gradient and a finite surface.

12. R. K. Cole, Jr., Letter Report to Ron Foulds, USNRC, "CVH Pool/Atmosphere Condensation," under FIN A 1339 (August 1992).

13. G. B. Wallis, One-dimensional Two-phase Flow, McGraw-Hill Book Company, New York, NY (1969), Chapter 9.

14. F. Gelbard and J. H. Seinfeld, “Simulation of Multicomponent Aerosol Dynamics,” J. Colloid and Interface Science, **78** (2) (December 1980).

15. D. R. H. Beattie and P. B. Whalley, “A Simple Two-Phase Frictional Pressure Drop Calculational Method,” Int. J. Multiphase Flow, **8** (1) pp. 83-87 (1982).

16. “Flow of Fluids through Valves, Pipes, and Fittings,” Technical Paper 410, Crane Co., Chicago, IL (1969).

17. R. D. Blevins, Applied Fluid Dynamics Handbook, Van Nostrand Reinhold, New York (1984); Section 6.3, with conversion of the Darcy-Weisbach form to the Fanning form used in MELCOR.

18. G. B. Wallis, One-dimensional Two-phase Flow, McGraw-Hill, New York (1969), Section 11.4.

19. P. C. Owczarski and K. W. Burk, SPARC-90: A Code for Calculating Fission Product Capture in Suppression Pools, NUREG/CR-5765, PNL-7723 (October 1991).

20. R. B. Bird, W. E. Stewart, and E. N. Lightfoot, Transport Phenomena, John Wiley & Sons, New York (1960), Equation 15.5-42, with identification of the sound speed as $(\gamma P / \rho)^{1/2}$.

21. RETRAN-02—A Program for Transient Thermal-Hydraulic Analysis of Complex Fluid Systems, Volumes 1-3, NP-1850-CCM, Electric Power Research Institute, Palo Alto, CA (May 1981).

22. D. Dobranich, SAFSIM Theory Manual—A Computer Program for the Engineering Simulation of Flow Systems. SAND92-0693, Sandia National Laboratories, Albuquerque, NM (1993).

23 I. F. Macdonald, M. S. El-Sayed, K. Mow, and F. A. L. Dullien, “Flow Through Porous Media – the Ergun Equation Revisited,” in Ind. Eng. Chem. Fundam., Vol. 18, No. 3, pp. 199-208, 1979, as cited in Reference 10.

24 E. Achenbach, “Heat Transfer and Pressure Drop of Pebble Beds Up to High Reynolds Number,” in Proceedings of Seventh International Heat Transfer Conference, Vol. 1, pp. 3-8, 1982, as cited in Reference 10.

25. S. Ergun, “Fluid Flow Through Packed Columns,” in Chem. Eng. Progress, Vol. 48, No. 2, p. 504 (1952), as cited in Reference 10.

26 R. K. Cole, Jr., “Incorporation of a Core flow Blockage Model into MELCOR,” letter report to John Ridgely, USNRC, November 30, 1995.

27. W. Henk, M. Witlox, and Philip J. Bowen, “Flashing Liquid Jets and Two-Phase Dispersion – A Review,” Health and Safety Executive Contract Research Report 403/2003, Her Majesty’s Stationary Office, England (2002).

28. P. Rosin and E. Rammler, J Inst. Fuel 7, p. 29 (1933), as cited in Reference 29.

29. W. K. Brown and K. H. Wohletz, J. Appl. Phys. 78 (4), pp. 2758-2763, (1955).

30. M. M. Eltkobt, “Fuel Atomization for Spray Modeling,” Prog. Energy Comb. Sci. 8, pp. 61-91 (1982), as cited in Reference 27.

31. M. Razzaghi, “Droplet Size Estimation of Two-Phase Flashing Jets,” Nuclear Engineering and Design 114, pp. 115-124, (1989).

32. A. Koestel, R. Gido, and D.E. Lamkin, “Drop Size Estimates for a Loss-of-Coolant-Accident,” NUREG/CR-1607, LA-8449-MS (1980) as cited in Reference 31.

33. A.F. Mills, Basic Heat and Mass Transfer, (Second Edition) Prentice Hall, 1999.

34 M. Epstein and M.A. Kenton, “Combined Natural Convection and Forced Flow Through Small Openings in a Horizontal Partition With Special Reference to Flows in Multicompartment Enclosures”, Journal of Heat Transfer, Transactions of the ASME Volume 111, pp980-987 (November 1989).

35 R.K., Cole, Jr., “Stratified Flow Model For MELCOR; Implementation Report”, Letter Report to the United States Nuclear Regulatory Commission dated July 28, 2009, Sandia National Laboratories, Albuquerque, NM.

36 RELAP5-3D Code Manual Volume II: User’s Guide and Input Requirements. INEEL-EXT-98-00834 rev 2.4. p 2-44–2-57. June 2005.

37 Olson. Single and Two-phase Performance Characteristics of the Mod-1 Semiscale Pump Under Steady State and Transient Fluid Conditions. ANCR-1165. October 1974.

38 Reeder. Loft System and Test Description. NUREG/CR-0247. July 1978.

39 Lahssuny, Jedral. Universal Correlations for Predicting Complete Pump Performance Characteristics. 2004.

40 Spore, et. al. TRAC-M/FORTRAN 90 (Version 3.0) Theory Manual. NUREG/CR-6274. July 2001.

41 Narabayashi, et. Al. Centrifugal Pump Behavior in Steady and Transient Two-Phase Flow. Journal of Nuclear Science and Technology 23[2] pp 136-150. February 1986.

Control Volume Thermodynamics (CVT) Package

The Control Volume Thermodynamics (CVT) package in the MELCOR code handles thermodynamic calculations for the control volumes included in a MELCOR calculation. Together with the Control Volume Hydrodynamics (CVH) and Flow (FL) packages, it is used to advance the description of the thermal/hydraulic state in the control volumes from one-time level to the next. It obtains the properties of the materials that occupy these volumes from the Non-Condensable Gas (NCG) and Water (H2O) packages. Details may be found in the Reference Manual for these packages. This Reference Manual describes the assumptions, models, and solution strategies used in the various subroutines that make up the CVT package. Because there is no user input for this package, there is no Users' Guide for it.

Contents

List of Figures

1. Introduction

The Control Volume Thermodynamics (CVT) package calculates the thermodynamic state of the materials in each control volume from the total volume, the energies, and the masses calculated by the Control Volume Hydrodynamics (CVH) package. The contents of the control volume may be divided into a *pool* containing water that may be subcooled (liquid) or saturated (two-phase), and an *atmosphere* containing water vapor, liquid water *fog*, and noncondensible gases (NCGs). While the terms "pool" and "atmosphere" suggest a quiescent, stratified configuration, the modeling in CVT assumes only that the two components occupy disjoint subvolumes of the total control volume. (Modeling in other areas of MELCOR, however, often assumes stratification.)

In addition to familiar thermodynamic properties such as pressure, heat capacity, and compressibility, the CVT package also calculates the derivatives of the pressure in the volume with respect to its energy and mass contents for use by the implicit flow solver in the CVH package.

Two thermodynamic options are available: equilibrium and nonequilibrium. In MELCOR, equilibrium thermodynamics assumes that the pool and the atmosphere are in thermal and mechanical equilibrium, i.e., that they have the same temperatures and pressures. This implies instantaneous energy and mass transfers between pool and atmosphere.

When the equilibrium option is used, the distinction between pool and atmosphere does not affect the thermodynamics. CVT uses the total mass and energy contents of the control volume to determine its pressure and temperature. All NCGs are assumed to reside in the atmosphere; the assignment of water to pool or atmosphere is made using time-dependent information from the CVH package. Under current modeling, no water vapor is assigned to the pool, but liquid water may be assigned to the atmosphere as *fog*.

Nonequilibrium thermodynamics, on the other hand, assumes mechanical equilibrium but not thermal equilibrium, so that pressures are equal but temperatures may be different. Complete equilibrium is assumed to exist within the subvolume occupied by the pool and within that occupied by the atmosphere, making this a more limited definition of nonequilibrium than is used in some other codes. The pool may contain water vapor, called *void*, and the atmosphere may contain *fog*. Energy and mass transfers between pool and atmosphere resulting from convection/conduction, radiation, and boiling are explicitly calculated in the CVH package. The elimination of void is also computed in the CVH package, using a bubble rise model. The precipitation of fog is treated by the RadioNuclide package (if it is active) and by the CVH package. The volume expansion or compression work (PdV), done by the pool on the atmosphere must also be accounted for; this is handled in the CVT package itself.

When the nonequilibrium option is used, the thermodynamic state is calculated based on the mass and the total energy of the pool, the masses of the individual components of the atmosphere and their total energy as defined by CVH, and the total volume available to

be shared by pool and atmosphere. Part of CVT's job is to determine the partition of the total volume between the pool and the atmosphere such that the pressures of each are equal. Each subvolume is treated as adiabatic in this calculation; thus *PdV* work is accounted for within the CVT package, but heat and mass transfer are not. Because the pool and the atmosphere are each in internal equilibrium, the nonequilibrium option requires application of the equilibrium assumptions within each subvolume.

The CVT package consists of various interfaces to a mixed-material (water and noncondensible gases) equation of state. Properties for water and for noncondensible gases are obtained from the H2O and NCG packages, respectively. An important feature of the formulation is that it is analytic as well as thermodynamically consistent. Therefore, in contrast to equations of state based on tables or independent polynomial fits, all calculated properties are consistent (for example, a small change in mass or energy produces a change in pressure that agrees to several significant figures with that estimated from the derivatives). The structure of CVT modeling is illustrated in Figure 1-1. The block in the figure labeled "subvolume" implements the mixed-material equation of state in MELCOR as described in Section 2; its use in the nonequilibrium model is described in Section 3.

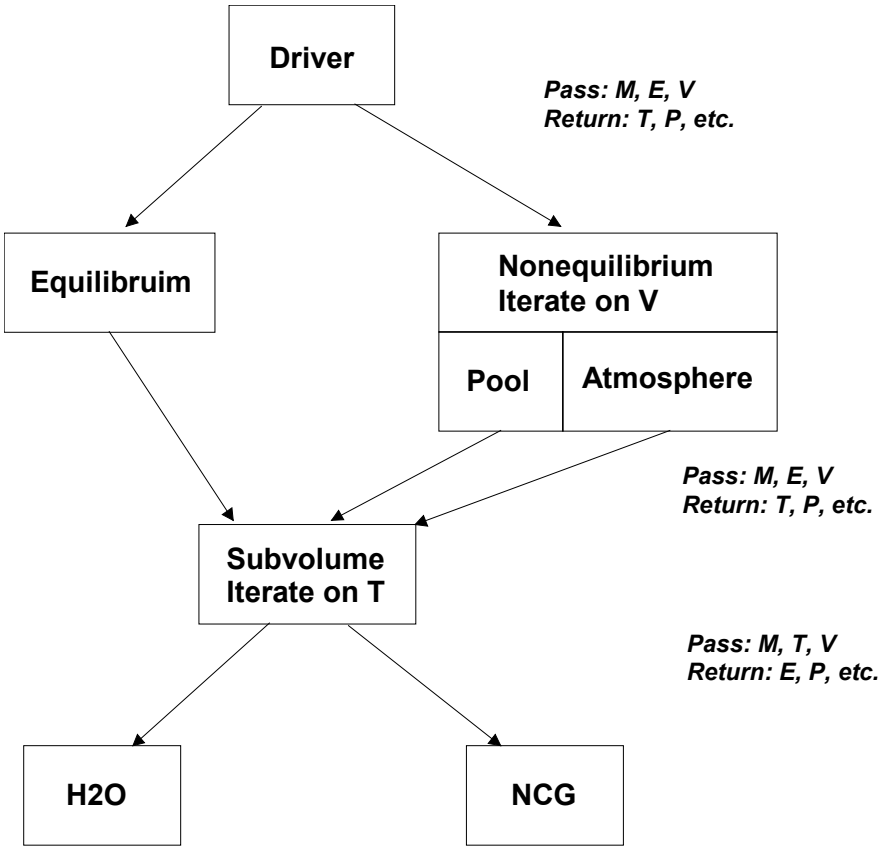


Figure 1.1 CVT Model Structure

2. Equilibrium Thermodynamics

Central to all thermodynamics in MELCOR is a mixed-material equilibrium equation of state that determines all thermodynamic properties of mixtures of water and noncondensible gases. This equation of state is applied directly to MELCOR control volumes for which the option of equilibrium thermodynamics has been specified. For volumes where nonequilibrium thermodynamics is specified, it is applied separately to the atmosphere and to the pool. In the latter case, it reduces to the equation of state for water, because NCGs are not currently permitted in the pool.

The remainder of this section describes the mixed-material equation of state, in the context of its application to an equilibrium volume. The application to a nonequilibrium volume is described in Section 3.

2.1 Governing Assumptions

Equilibrium thermodynamics assumes that the pool and atmosphere temperatures are identical. Given the total volume, the total internal energy, and the masses of water and NCGs, the problem is to calculate the temperature, the total pressure, the partial pressures of each material, and various other thermodynamic properties. Because all temperatures are assumed equal and surface tension effects are not modeled, the distinction between liquid water as fog or in the pool is immaterial within equilibrium thermodynamics, as is the distinction between water vapor in the atmosphere or in a saturated pool. The assignment of liquid and vapor water to the different possible locations is made after the equilibrium state is determined, using time-dependent information from the CVH package. Therefore, within this section, *pool* is used to mean "liquid water" and *atmosphere* to mean "water vapor plus NCGs." In defining the pressure, there are three basic cases:

(1). If no atmosphere is present (which can only occur if there is no NCG), the pressure is that of the pool.

(2). If no pool is present, the NCGs occupy the total volume together with the water vapor; the total pressure is the sum of their partial pressures.

(3). If both pool and atmosphere are present, their pressures as well as their temperatures are equal, and the partial pressure of water in the atmosphere is the saturation pressure for that temperature. If the atmosphere contains NCG, the pool must be subcooled (its pressure is equal to the sum of the saturation pressure and the partial pressures of the NCGs), and its density is therefore greater than saturation density. Thus, the volume of the atmosphere is greater than that which would be occupied by water vapor in the absence of NCGs. As is noted below, certain linearizations of the equation of state of water are used to simplify this calculation.

2.2 Governing Equations

The equilibrium state for a mixture of water and NCGs in a volume, under the assumptions that the NCGs are insoluble in liquid water and form an ideal mixture with water vapor, is given by the simultaneous solution of the equations

$$\rho_\ell = \frac{M_\ell}{V_\ell} \tag{2-1}$$

$$\rho_v = \frac{M_v}{V_v} \tag{2-2}$$

$$M_\ell + M_v = M_w \tag{2-3}$$

$$V_\ell + V_v = V \tag{2-4}$$

$$g_\ell(\rho_\ell, T) = g_v(\rho_v, T) \tag{2-5}$$

$$E = M_\ell e_\ell(\rho_\ell, T) + M_v e_v(\rho_v, T) + \sum_{i=4}^{N} M_i e_i(T) \tag{2-6}$$

$$P = P_\ell(\rho_\ell, T) = P_v(\rho_v, T) + \sum_{i=4}^{N} \frac{M_i R_i T}{V_v} \tag{2-7}$$

where

ρ	is density (kg/m^3),
M	is mass (kg),
V	is volume (m^3),
g	is Gibbs free energy (J/kg),
T	is the temperature (K),
E	is the total internal energy (J),
e	is specific internal energy (J/kg),
P	is pressure (Pa),
R	is gas constant (8.314 J/mole/K),

and the subscripts *w*, ℓ, v, and *i* refer to water, liquid water, water vapor, and the i^{th} NCG, respectively. (Within MELCOR, $4 \le i \le N$ for NCG, where *N* is the total number of *materials* in the problem.) If the Gibbs free energy, *g*, is considered a function of its natural variables (*P*,*T*) rather than of (ρ,*T*), then g_ℓ and g_V are evaluated at $P_\ell(\rho_\ell, T)$ and $P_v(\rho_v, T)$, respectively. (It is helpful to keep in mind that when these equations are applied in MELCOR, it is the masses, the volume, and the total internal energy that are known, so that densities are more natural variables than are pressures. The temperature, which must be determined from densities and energy contents, continues to be thought of as an independent variable in most cases.)

If water is present, but not NCGs, these relations define the equation of state for water (as calculated in the H2O package), which may exist as liquid, as vapor, or as a two-phase mixture. Equations (2-5) and (2-7), expressing equality of the chemical potentials and of the pressures of the two phases, respectively, require that the phases coexist at temperature *T* only at the saturation pressure $P^o_{sat}(T)$, with densities $\rho^o_{\ell,sat}(T)$ and $\rho^o_{v,sat}(T)$, where the superscript "o" denotes "in the absence of NCGs."

In the presence of NCGs, the conditions for equilibrium between liquid water and water vapor are modified. The principal effect, that the pressure in the liquid must be the total pressure, is expressed in Equation (2-7). In addition, the partial pressure of water vapor at equilibrium is slightly modified by the presence of NCGs. However, this change and most of its consequences are negligibly small for the temperature and pressure ranges of interest in MELCOR, as is shown below.

If Equation (2-5) is expanded at fixed temperature about the equilibrium state for pure water, using $dg = v\,dP - s\,dT$, and noting that $g_\ell(\rho^o_{\ell,sat}, T) = g_v(\rho^o_{v,sat}, T)$, the result is

$$\frac{P_\ell(\rho_{\ell,sat}, T) - P^o_{sat}(T)}{\rho^o_{\ell,sat}} = \frac{P_v(\rho_{v,sat}, T) - P^o_{sat}(T)}{\rho^o_{v,sat}}. \qquad (2\text{-}8)$$

Comparison with Equation (2-7) shows that the pressure of water vapor is increased by roughly $\rho^o_{v,sat} \big/ \rho^o_{\ell,sat}$ times the contribution of NCGs to the total pressure. This is a quite small correction under conditions of interest, because the ratio $\rho^o_{v,sat} \big/ \rho^o_{\ell,sat}$ is very small unless the pressure of water vapor is near the critical pressure. Even near the critical point of water, the pressure of the NCGs must be at least comparable to that of the water for the effect to be large. Problems in which such high (supercritical) pressures occur are outside the intended range of application of MELCOR. (Furthermore, if both water and NCG pressures are large, the assumption of ideality used to derive the correction is clearly in error and the correction calculated would be invalid.) Therefore, we neglect the difference between $\rho_{v,sat}$ and $\rho^o_{v,sat}$, and between $P_{sat}(T)$ and $P^o_{sat}(T)$. Because the

change in saturation vapor pressure is neglected, all water vapor in equilibrium with liquid is treated as having *unmodified* saturation properties.

The presence of NCGs increases the pressure—and therefore the density—of the liquid compared to a state at the same temperature in the absence of NCGs. Although the resulting difference between $e(\rho_{\ell,sat}, T)$ and $e(\rho^{o}_{\ell,sat}, T)$ could be calculated, examination of steam tables (such as Keenan and Keyes) shows the difference to be comparable to the effect of a temperature change of only a fraction of a Kelvin for NCG pressures less than 100 MPa. In the interest of simplicity, this difference is neglected. However, the difference in the enthalpy, $h = e + P/\rho$, of the two states is significantly greater, and the difference between h and e is responsible for flow work done in the CVH package. Therefore, a first-order correction is made using $dh = dP/\rho$, and the enthalpy of liquid at pressure P in equilibrium with water vapor is taken as

$$h_{\ell,sat}(P,T) = h(\rho^{o}_{\ell,sat}, T) + \frac{P - P_{sat}(T)}{\rho^{o}_{\ell,sat}}. \tag{2-9}$$

The most important effect of the compression of liquid water by the partial pressure of NCGs is the resulting increase in the volume available to the NCGs. This is included in detail in the model. (To understand its importance, consider the case where there would be no water vapor in the volume in the absence of NCGs. If the compression were ignored, there would be no volume available to the NCGs, and their pressure would be infinite.)

The compression also increases the volume available to water vapor and, because the density of the water vapor continues to be $\rho^{o}_{v,sat}$ under the present assumptions, the presence of NCGs acts to increase the mass of water vapor for a given temperature. In the extreme case where the water would be subcooled in the absence of NCGs (and there would be *no* water vapor in the volume), the NCGs occupy a volume from which liquid water is excluded, and it therefore contains water vapor. The associated change in the amount of each water phase is treated very simply as a small correction after the primary calculation has been completed. Any implications for the energy content of the mixture at a given temperature are ignored.

Under the approximations described above, the need for consideration of Equation (2-5) is eliminated, and Equation (2-6) may be replaced by

$$E = M_w e_w(\rho_w, T) + \sum_{i=4}^{N} M_i e_i(T), \tag{2-10}$$

$$\rho_w = \frac{M_w}{V}, \tag{2-11}$$

involving only the normal thermal equation of state for water in terms of its bulk density. Because the energy of the NCGs does not depend on their volume, Equation (2-10) may be solved for T without the volume available to NCGs being known. This is done iteratively, with repeated calls to the water and NCG equations of state. Newton's method is used, with a secant (and ultimately a bisection) backup. When the iteration has converged, there are several cases to be considered:

(1). If there are no NCGs in the volume, the solution is essentially complete at this point as the water equation of state returns mixed-phase properties when appropriate. The total pressure is that returned by the equation of state for water,

$$P_{tot} = P_w(\rho_w, T) \qquad (no \;\; NCG) \tag{2-12}$$

(2). If there is no liquid water in the volume, the entire volume is available to the NCG, and the total pressure is given by

$$P_{tot} = P_w(\rho_w, T) + \sum_{i=4}^{N} \frac{M_i R_i T}{V} \qquad (no \;\; liquid) \tag{2-13}$$

(from Equation (2-7)), which assumes that water vapor and NCG form an ideal mixture.

(3). Otherwise, the volume contains both liquid water and NCGs. (Note that if the water occupied the volume alone at the same temperature, the state might be either two-phase or subcooled.) As suggested by the discussion above, we treat the effects of NCGs on the properties of water as a relatively small perturbation. In the absence of NCGs, the mass balance

$$V_\ell \rho_{\ell,sat} + V_v \rho_{v,sat} = M_w \tag{2-14}$$

may be used to show that a volume

$$V_v = \min\left\{V, \max\left[0, \frac{V\,\rho_{\ell,sat}(T) - M_w}{\rho_{\ell,sat}(T) - \rho_{v,sat}(T)}\right]\right\} \tag{2-15}$$

would be occupied by water vapor, and the remaining volume

$$V_\ell = V - V_v \tag{2-16}$$

would be occupied by liquid, where $\rho_{\ell,sat}(T)$ and $\rho_{v,sat}(T)$ are the densities of saturated liquid and vapor at the temperature T. (The coding contains a

modification to the water phase boundary to maintain a continuous definition of "liquid" and "vapor" at high pressures and supercritical temperatures; while water should never reach such a state in a reactor, this region of the equation of state may be encountered during iterations within the code.)

The requirement that pool and atmosphere have the same pressure when NCGs are included is imposed by assuming: (1) that the presence of the NCGs causes a reduction of the volume of the liquid by δV ; (2) that the NCGs then occupy the new volume V_n, where

$$V_n = V_v + \delta V; \tag{2-17}$$

and (3) that the resulting pressure is given by the linearization of Equation (2-7) as

$$P_{tot} = \begin{cases} P_w(\rho_w, T) + \dfrac{\delta V}{K V_\ell} & (pool) \\ P_{sat}(T) + \displaystyle\sum_{i=4}^{N} \dfrac{M_i R_i T}{V_v + \delta V} & (atmosphere) \end{cases} \tag{2-18}$$

where P_ℓ in Equation (2-7) has been replaced by P_w (the two are equal in this case, and P_v has been replaced by $P_{sat}(T)$, the partial pressure of water vapor in the atmosphere. K is defined in terms of the compressibility of the liquid phase as

$$K = \left(\rho_\ell \frac{\partial P_\ell}{\partial \rho_\ell} \right)^{-1} \quad . \tag{2-19}$$

Equation (2-18) results in a quadratic equation for δV,

$$(\delta V)^2 + (V_v + K V_\ell \Delta P)\delta V - K V_\ell \sum_{i=4}^{N} M_i R_i T = 0 \quad , \tag{2-20}$$

where

$$\Delta P = P_w - P_{sat}(T) \quad ; \tag{2-21}$$

and the fact that either V_v or ΔP is zero has been used (the former if the water would be subcooled in the absence of NCGs, the latter if it would be two-phase). If the water is two-phase, the total pressure is best evaluated from the second form of Equation (2-18),

$$P_{tot} = P_w(\rho_w, T) + \sum_{i=4}^{N} \frac{M_i R_i T}{V_v + \delta V} \tag{2-22}$$

noting that $P_w(\rho_w, T) = P_{sat}(T)$. Otherwise, for pure liquid (saturated or subcooled), the first form of Equation (2-18) gives

$$P_{tot} = P_w(\rho_w, T) + \frac{\delta V}{KV} \tag{2-23}$$

noting that $V_\ell = V$.

At this point, the new atmosphere volume, V_n, is greater than the volume of water vapor from the water equation of state, V_v, by an amount δV (Equation (2-17)). Therefore, in order to maintain the density of water vapor in the atmosphere as $\rho_{v,sat}(T)$, a mass of water equal to $\rho_{v,sat}(T)\delta V$ is transferred from the liquid to the vapor state by simply modifying the quality of water in the volume to be

$$x' = x + \frac{\rho_{v,sat}(T)\delta V}{M_w}. \tag{2-24}$$

The mass involved is an extremely small fraction of the liquid mass because δV is much smaller than V_ℓ (liquid water is almost incompressible), and $\rho_{v,sat}(T)$ is much smaller than $\rho_{\ell,sat}(T)$. The energy implications of assigning vapor energy rather than liquid energy to this mass are also very small, and are dealt with by the final adjustment to internal energy described below.

Just as V_n is greater than V_v, the remaining liquid volume is *less* than the value V_ℓ used in the water equation of state. In addition to the correction to the liquid enthalpy given by Equation (2-9), a correction is made to the liquid density. As discussed in conjunction with Equation (2-9), the effect on liquid internal energy is small, and has been ignored. Because the uncorrected value has been used consistently, there are no implications with respect to conservation of energy.

The solution is now complete. A temperature has been found such that the known total internal energy is matched *within the convergence tolerance of the iterative solution procedure*. The pressure and all other thermodynamic properties have been evaluated consistently using that temperature and the known volume and masses. Because the total internal energy is the primary—and conserved—quantity, a final adjustment is made to the calculated specific internal energies so that Equation (2-10) is satisfied exactly. The adjustment is made either to the water or to the NCGs, depending of which has the greater total heat capacity.

2.3 Required Derivatives

A number of derivatives of the total pressure are required for the implicit solution of the flow equation (see the CVH/FL Reference Manual) or, as is shown in a later section, for use in solving the nonequilibrium thermodynamic relations in volumes that employ that option. The natural variables for these derivatives are the total energy in the volume, E, the vector of masses in the volume, M, and the available hydrodynamic volume, V. The required derivatives are $(\partial P/\partial E)_{M,V}$, $(\partial P/\partial M_w)_{E,V}$, $(\partial P/\partial M_i)_{E,V}$ [where $4 \le i \le N$ denotes the NCGs], and $(\partial P/\partial V)_{E,M}$. Evaluation of these derivatives is straightforward but tedious. The required expressions are presented in Appendix A.

2.4 Partition of Liquid Water between Pool and Atmosphere

Preceding discussions describe a mixed-material equation of state in which it is assumed that water vapor and noncondensible gases (if both are present) are mixed within a common (sub)volume. For a volume in which nonequilibrium thermodynamics is specified, this equation of state is applied to the pool and atmosphere separately as described in Section 3. Separate mass inventories are calculated for the pool and the atmosphere, and the pool is assumed to contain no noncondensible gases. Therefore, water vapor in the pool is unambiguously interpreted as vapor bubbles (*void*) and liquid water in the atmosphere as vapor droplets (*fog*).

For a volume in which equilibrium thermodynamics is specified, and the equation of state is applied to the entire contents of the volume, the situation is not so simple. The reason is that heat and mass transfer between pool and atmosphere are implicitly included. Therefore, only the total water content and the total energy of the control volume are known, and not the energies and water contents of pool and atmosphere individually.

As stated in the introduction, a basic assumption of the model is that the pool contains water only. If NCGs are present in an equilibrium volume, any liquid water must be subcooled because of the additional partial pressures of the NCGs. The pool—which can contain no NCGs—can therefore contain no bubbles. In the absence of NCGs, there is no such restriction but, to avoid introduction of an unacceptable discontinuity between the two cases, the equilibrium model assumes that there is no void in the pool.

There is no such thermodynamic basis for defining the partition of liquid water between pool and fog. Both consist of subcooled (in the presence of NCGs) or saturated (in their absence) liquid water. As currently coded, the equilibrium model also assumes that there is preferential evaporation of, or condensation to, fog, so that the mass of water in the atmosphere is conserved if possible. That is, the new fog mass is calculated as

$$M_{fog}^{new} = max(M_{fog}^{old} + M_{vap}^{old} - M_{vap}^{new}, 0) \qquad (2\text{-}25)$$

and any remaining mass of liquid water (as calculated by the equilibrium equation of state) is assigned to the pool.

3. Nonequilibrium Thermodynamics

As discussed in Section 1, the implementation of nonequilibrium thermodynamics in MELCOR is more restrictive than in some other codes. The underlying equation of state is strictly equilibrium and does not allow subcooled vapor or superheated liquid. However, if the nonequilibrium option is selected, the pool and atmosphere within a control volume (which are also the two fields in the hydrodynamic equations) are not required to be in thermal or evaporative equilibrium.

3.1 Governing Assumptions

Nonequilibrium thermodynamics assumes that the pool and the atmosphere have equal pressures but that their temperatures may be unequal. As currently implemented, it assumes that all NCGs are in the atmosphere. The pool may contain some water vapor as well as liquid, and the atmosphere may contain some liquid water (*fog*) in addition to vapor. For each field, the total *water* is determined by the mass inventory; the liquid/vapor state is determined implicitly by the energy content.

Given the total volume, the start-of-timestep subvolumes of the pool and of the atmosphere, the total internal energies of the pool and of the atmosphere at these start-of-timestep subvolumes, the mass of water in the pool, and the masses of water and NCGs in the atmosphere, the problem is to calculate the new subvolumes of the pool and of the atmosphere, the temperatures of each, and the common pressure. In computing the subvolumes, it is assumed that the boundary between pool and atmosphere is adiabatic because all heat and mass transfer has already been calculated in the CVH package. However, the work done by displacement of the interface has not yet been calculated, and must be included in CVT. It is for this reason that the start-of-timestep subvolumes are needed.

The general equilibrium model described above for volumes in which equilibrium thermodynamics is specified is used to determine the properties of the atmosphere. It may be seen that the same three cases listed in Section 2.1 for equilibrium thermodynamics may occur:

(1). The atmosphere may contain liquid only (only if there are no NCGs),
(2). It may contain no liquid, and
(3). It may contain water vapor and/or NCGs in addition to liquid.

The first case, of course, has no physical significance because an “atmosphere” containing liquid water only is indistinguishable from “pool”. It is included for completeness and then dealt with outside of the CVT package.

Because of the assumption that there are no NCGs in the pool, it could be treated by using the equation of state for water alone. However, in the interest of consistency, the same general equilibrium model is also used to determine its properties. This also simplifies changes to allow NCGs in the pool if desired in future versions of MELCOR.

3.2 Governing Equations

If a volume is to be treated in the nonequilibrium approximation, the pool and the atmosphere are each described by equilibrium thermodynamics. If we consider the pressure to be a function of masses, energy, and volume, and denoted by $P_{equil}(M,E,V)$ the function defined by the treatment of equilibrium thermodynamics in Section 2.2, the problem is to partition the total volume, V, into two components V_p and V_a such that

$$V_p + V_a = V \tag{3-1}$$

and

$$P_{equil}\left[M_p, E_p + P^{old}(V_p^{old} - V_p), V_p\right] = P_{equil}\left[M_a, E_a + P^{old}(V_a^{old} - V_a), V_a\right] \tag{3-2}$$

Here the superscript “old” refers to the value at the start of the timestep. The terms in Equation (3-2) use the old pressure and the change in volume during the timestep to represent the volume work done by the pool on the atmosphere and vice versa. The old pressure (rather than some average over the timestep) must be used for consistency because, for the solution scheme used in the CVH package, mass moved through the flow paths carries *old* enthalpies, and therefore does work with *old* pressures. For example, in an extreme case where all but an insignificant amount of the atmosphere leaves a control volume during a timestep, it transports an energy of $M_a^{old} h_a^{old}$ in the limit as the remaining mass of the atmosphere goes to zero, consisting of its internal energy plus the volume work done by its motion. This leaves the remaining atmosphere with an energy approaching $-P^{old}V_a^{old}$ before the work done on it by the pool is accounted for in CVT. Its new volume and new energy as determined by CVT must also go to zero, which can only happen if the old pressure is used in the volume work term.

Equation (3-2) is solved iteratively, subject to the constraint of Equation (3-1), by using Newton’s method with a secant (and ultimately a bisection) backup. In order to avoid roundoff problems, the smaller of V_p and V_a is treated as the primary variable within the iteration. The same model used for equilibrium thermodynamics is used to evaluate P_{equil} and its derivatives for the pool and for the atmosphere subvolumes. Because no heat

transfer is included in CVT, the constant entropy (S) (adiabatic) derivatives of pressure with respect to volume

$$\left(\frac{\partial P}{\partial V}\right)_{S,M} = \left(\frac{\partial P}{\partial V}\right)_{E,M} - P\left(\frac{\partial P}{\partial E}\right)_{M,V} \tag{3-3}$$

for the pool and for the atmosphere are used in the Newton iteration. This is consistent with inclusion of the *PdV* terms in Equation (3-2) [note that $(\partial E / \partial V)_{S,M} = -P$].

3.3 Required Derivatives

Once Equation (3-2) has been solved to determine the volumes V_p and V_a, the derivatives required for solution of the flow equation are easily found from those already available for the subvolumes alone. For a differential change in some variable (mass, energy, or volume) for the pool or for the atmosphere, the differential change in pressure (under the current nonequilibrium assumptions) may be found by simultaneous solution of

$$\begin{aligned} dP_p = &\left(\frac{\partial P_p}{\partial E_p}\right)_{M_p,V_p} (dE_p - PdV_p) + \left(\frac{\partial P_p}{\partial M_{w,p}}\right)_{E_p,V_p} dM_{w,p} \\ &+ \sum_{i=4}^{N} \left(\frac{\partial P_p}{\partial M_{i,p}}\right)_{E_p,V_p} dM_{i,p} + \left(\frac{\partial P_p}{\partial V_p}\right)_{E_p,M_p} dV_p \end{aligned} \tag{3-4}$$

$$\begin{aligned} dP_a = &\left(\frac{\partial P_a}{\partial E_a}\right)_{M_a,V_a} (dE_a - PdV_a) + \left(\frac{\partial P_a}{\partial M_{w,a}}\right)_{E_a,V_a} dM_{w,a} \\ &+ \sum_{i=4}^{N} \left(\frac{\partial P_a}{\partial M_{i,a}}\right)_{E_a,V_a} dM_{i,a} + \left(\frac{\partial P_a}{\partial V_a}\right)_{E_a,M_a} dV_a \end{aligned} \tag{3-5}$$

subject to

$$dP = dP_p = dP_a \quad , \tag{3-6}$$

$$dV_p + dV_a = 0 \quad . \tag{3-7}$$

(The summation over NCGs in the pool is retained in Equation (3-4) to allow future generalization, but the corresponding NCG masses are identically zero under current modeling assumptions.) In these equations P_p and P_a are used to represent P_{equil} (pool) and P_{equil} (atmosphere) and, to avoid further complicating the notation, it has been

assumed that a derivative with respect to a component mass is evaluated with all *other* component masses held constant. Note the inclusion of the PdV terms in Equations (3-4) and (3-5); here dE includes only those changes in energy treated by the CVH package, which does not consider volume work done by the pool on the atmosphere of the same calculational volume (or vice versa).

Solution of these equations leads to the desired derivatives. For example, if only dE_p is non-zero, Equations (3-4), (3-5), and (3-6) yield

$$dP = dP_p = \left(\frac{\partial P_p}{\partial E_p}\right)_{M_p,V_p} dE_p + \left[\left(\frac{\partial P_p}{\partial V_p}\right)_{E_p,M_p} - P\left(\frac{\partial P_p}{\partial E_p}\right)_{M_p,V_p}\right] dV_p \quad , \tag{3-8}$$

$$dP = dP_a = \left[\left(\frac{\partial P_a}{\partial V_a}\right)_{E_a,M_a} - P\left(\frac{\partial P_a}{\partial E_a}\right)_{M_a,V_a}\right] dV_a \quad , \tag{3-9}$$

Eliminating dV_p and dV_a by using Equation (3-7), and identifying the adiabatic derivatives defined by Equation (3-3), leads to the result

$$\left(\frac{\partial P}{\partial E_p}\right)_{E_a,M_p,M_a,V} = \frac{\left(\frac{\partial P_a}{\partial V_a}\right)_{S_a,M_a}\left(\frac{\partial P_p}{\partial E_p}\right)_{M_p,V_p}}{\left(\frac{\partial P_a}{\partial V_a}\right)_{S_a,M_a} + \left(\frac{\partial P_p}{\partial V_p}\right)_{S_p,M_p}} \quad . \tag{3-10}$$

The general expressions from which all the required derivatives may be evaluated may be seen to be

$$\left(\frac{\partial P}{\partial \xi_p}\right)_{\{...\}} = \frac{\left(\frac{\partial P_a}{\partial V_a}\right)_{S_a,M_a}\left(\frac{\partial P_p}{\partial \xi_p}\right)_{\{...\}}}{\left(\frac{\partial P_a}{\partial V_a}\right)_{S_a,M_a} + \left(\frac{\partial P_p}{\partial V_p}\right)_{S_p,M_p}} \quad , \tag{3-11}$$

and

$$\left(\frac{\partial P}{\partial \xi_a}\right)_{\{\ldots\}} = \frac{\left(\frac{\partial P_p}{\partial V_p}\right)_{S_p,M_p}\left(\frac{\partial P_a}{\partial \xi_a}\right)_{\{\ldots\}}}{\left(\frac{\partial P_a}{\partial V_a}\right)_{S_a,M_a} + \left(\frac{\partial P_p}{\partial V_p}\right)_{S_p,M_p}}. \qquad (3\text{-}12)$$

where ξ_φ is any one of the variables $E_\varphi, M_{w,\varphi}$, or $M_{i,\varphi}$, "φ" is "p" or "a", and {…} denotes the variables that are to be kept constant under variation of ξ_φ.

3.4 Limit of Vanishing Pool or Atmosphere

The separate mass and energy inventories calculated by CVH for the pool and the atmosphere are used to determine the thermodynamic properties of nonequilibrium volumes. In the limit where either the pool or the atmosphere in such a volume becomes extremely small, any loss of precision in calculating the energy content of the vanishing subvolume has a large effect on the calculated temperature, since its heat capacity goes to zero. Blind application of the model described above leads to unacceptably large excursions in the calculated temperatures of very small pools and atmospheres.

This problem is handled within current coding by switching to the equilibrium formulation whenever a volume specified as nonequilibrium contains only an extremely small (10^{-6}) volume fraction of pool or of atmosphere. One result is that when a nonequilibrium volume becomes completely filled by a two-phase pool, application of the equilibrium model results in an inappropriate separation of the two-phase pool into a pool containing only saturated liquid and an atmosphere containing the saturated water vapor from the bubbles.

This is clearly not a satisfactory solution. The principal problem is that the equilibrium model does not correspond to the infinite mass-and-heat-transfer limit of the nonequilibrium model. However, making the treatment of equilibrium and nonequilibrium volumes completely consistent would not be trivial, and might require elimination of the assumption that a pool can contain no NCGs.

On the other hand, extending the nonequilibrium model to handle the limit of a vanishing pool or atmosphere would require some modification to constrain the calculated temperature difference between pool and atmosphere. This would almost certainly require inclusion of an implicit contribution to the heat transfer (and possibly the mass transfer) between the pool and the atmosphere, either in CVH or (more likely) within CVT.

4. Other Required Properties

Other thermodynamic properties may be calculated from those returned by the H2O and NCG packages, using familiar thermodynamic relationships.

4.1 Specific Heat at Constant Pressure

The H2O package does not return the specific heat of water at constant pressure. Where this is needed, it is evaluated from the standard relationship

$$c_p = c_v + \frac{T\left(\frac{\partial P}{\partial T}\right)_\rho^2}{\rho^2\left(\frac{\partial P}{\partial \rho}\right)_T} \quad . \tag{4-1}$$

4.2 Sound Speed

The speed of sound in the pool and that in the atmosphere are calculated by CVT for use in other packages. The speed of sound, C_s, in a material is defined in terms of the adiabatic bulk modulus as

$$C_S^2 = \left(\frac{\partial P}{\partial \rho}\right)_S . \tag{4-2}$$

Through the use of standard thermodynamic manipulations, Equation (4-2) may be recast in the form

$$C_S^2 = \left(\frac{\partial P}{\partial \rho}\right)_T + \frac{T\left(\frac{\partial P}{\partial T}\right)_\rho^2}{\rho^2 c_v}; \tag{4-3}$$

All of the variables in the latter form are available from the H2O and NCG packages. For a mixture, the sound speed calculated is that given by Equation (4-3) using mixture properties. It corresponds to an equilibrium (long-wavelength) limit, where there is adequate time for energy exchange between species.

Appendix A: Derivatives

This Appendix presents the equations necessary for evaluation of the derivatives of the pressure of an equilibrium mixture with respect to energy, total volume, and various masses. Considerable care has been taken to indicate, using subscripts, what is kept constant in each derivative. When no subscripts are present, the explicit derivative is intended—that is, the derivative in terms of variables appearing explicitly in the definition of the function. However, subscripts are included in some cases where the derivative is, in fact, an explicit one. In order to avoid further complicating the notation, it has been assumed that a derivative with respect to a component mass is evaluated with all *other* component masses held constant.

In the following sections, the pressure is considered to be an explicit function of ρ_w, T, V, and V_n, and of $M_i R_i$ for each NCG. The chain rule is used in evaluating the desired results. V_n is treated as an explicit function of T, M_w, and V, and the $M_i R_i$, with derivatives given in Section A.2. V_v is treated as an explicit function of T, M_w, and V, with derivatives given in Section A.3.

A.1 Derivatives of the Pressure

The various results presented in Equations (2-12), (2-13), (2-22), and (2-23) reduce to only two cases. They are:

(1). If the water would be single-phase (saturated or subcooled) liquid in the absence of NCGs, the total pressure including the effects of NCG is given by Equation (2-23) (noting that $\delta V = V_n$) as

$$P = P_w(\rho_w, T) + \frac{V_n}{KV}, \tag{A-1}$$

with explicit derivatives

$$\frac{\partial P}{\partial \rho_w} = \frac{\partial P_w}{\partial \rho_w}, \tag{A-2}$$

$$\frac{\partial P}{\partial T} = \frac{\partial P_w}{\partial T}, \tag{A-3}$$

$$\frac{\partial P}{\partial V} = -\frac{V_n}{KV^2}, \tag{A-4}$$

$$\frac{\partial P}{\partial V_n} = \frac{1}{KV} \ , \tag{A-5}$$

$$\frac{\partial P}{\partial (M_i R_i)} = 0 \ . \tag{A-6}$$

(2). In all other cases, the pressure is

$$P = P_w(\rho_w, T) + \sum_{i=4}^{N} \frac{M_i R_i T}{V_n} \ , \tag{A-7}$$

with explicit derivatives

$$\frac{\partial P}{\partial \rho_w} = \frac{\partial P_w}{\partial \rho_w} \ , \tag{A-8}$$

$$\frac{\partial P}{\partial T} = \frac{\partial P_w}{\partial T} + \sum_{i=4}^{N} \frac{M_i R_i}{V_n} \ , \tag{A-9}$$

$$\frac{\partial P}{\partial V} = 0 \ , \tag{A-10}$$

$$\frac{\partial P}{\partial V_n} = -\sum_{i=4}^{N} \frac{M_i R_i T}{V_n^2} \ , \tag{A-11}$$

$$\frac{\partial P}{\partial (M_i R_i)} = \frac{T}{V_n} \ . \tag{A-12}$$

The derivative of pressure with respect to energy is evaluated from

$$\left(\frac{\partial P}{\partial E}\right)_{M,V} = \left(\frac{\partial P}{\partial T}\right)_{M,V} \Big/ \left(\frac{\partial E}{\partial T}\right)_{M,V} \ , \tag{A-13}$$

where

$$\left(\frac{\partial P}{\partial T}\right)_{M,V} = \frac{\partial P}{\partial T} + \frac{\partial P}{\partial V_n}\left(\frac{\partial V_n}{\partial T}\right)_{M,V} \ , \tag{A-14}$$

$$\left(\frac{\partial E}{\partial T}\right)_{M,V} = M_w c_{v,w} + \sum_{i=4}^{N} M_i c_{v,i} \quad . \tag{A-15}$$

Here, c_v is the specific heat at constant volume, all water properties and their derivatives come from the H2O package, and NCG properties come from the NCG package. Derivatives of V_n are discussed in Section A.2.

The derivative of pressure with respect to total volume is given by

$$\left(\frac{\partial P}{\partial V}\right)_{E,M} = \frac{\partial P}{\partial V} + \frac{\partial P}{\partial V_n}\left(\frac{\partial V_n}{\partial V}\right)_{T,M} - \frac{\rho_w}{V}\frac{\partial P_w}{\partial \rho_w} + \left(\frac{\partial P}{\partial T}\right)_{M,V}\left(\frac{\partial T}{\partial V}\right)_{E,M} \tag{A-16}$$

Derivation of Equation (A-16) requires multiple application of the chain rule since, for example, V_n is an explicit function of T rather than of E. The additional term is responsible for the appearance of $(\partial P / \partial T)_{M,V}$ (Equation (A-14)) rather than simply $(\partial P / \partial T)$. Finally,

$$\left(\frac{\partial T}{\partial V}\right)_{E,M} = -\left(\frac{\partial E}{\partial V}\right)_{T,M} \Big/ \left(\frac{\partial E}{\partial T}\right)_{M,V} , \tag{A-17}$$

in which

$$\left(\frac{\partial E}{\partial V}\right)_{T,M} = -P_w + T\frac{\partial P_w}{\partial T} \quad ; \tag{A-18}$$

derivation of Equation (A-18) requires use of the Maxwell relation $(\partial S / \partial V)_{T,M} = (\partial P / \partial T)_{M,V}$, and only water contributes because the internal energy of an ideal gas is independent of its volume.

The derivative of pressure with respect to water mass is given by

$$\left(\frac{\partial P}{\partial M_w}\right)_{E,V} = \frac{\partial P}{\partial V_n}\left(\frac{\partial V_n}{\partial M_w}\right)_{T,V} + \frac{1}{V}\frac{\partial P_w}{\partial \rho_w} + \left(\frac{\partial P}{\partial T}\right)_{M,V}\left(\frac{\partial T}{\partial M_w}\right)_{E,V} \quad , \tag{A-19}$$

where

$$\left(\frac{\partial T}{\partial M_w}\right)_{E,V} = -\left(\frac{\partial E}{\partial M_w}\right)_{T,V} \Big/ \left(\frac{\partial E}{\partial T}\right)_{M,V} \quad , \tag{A-20}$$

$$\left(\frac{\partial E}{\partial M_w}\right)_{T,V} = e_w(\rho_w, T) - \frac{1}{\rho_w}\left(\frac{\partial E}{\partial V}\right)_{T,M} \quad ; \qquad \text{(A-21)}$$

the relation between $(\partial E / \partial M_w)_{T,V}$ and $(\partial E / \partial V)_{T,M}$ given by Equation (A-18) is easily seen from Equation (2-10).

The derivative of pressure with respect to the mass of the NCG is given by

$$\left(\frac{\partial P}{\partial M_i}\right)_{E,V} = R_i\left(\frac{\partial P}{\partial (M_i R_i)} + \frac{\partial P}{\partial V_n}\frac{\partial V_n}{\partial (M_i R_i)}\right) - \left(\frac{\partial P}{\partial E}\right)_{M,V}\left(\frac{\partial E}{\partial M_i}\right)_{T,V} \quad , \qquad \text{(A-22)}$$

where

$$\left(\frac{\partial E}{\partial M_i}\right)_{T,V} = e_i(T)\,. \qquad \text{(A-23)}$$

A.2 Derivatives of the Volume Available to NCG

The equations in the preceding section contain derivatives of the volume available to NCGs, V_n. Unless there are both NCGs and liquid water in the volume, δV is zero, V_n is identical to V_v, and we have

$$\left(\frac{\partial V_n}{\partial V}\right)_{T,M} = \left(\frac{\partial V_v}{\partial V}\right)_{T,M} \qquad (no\ \ NCG\ \ or\ \ no\ \ liquid) \qquad \text{(A-24)}$$

$$\left(\frac{\partial V_n}{\partial T}\right)_{M,V} = \left(\frac{\partial V_v}{\partial T}\right)_{M,V} \qquad (no\ \ NCG\ \ or\ \ no\ \ liquid) \qquad \text{(A-25)}$$

$$\left(\frac{\partial V_n}{\partial M_w}\right)_{T,V} = \left(\frac{\partial V_v}{\partial M_w}\right)_{T,V} \qquad (no\ \ NCG\ \ or\ \ no\ \ liquid) \qquad \text{(A-26)}$$

(Expressions for derivatives of V_v are given in Section A.3.) The derivative with respect to NCG masses is more complicated and must be found from the general case below.

If δV is non-zero (case 3 of Section 2.2), we form the differential of Equation (2-20), the quadratic defining δV. It may be put in the form

$$(V^* + 2\delta V)dV_n = \left(V^* + \delta V - V_n\frac{\delta V}{V_\ell}\right)dV_v + \left(V_n\frac{\delta V}{V_\ell}\right)dV$$
$$-KV_\ell\delta V d(\Delta P) + KV_\ell d\left(\sum_{i=4}^{N} M_i R_i T\right) \quad , \qquad \text{(A-27)}$$

where

$$V^* = V_v + KV_\ell\Delta P, \qquad \text{(A-28)}$$

through use of Equation (2-20) itself. As noted in the discussion of Equation (2-20), either $V_v = 0$ if the water is subcooled, or $\Delta P = 0$ if it is saturated. The variable V^* is used to simplify the analysis by reducing the number of cases to be considered.

From Equation (A-27), we may identify the following derivatives (noting that the pressure derivatives vanish unless the water is subcooled, in which case $V_\ell = V$):

$$(V^* + 2\delta V)\left(\frac{\partial V_n}{\partial V}\right)_{T,M} = \left(V^* + \delta V - V_n\frac{\delta V}{V_\ell}\right)\left(\frac{\partial V_v}{\partial V}\right)_{T,M}$$
$$+V_n\frac{\delta V}{V_\ell} + K\delta V\rho_w\frac{\partial P_w}{\partial \rho_w} \qquad \text{(A-29)}$$

$$(V^* + 2\delta V)\left(\frac{\partial V_n}{\partial T}\right)_{M,V} = \left(V^* + \delta V - V_n\frac{\delta V}{V_\ell}\right)\left(\frac{\partial V_v}{\partial T}\right)_{M,V}$$
$$+KV_\ell\left[\delta V\left(\frac{dP_{sat}}{dT} - \frac{\partial P_w}{\partial T}\right) + \sum_{i=4}^{N} M_i R_i\right] \qquad \text{(A-30)}$$

$$(V^* + 2\delta V)\left(\frac{\partial V_n}{\partial M_w}\right)_{T,V} = \left(V^* + \delta V - V_n\frac{\delta V}{V_\ell}\right)\left(\frac{\partial V_v}{\partial M_W}\right)_{T,V}$$
$$-K\delta V\frac{\partial P_w}{\partial \rho_w} \qquad \text{(A-31)}$$

and

$$(V^* + 2\delta V)\left[\frac{\partial V_n}{\partial (M_i R_i)}\right]_{T,V} = KV_\ell T \ . \qquad \text{(A-32)}$$

Unless there are both NCGs and liquid water, Equations (A-29), (A-30), and (A-31) reduce to Equations (A-24), (A-25), and (A-26), respectively.

If there are no NCGs, the derivative of pressure with respect to NCG mass, given by Equation (A-32), reduces to

$$\left[\frac{\partial V_n}{\partial(M_i R_i)}\right]_{T,V} = \frac{K V_\ell T}{V^*} \qquad (no\ \ NCG) \qquad \text{(A-33)}$$

a result that could not have been obtained from the results derived in the absence of NCGs. Some care must be taken in evaluating Equation (A-33) in the case of saturated liquid. Whether this situation is approached from the subcooled side ($V_v = 0, P \to 0+$) or from the two-phase side ($\Delta P = 0, V_v \to 0+$), the denominator V^* goes to 0.

Ideally, one should maintain consistency regardless of how this state of saturated liquid water and no NCG is approached, taking into account that it may involve either case 1 or case 2 in Section A.1. The current coding simply prevents the division by zero. This simple approach has not been found to cause any problems in practice.

A.3 Derivatives of the Vapor Volume

The equations in the preceding section contain derivatives of the volume that would be occupied by water vapor in the absence of NCG with respect to total volume, temperature, and water mass. There are three cases to be considered:

(1). If there is no liquid (no water, or vapor only), $V_v = V$ and

$$\left(\frac{\partial V_V}{\partial V}\right)_{T,M_W} = 1 \qquad \text{(A-34)}$$

$$\left(\frac{\partial V_v}{\partial T}\right)_{M_w,V} = 0\ , \qquad \text{(A-35)}$$

$$\left(\frac{\partial V_v}{\partial M_w}\right)_{T,V} = 0\ . \qquad \text{(A-36)}$$

(2). If there is water but no vapor (liquid only), $V_v = 0$, and

$$\left(\frac{\partial V_v}{\partial V}\right)_{T,M_w} = 0\ , \qquad \text{(A-37)}$$

$$\left(\frac{\partial V_v}{\partial T}\right)_{M_w,V} = 0\ , \qquad \text{(A-38)}$$

$$\left(\frac{\partial V_v}{\partial M_w}\right)_{T,V} = 0 \ . \qquad \text{(A-39)}$$

Otherwise (two-phase water), V_v is given by the primary form of Equation (2-15) (not the bounds), and the derivatives are

$$\left(\frac{\partial V_v}{\partial V}\right)_{T,M_w} = \frac{\rho_{\ell,sat}}{\rho_{\ell,sat} - \rho_{v,sat}} \ , \qquad \text{(A-40)}$$

$$\left(\frac{\partial V_v}{\partial T}\right)_{M_w,V} = \frac{V_\ell \frac{d\rho_{\ell,sat}}{dT} + V_v \frac{d\rho_{v,sat}}{dT}}{\rho_{\ell,sat} - \rho_{v,sat}} \ , \qquad \text{(A-41)}$$

$$\left(\frac{\partial V_v}{\partial M_w}\right)_{T,V} = -\frac{1}{\rho_{\ell,sat} - \rho_{v,sat}}. \qquad \text{(A-42)}$$

The total derivatives of saturation densities with respect to temperature are, of course, to be taken along the phase boundaries. They are evaluated from tables included in the H2O package.

Decay Heat (DCH) Package

The MELCOR decay heat (DCH) package models the decay heat power resulting from the radioactive decay of fission products. Decay heat is evaluated for core materials in the reactor vessel and cavity and for suspended or deposited aerosols and gases. MELCOR couples thermal-hydraulic processes and fission product behavior during the calculation.

Both the radionuclides present in the reactor at the time of the accident and the radionuclide daughter products contribute to the decay heat. In the calculation of decay heat, MELCOR does not explicitly treat each decay chain, since detailed tracking of radionuclide decay chains would be too costly. When the RadioNuclide package is active, the decay heat is calculated for each radionuclide class by using pre-calculated tables from ORIGEN calculations. If the RadioNuclide package is not active, the whole-core decay heat is computed from one of several possible user-specified calculations.

This Reference Manual describes the various models and options available in the DCH package. User input for these models and options is described in the DCH Package Users' Guide.

Contents

List of Figures

List of Tables

1. Introduction

The MELCOR Decay Heat Power (DCH) package models the heating from the radioactive decay of fission products. Decay heat power is evaluated for the fission products assumed to reside in reactor core materials, cavity materials, and in suspended or deposited aerosols and vapors. Decay heat power levels as a function of time are supplied as a utility function within MELCOR that may be called by other phenomenological packages. The DCH package is not involved in the calculation of fission product transport or chemical interactions. These processes are calculated by the RadioNuclide (RN) package (see the RN Package Reference Manual).

Both the radionuclides present in the reactor core and/or cavity from the time of reactor shutdown and the radionuclide daughters from decay contribute to the total decay heat power. In the calculation of decay heat power, the DCH package does not explicitly treat decay chains. Detailed tracking of radionuclide decay chains was seen as computationally costly and too detailed for MELCOR. Instead, when the RN package is active, elemental decay heat power information based on ORIGEN calculations[1,2] is summed into the RN class structure, as described in Section 2.

There are also several options for calculating decay heat power when the RN package is not active (that is, when tracking of fission products is not desired). These are called *whole-core calculations* in the DCH package, although they may be applied to cavity inventories of melt debris as well, and are described in Section 3.

2. Elemental and Radionuclide Class Decay Heat

The DCH package models the decay heat power as a function of time and the total initial inventories of individual *elements*. The default decay heat curves and inventories were obtained from ORIGEN calculations [1], as described in Section 2.1. The grouping of elements into classes for use by the RadioNuclide package is described in Section 2.2.

2.1 SANDIA-ORIGEN Calculations

Calculations were made for prototypical BWR and PWR reactors using the Sandia National Laboratories version of the ORIGEN computer code, and tables of the associated fission product initial inventories and their decay heat powers out to ten days were generated [1,2]. In these tables, all isotopes of an element were summed, and daughters were assumed to remain with the parents. This resulted in 29 elemental groups accounting for over 99% of the decay heat power out to at least two days after reactor shutdown.

The base case ORIGEN run for a PWR used the following assumptions:

1.	3412 MWt Westinghouse PWR
2.	end-of-cycle equilibrium core
3.	three region core, each initially loaded with fuel enriched to 3.3% U-235
4.	constant specific power density of 38.3 MW per metric ton of U
5.	three-year refueling cycle
6.	80% capacity factor
7.	three regions having burnups of 11,000, 22,000, and 33,000 MWd per metric ton of uranium

The base case ORIGEN run for a BWR used the following assumptions:

1.	3578 MWt General Electric BWR
2.	five types of assembly groups
3.	initial enrichment for assemblies, either 2.83% or 2.66% U-235, depending on assembly group
4.	assemblies in core for either 3 or 4 years, depending on assembly group
5.	refueled annually
6.	80% capacity factor

Within the RN package, daughter isotopes are assumed to be transported along with the parents. Thus, the daughter products are assumed to retain the physical characteristics of their parents. This assumption may not be appropriate in some cases, but the ORIGEN analyses showed that the decay heat from the parent elements is generally much greater than that of the daughter products. Because of these considerations, the decay heat of an element's daughter products is included in the decay heat tabulation for the parent element.

The ORIGEN decay heat data are represented in the DCH package in normalized form as decay heat power per unit of reactor operating power at 28 time values after reactor shutdown for each of the 29 elements treated. The ORIGEN results for the PWR were nearly the same as those for the BWR during the first few days after reactor shutdown (within 4%). This similarity results because (1) both reactors use thermal fission of U-235 and Pu-239 as the power source, and (2) decay power during the first few days after shutdown results principally from short-lived radionuclides. Inventories of short-lived radionuclides are proportional to reactor operating power and are relatively insensitive to reactor design and fuel management. Therefore, a single table of normalized decay powers out to 10 days after shutdown is used in the DCH package as representative of both PWRs and BWRs. However, the user may redefine the decay heat power for a given

element (or create one for a “new” element) using the DCH_EL input record, or the user may apply multipliers to the default curves with sensitivity coefficients 3210 and 3211, as described in the DCH Package Users’ Guide.

In general, mass inventories of elements are sensitive to fuel burnup and reactor design. Therefore, two default mass inventories are included in the DCH package for the representative BWR and PWR used in the ORIGEN calculations. The inventory masses of the elements, normalized to grams per unit of reactor operating power (for the PWR and for the BWR), were given by ORIGEN at four times in the equilibrium fuel cycle: start-of-cycle, one-third point, two-thirds point, and end-of-cycle. By default, end-of-cycle values are used, but the user may specify a different fraction of the equilibrium cycle (through sensitivity coefficient 3212), in which case linear interpolation is used to determine the elemental masses at shutdown. For analyses of specific reactors, for which fission product inventories are known (perhaps through separate ORIGEN calculations), the MELCOR user can directly input the element masses using the DCH_EL input record (see the DCH Package Users’ Guide).

The decay heat power and mass for each element were summed over only core fission products and actinides. Thus, the total mass of zirconium (Zr) in the core at the time of shutdown does not include the mass of the Zr in core structural materials.

The decay heat power for a given element at a certain time is estimated by logarithmic interpolation in time of the normalized decay heat powers and dividing by the normalized mass of the particular element in the reactor at the time of shutdown (which includes the masses of its daughter products and is therefore constant) to get a decay heat power per unit mass of the element.

2.2 Radionuclide Classes

The 29 radioactive elements treated by the DCH package are further grouped into chemical classes for tracking by the RN package. Table 2.1 lists the default classes treated by the RN and DCH packages. The remaining elements that do not contribute significant decay heat (< 1%) are enclosed in parentheses. More discussion on classes and their properties is given in the RN Package Reference Manual.

The decay heat power is computed for each class by weighting the elemental decay heats by the relative mass of each element in the class given by the ORIGEN calculations described in Section 2.1. The user may redefine the default class element compositions or define the composition of new classes through input (see input record DCH_CL in the DCH Package Users’ Guide).

All packages that require decay heat power (i.e., COR, CAV, and RN) access a utility provided by the RN package to calculate the total power for the RN class masses residing at a particular location. When the RN package requests a class decay heat power from

the DCH package for any problem time within the range of the present timestep, the returned answer is the average of the class decay heat at the current problem time and the class decay heat at the end of the timestep. Thus, the energy balance calculation is done consistently in the DCH package and the other MELCOR packages distributing the decay heat power. The DCH package edits also reflect this averaging. However, since the first timestep size is not known during the MELGEN setup phase, the MELGEN edit does not show exactly the same decay heat powers as those shown in the first MELCOR edit.

Table 2.1. Default Radionuclide Classes and Member Elements

Class Number and Name	Member Elements
1. Noble gases	Xe, Kr, (Rn), (He), (Ne), (Ar), (H), (N)
2. Alkali Metals	Cs, Rb, (Li), (Na), (K), (Fr), (Cu)
3. Alkaline Earths	Ba, Sr, (Be), (Mg), (Ca), (Ra), (Es), (Fm)
4. Halogens	I, Br, (F), (Cl), (At)
5. Chalcogens	Te, Se, (S), (O), (Po)
6. Platinoids	Ru, Pd, Rh, (Ni), (Re), (Os), (Ir), (Pt), (Au)
7. Transition Metals	Mo, Tc, Nb, (Fe), (Cr), (Mn), (V), (Co), (Ta), (W)
8. Tetravalents	Ce, Zr, (Th), Np, (Ti), (Hf), (Pa), (Pu), (C)
9. Trivalents	La, Pm, (Sm), Y, Pr, Nd, (Al), (Sc), (Ac), (Eu), (Gd), (Tb), (Dy), (Ho), (Er), (Tm), (Yb), (Lu), (Am), (Cm), (Bk), (Cf)
10. Uranium	U
11. More Volatile Main Group Metals	(Cd), (Hg), (Pb), (Zn), As, Sb, (Tl), (Bi)
12. Less Volatile Main Group Metals	Sn, Ag, (In), (Ga), (Ge)
13. Boron	(B), (Si), (P)
14. Water	(Wt)
15. Concrete	(Cc)
16. Cesium Iodide	(classes 2 and 4)

3. Whole Core Decay Heat Calculation

If the RN package is not active in MELCOR, the decay heat power is calculated for the entire core. The user may specify one of four possible options on input record DCH_DPW for this calculation:

(1) a summation of decay heat data from the ORIGEN-based fission product inventories for representative BWRs and PWRs [1,2], scaled if necessary,

(2) the 1979 ANS standard for decay heat power [3],

(3) a user-specified tabular function of whole-core decay as a function of time, or

(4) a user-specified control function to define decay heat.

Each option is described in the following subsections.

3.1 Summation of ORIGEN Data

As discussed in Section 2, a Sandia version of ORIGEN [2] has been used to perform decay heat calculations for prototypical PWR and BWR systems [1]. For the whole-core calculation, the tabulated results of the ORIGEN calculation are summed to produce a total reactor decay heat power, P_{WC}. No elemental or class information is retained; a single decay power value is returned when called by other packages. This is the default whole-core calculation and is the same for PWRs and BWRs.

3.2 ANS Standard Calculation

MELCOR can compute the total decay heat power from the American Nuclear Society's National Standard for light water reactors [3]. This standard prescribes fission product decay heat power for reactor operating histories. Currently, the DCH package uses a user-specified operating time (input on record DCH_OPT) with a constant reactor power, and it also assumes an instantaneous shutdown. The standard prescribes the recoverable energy release rates from fission product decay, but it does not specify the spatial distribution of the deposition of the energy in the reactor materials. This aspect of the problem is reactor specific and must be dealt with by the MELCOR Core package.

The decay heat power is related to the operating power of the reactor via the fission rate and the recoverable energy per fission during operation. The ANS standard assumes that the energy release per fission is independent of time and depends upon the energy spectrum of the neutron flux in the operating reactor and the composition of the reactor core. The energies per fission for U-235, Pu-239, and U-238 are defined in sensitivity coefficient array 3201.

Decay heat power from activation products in reactor structural materials is not specified in the standard, but decay heat powers from U-239 and Np-239 as prescribed by the standard are implemented in the DCH package. The effect of neutron capture in fission products is accounted for by using a formula from the ANS standard for the correction out to a time-since-shutdown of 10^4 s. The DCH package then uses Table 10 from the standard that sets an upper bound on the effect of neutron capture and provides a conservative estimate of the decay heat power. The values from this table are reproduced here in Table 3.1. Because of the conservatism of this table, the ANS standard decay heat power actually contains a discontinuity manifested by a small increase at 10^4 seconds.

MELCOR uses the tables from the ANS standard that prescribe decay heat power from products resulting from the fission of the major fissionable nuclides present in LWRs, specifically thermal fission of U-235 and Pu-239, and fast fission of U-238. These values (from ANS standard Tables 4, 5, and 6) are also reproduced in Table 3.1. The values at the time of shutdown (t = 0.0) were calculated from Tables 7, 8, and 9 of the standard.

Table 3.1 Tabular Values from ANS Standard [3] Used in MELCOR

Time After Shutdown, (sec)	Neutron Capture Correction Factor	Decay Heat Power $F(T,\infty)$		
	$G_{max}(t)$	^{235}U	^{239}Pu	^{238}U
1.0	1.020	1.231E+1	1.027E+1	1.419E+1
1.5	1.020	1.198E+1	1.003E+1	1.361E+1
2.0	1.020	1.169E+1	9.816	1.316E+1
4.0	1.021	1.083E+1	9.206	1.196E+1
6.0	1.022	1.026E+1	8.795	1.123E+1
8.0	1.022	9.830	8.488	1.070E+1
1.0E+1	1.022	9.494	8.243	1.029E+1
1.5E+1	1.022	8.882	7.794	9.546
2.0E+1	1.022	8.455	7.476	9.012
4.0E+1	1.022	7.459	6.707	7.755
6.0E+1	1.022	6.888	6.251	7.052
8.0E+1	1.022	6.493	5.929	6.572
1.0E+2	1.023	6.198	5.685	6.217
1.5E+2	1.023	5.696	5.262	5.621
2.0E+2	1.025	5.369	4.982	5.241
4.0E+2	1.028	4.667	4.357	4.464

Time After Shutdown, (sec)	Neutron Capture Correction Factor	Decay Heat Power $F(T,\infty)$		
	$G_{max}(t)$	^{235}U	^{239}Pu	^{238}U
6.0E+2	1.030	4.282	3.993	4.072
8.0E+2	1.032	4.009	3.726	3.804
1.0E+3	1.033	3.796	3.516	3.598
1.5E+3	1.037	3.408	3.128	3.220
2.0E+3	1.039	3.137	2.857	2.954
4.0E+3	1.048	2.534	2.276	2.366
6.0E+3	1.054	2.234	2.002	2.078
8.0E+3	1.060	2.044	1.839	1.901
1.0E+4	1.064	1.908	1.727	1.777
1.5E+4	1.074	1.685	1.548	1.578
2.0E+4	1.081	1.545	1.437	1.455
4.0E+4	1.098	1.258	1.204	1.204
6.0E+4	1.111	1.117	1.081	1.077
8.0E+4	1.119	1.030	1.000	9.955E-1
1.0E+5	1.124	9.691E-1	9.421E-1	9.383E-1
1.5E+5	1.130	8.734E-1	8.480E-1	8.459E-1
2.0E+5	1.131	8.154E-1	7.890E-1	7.884E-1
4.0E+5	1.126	6.975E-1	6.634E-1	6.673E-1
6.0E+5	1.124	6.331E-1	5.944E-1	6.002E-1
8.0E+5	1.123	5.868E-1	5.462E-1	5.530E-1
1.0E+6	1.124	5.509E-1	5.097E-1	5.171E-1
1.5E+6	1.125	4.866E-1	4.464E-1	4.544E-1
2.0E+6	1.127	4.425E-1	4.046E-1	4.125E-1
4.0E+6	1.134	3.457E-1	3.163E-1	3.224E-1
6.0E+6	1.146	2.983E-1	2.741E-1	2.784E-1
8.0E+6	1.162	2.680E-1	2.477E-1	2.503E-1
1.0E+7	1.181	2.457E-1	2.282E-1	2.296E-1
1.5E+7	1.233	2.078E-1	1.945E-1	1.941E-1
2.0E+7	1.284	1.846E-1	1.728E-1	1.717E-1
4.0E+7	1.444	1.457E-1	1.302E-1	1.299E-1

Time After Shutdown, (sec)	Neutron Capture Correction Factor	Decay Heat Power $F(T,\infty)$		
	$G_{max}(t)$	^{235}U	^{239}Pu	^{238}U
6.0E+7	1.535	1.308E-1	1.099E-1	1.113E-1
8.0E+7	1.586	1.222E-1	9.741E-2	1.001E-1
1.0E+8	1.598	1.165E-1	8.931E-2	9.280E-2
1.5E+8	1.498	1.082E-1	7.859E-2	8.307E-2
2.0E+8	1.343	1.032E-1	7.344E-2	7.810E-2
4.0E+8	1.065	8.836E-2	6.269E-2	6.647E-2
6.0E+8	1.021	7.613E-2	5.466E-2	5.746E-2
8.0E+8	1.012	6.570E-2	4.783E-2	4.979E-2
1.0E+9	1.007	5.678E-2	4.195E-2	4.321E-2

For the ANS standard option, the whole-core power (wc), $P_{wc}(t)$, is given by:

$$P_{wc}(t) = M_{user} G(t) \sum_{i=1}^{3} \frac{P_i F_i(t,T)}{Q_i} + P_{dHE}(t,T) \qquad (3\text{-}1)$$

where

M_{user} = user-input multiplier (default = 1.0)

$G(t)$ = neutron capture correction factor

t = time since reactor shutdown(s)

i = index for fissioning nuclides: U-235, Pu-239, U-238

T = reactor operating time(s)

P_i = power from fissioning of nuclide i (W)

$F_i(t,T)$ = decay power due to nuclide i (MeV/fission)

Q_i = energy per fission of nuclide i (MeV/fission)

The additive term $P_{dHE}(t,T)$ is the decay power from U-239 and Np-239, prescribed by the ANS standard as:

$$P_{dHE}(t,T) = \sum_{i=1}^{3} \frac{P_i}{Q_i} \; [F_{239U}(t,T) + F_{239Np}(t,T)] \tag{3-2}$$

where

$$F_{239U}(t,T) = E_{239U} R[1 - exp(-\lambda_1 T)]\, exp(-\lambda_1 t) \tag{3-3}$$

$$F_{239Np}(t,T) = E_{239Np} R\{\lambda_1[1 - exp(-\lambda_2 T)]\}\; exp(-\lambda_2 t)/(\lambda_1 - \lambda_2)$$
$$-\lambda_2[(1 - exp(-\lambda_1 T))\, exp(-\lambda_1 t)/(\lambda_1 - \lambda_2)] \tag{3-4}$$

E_{239U} = average energy from decay of one U-239 atom (MeV/atom)

E_{239Np} = average energy from decay of one Np-239 atom (MeV/atom)

R = number of atoms of U-239 produced per second per fission per second at shutdown

λ_1 = decay constant for U-239

λ_2 = decay constant for Np-239

For shutdown times less than 10^4 s, the neutron capture correction factor *G(t)* is given by the ANS standard as:

$$G(t) = 1.0 + (3.24\ x\ 10^{-6} + 5.23\ x\ 10^{-10} t) T^{0.4} \psi \tag{3-5}$$

where ψ is the number of fissions per initial fissile atom (user input). For times greater than 10^4 s, *G(t)* is given in tabular form by $G_{\max}(t)$, which may be input as sensitivity coefficients or allowed to default to the values given by the ANS standard.

$F_i(t,T)$ is used in tabular form as given in the ANS standard. The values at each time *t* are found by logarithmic interpolation between successive points in the ANS tables. This form of evaluation does not have significant accuracy loss and is much faster when compared with the primary ANS formulation expressed as a sum of exponentials.

Table 3.2 lists the MELCOR input variables and sensitivity coefficients that are used to implement the ANS decay heat power calculation.

3.3 User-Defined Functions

The whole-core decay heat power, P_{wc}, can be defined by a user-input tabular function of time after shutdown. Alternatively, P_{wc} can be defined as a user-specified control function

of other MELCOR system variables. Either option may be specified on input record DCH_DPW.

Table 3.2 DCH Package Input Variables for ANS Decay Heat Power

ANS Parameter	MELCOR Variable	Input Record
$P_i, i = 1, 2, 3$	U235P, PU239P, U238P	DCH_FPW
T	OPRTIM	DCH_OPT
ψ	PSINC	DCH_NFA
$Q_i, i = 1, 2, 3$	FEU235, FEP239, FEU238	SC3201(I), I=1,2,3
t, time in tabular functions	TIMDCH(I), I=1,…,56	SC3202(I), I=1,…,56
$F_i(t, \infty), i = 1,2,3$	DCHPOW(I,J),I=1,…,56, J=1,2,3	SC3203(I), I=1,…,56 J=1,2,3
$G_{MAX}(t)\{10^4 < t < 10^9\}$	CAPNEU(I),I=1,…,56	SC3204(I), I=1,…,56
R	ATU239	SC3205(1)
E_{239U}	EU239	SC3205(2)
E_{239Np}	ENP239	SC3205(3)
λ_1	DU239	SC3205(4)
λ_2	DNP239	SC3205(5)
M_{user}	ANSMLD	SC3200(1)

4. Activity Calculations (BONUS)

The MELCOR code has a limited capability to perform activity calculations within the code. Consistent with the level of accuracy inherent in an integral severe accident code, BONUS, a simplified code for tracking radioactive decay has been implemented into MELCOR for this purpose. This version of BONUS only evaluates the concentration of fission products after reactor shutdown. The user supplies input for describing decay chains and radionuclide properties (half-life times, decay energies, direct and cumulative yields). The one limitation on decay chains is that they be unidirectional, preventing their cycling (though for actinides this restriction is not imposed). Also, this implementation does not account for transformation between chemical classes. The chemical class inventory is provided at shut-down and no movement between classes due to decay are considered.

4.1 Algorithm For Activity Calcualtions In MELCOR

In what follows, each element is identified by its atomic number *Z*; whereas, the isotopes are identified by *Z* and mass number *A*[1], the MELCOR classes are identified by class name *CL*. The activity A_{ZA} (Bq) of isotope (*Z*, *A*) in the volume under interest is calculated as:

$$A_{ZA}(t) = \lambda_{ZA} N_{ZA}(t) \tag{4-1}$$

where N_{ZA} is the total number of isotopes (*Z*, *A*) in the volume and λ_{ZA} is the radioactive decay constant (s^{-1}).

It is convenient to introduce the relative activity $a_{CL,ZA}(t)$ of isotope (*Z*, *A*) for MELCOR class *CL* since in MELCOR we have data for classes only:

$$a_{CL,ZA}(t) \equiv \frac{A_{ZA}(t)}{A_{CL}(t)} = \frac{A_{ZA}(t)}{\sum_{Z \in CL} \sum_{A} A_{ZA}(t)} = \frac{\lambda_{ZA} N_{ZA}(t)}{\sum_{Z \in CL} \sum_{A} \lambda_{ZA} N_{ZA}(t)} \tag{4-2}$$

where the summation is performed over all isotopes of element *Z* and over all elements *Z* forming RN class *CL*. The variable $a_{CL,ZA}$ measures the relative contribution of isotope (*Z*, *A*) to the total activity A_{CL} of the class *CL*. The variables introduced in Equations (4-1) and (4-2) are similarly considered for all control volumes and their time evolutions are calculated by BONUS subroutines.

Given the values of $a_{CL,ZA}$, the other relative variables can be easily evaluated for the class. For instance, the relative decay heats $h_{CL,ZA}$ are calculated as:

$$h_{CL,ZA}(t) = \frac{E_{ZA} A_{ZA}(t)}{\sum_{Z \in CL} \sum_{A} E_{ZA} A_{ZA}(t)} = \frac{E_{ZA} a_{CL,ZA}(t)}{\sum_{Z \in CL} \sum_{A} E_{ZA} a_{CL,ZA}(t)} \tag{4-3}$$

where E_{ZA} is the decay energy of isotope (*Z*, *A*) in Joules. Moreover, if some dimensional quantity (say, the total mass M_{CL} (kg) in the volume under interest) is known for the class in addition to the relative activities then all other dimensional quantities can be easily calculated. For instance, given the total specific decay heat h_{CL} (J/kg/s) of the class, its total activity A_{CL} (Bq) is calculated as:

[1]In the case of isomer states the mass number is supplied by symbol 'm': ^{135m}Xe.

$$A_{CL}(t) = \frac{H_{CL}(t)}{\sum_{Z\in CL}\sum_{A} E_{ZA} a_{CL,ZA}(t)} = \frac{M_{CL}(t)}{m_n \sum_{Z\in CL}\sum_{A} \lambda_{ZA}^{-1} A a_{CL,ZA}(t)} = \frac{h_{CL}(t) M_{CL}(t)}{\sum_{Z\in CL}\sum_{A} E_{ZA} a_{CL,ZA}(t)} \quad (4\text{-}4)$$

where m_n is the mass of nucleon (kg), A is a mass number. Note that we ignore the difference between proton and neutron masses as well as the nuclear mass defect.

Knowing the total activity of the RN class, we can easily obtain the activity of each isotope in the class:

$$A_{ZA}(t) = A_{CL}(t) a_{CL,ZA}(t) \quad (4\text{-}5)$$

So, for each element, it is enough to know some dimensional quantity and relative activities of its isotopes to evaluate all other radioactive characteristics. Of course, one could choose other basic values instead of a_{ZA}, say, relative concentrations:

$$c_{ZA} \equiv \frac{N_{ZA}}{\sum_{Z\in CL}\sum_{A} N_{ZA}} \quad (4\text{-}6)$$

However, the advantage of our choice is that relative activities generally are not sensitive to the details of irradiation regime (see the next section). This is important as it allows activity estimates when the information about the reactor campaign is incomplete or even lost.

4.2 Initial Inventory

To begin the calculation, initial data are needed for the relative isotope activities of all isotopes at the time of reactor shutdown. Therefore, initial isotopic masses are specified at reactor shutdown, which are reformulated as the relative isotope activity values. It is reasonable to prepare beforehand a special library for several typical reactor campaigns to use in practical calculations in combination with the total decay heats of elements. The library can be prepared by BONUS or some other code. The number of different campaigns can be rather small. Moreover, it can be foreseen situation when the User utilizes the old MELCOR input file without providing any information about initial inventory. In this case some default library variant is chosen.

Such an approach seems to be reasonable as one expects that the initial relative inventories do not drastically differ from one another. Indeed, suppose there is a single fissionable actinide (say ^{235}U). Then the activities of the short-living fission products (for which the life-time is notably less than the stationary irradiation time) can be evaluated in the framework of quasi-stationary approach as:

$$A_{ZA} = \gamma_{ZA}^{235} G \tag{4-7}$$

where *G* is the fission rate (the number of fissions in the volume per second) and γ_{ZA}^{235} is the cumulative yield of isotope (*Z*, *A*) from ^{235}U fission. In this case Equation (4-2) takes the form:

$$a_{CL,ZA} = \frac{\gamma_{ZA}^{235}}{\sum_{Z \in CL} \sum_{A} \gamma_{ZA}^{235}} \tag{4-8}$$

so that $a_{CL,ZA}$ are determined only by the universal constants for this isotope and hence do not depend on irradiation conditions. However, in practice there is a mixture of fissionable actinides (mainly ^{235}U and ^{239}Pu) in the reactor core and their composition depends on the burnup (generally, ^{235}U dominates at low burnups while at high burnups contributions of ^{239}Pu and ^{235}U can be comparable). Furthermore, for the long-living isotopes Equation (4-7) is not valid and A_{ZA} depends on the burnup. In fact for almost stable fission products, A_{ZA} may be proportional to the burnup. However, these effects generally result in only small corrections to the main trend.

The above consideration is illustrated by Figure 4.1 in which the relative activities a_{ZA} are compared for three irradiation regimes with burnups 13.6, 27.2 and 40.8 GWt·d/t. The calculations were performed for six elements (Sr, Zr, and Mo) from the light fission product group and for three elements (I, Xe and Cs) from the heavy FP group.

As seen, the distributions for different regimes are close to each other. Generally, the differences are of several percent, though there are exceptions. For instance, the contribution of ^{106}Mo to the total Mo activity at burnup of 40.8 GW·d/t is one third greater than at 13.6 GW·d/t. This is due to radical difference (by a factor of 4) for this isotope in cumulative yields for ^{235}U and ^{239}Pu. Moreover, the most significant difference (by a factor of 40.8/13.6 = 3) occur for long-lived isotopes (^{90}Sr and ^{137}Cs with half-lives of near 30 years). However, the contributions of these isotopes to the total element activities are negligible, at least during several days after the reactor shut-down.

Finally note that the suggested model can be straightforwardly generalized to grouping of elements into chemical classes. It is enough to simply replace the summation over the elements by summation over the classes.

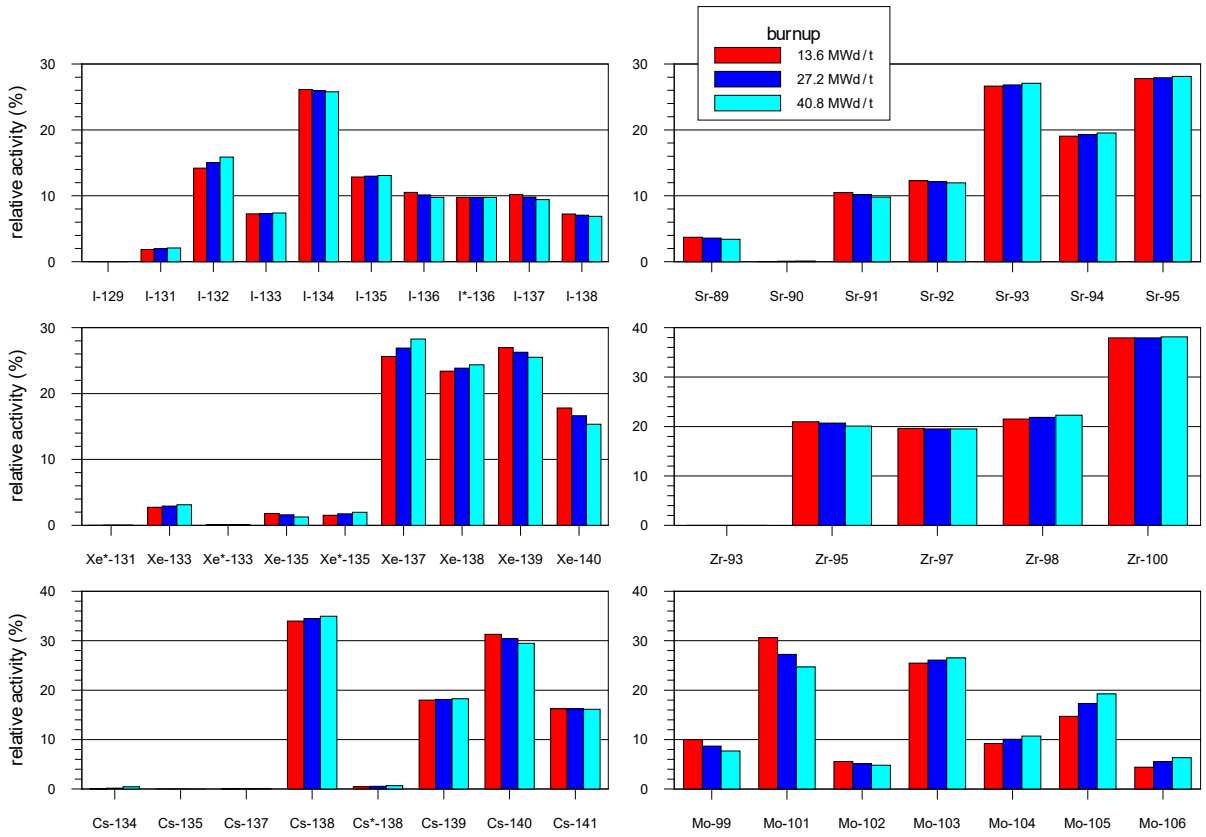


Figure 4.1 Histogram for relative activities of isotopes for three burnup values.

References

1. R. M. Ostmeyer, An Approach to Treating Radionuclide Decay Heating for Use in the MELCOR Code System, NUREG/CR-4169, SAND84-1404, Sandia National Laboratories, Albuquerque, NM (May 1985).

2. D. E. Bennett, SANDIA-ORIGEN User's Manual, NUREG/CR-0987, SAND79-0299, Sandia National Laboratories, Albuquerque, NM (October 1979).

3. American Nuclear Society Standards Committee Working Group ANS-5.1, American National Standard for Decay Heat Power in Light Water Reactors, ANSI/ANS-5.1-1979, American Nuclear Society, La Grange Park, IL (1979).

Fan Cooler (FCL) Package

The MELCOR ESF Package models the physics for the various Engineered Safety Features (ESFs) in a nuclear power plant. The Fan Cooler (FCL) package constitutes a subpackage within the ESF Package and calculates the heat and mass transfer resulting from operation of the fan coolers. The removal of fission product vapors and aerosols by fan coolers is to be modeled within the RN package. Those models have not yet been implemented. This Reference Manual gives a description of the physical models and numerical solution schemes implemented in the FCL package.

User input for running MELGEN and MELCOR with the FCL package activated is described separately in the Fan Cooler Package Users' Guide.

Contents

1. Introduction

The MELCOR ESF package models the thermal-hydraulic behavior of various engineered safety features (ESFs) in nuclear power plants. One important ESF is a fan cooler, which is a large heat exchanger used to remove heat from the containment building. Such coolers circulate hot containment atmosphere gases over cooling coils through which secondary water coolant at low temperatures is circulated. This results in the removal of heat by convection and condensation heat transfer.

The Fan Cooler (FCL) package constitutes a subpackage within the ESF package and calculates the heat and mass transfer resulting from operation of the fan coolers. Two models are available in MELCOR for simulating fan coolers, the simple MARCH model based on the fan cooler model in the MARCH 2.0 code [1] and a more rigorous mechanistic model based on the CONTAIN mechanistic fan cooler model [2]. The user may select between these two models and vary cooler design parameters through the MELGEN input described in detail in the User Guide.

The removal of fission product vapors and aerosols by fan coolers is not modeled within the FCL package. Models to simulate those processes have not yet been implemented but will eventually be included in the RadioNuclide (RN) package.

2. Model Description

2.1 MARCH Model

Note that the MARCH fan model is based on the MARCH 2.0 code, though several important extensions to the MARCH model have been made and are noted here. The user may optionally specify a separate discharge control volume for the fan cooler outlet air flow. The user may also specify a control function to switch the cooler on or off. The maximum condensation rate is limited to the water vapor inlet flow rate. Finally, the MELCOR implementation roughly partitions the total heat transfer coefficient into separate convection and condensation components to try to account for the effects of noncondensible gases and superheated atmosphere. The user can control how this partitioning is made by adjusting the sensitivity coefficients used in the heat transfer correlation.

The total effective heat transfer coefficient, h_T, used in the MARCH fan cooler model is an empirical relation taken from the Oconee Power Reactor Final Safety Analysis Report [3] (British units of Btu/hr-ft^2-F have been converted to SI units of W/m^2-K) which is given by:

$$h_T = 590.54 + 3603.4 X_{H2O} \qquad \text{for } X_{H2O} \le 0.26 \qquad (2\text{-}1)$$

$$h_T = h_T(0.26) + 2325.25(X_{H2O} - 0.26) \qquad \text{for } X_{H2O} > 0.26 \qquad (2\text{-}2)$$

where X_{H2O} is the water vapor mole fraction and h_T in Equation (2-2) is evaluated from Equation (2-1) for X_{H2O} equal to 0.26, yielding a value of 1527.42 W/m^2-K. The heat transfer coefficient, h_T, is to be applied with the total effective fan cooler surface area, A_{eff}, and the temperature difference between the primary and secondary average fluid temperatures, $T_{P,avg}$ and $T_{S,avg}$, respectively. In MELCOR, it is assumed that this heat transfer coefficient can be divided into two components:

(1). a convective component, h_H, transferring only sensible heat, and
(2). a condensation component, h_M, transferring only latent heat.

The convective component is assumed to correspond to the heat transfer for completely dry conditions (i.e., X_{H2O}=0.0) times a sensitivity coefficient multiplier, F_H (default value of 1.0), such that

$$h_H = 590.54 F_H \qquad (2\text{-}3)$$

It follows that

$$h_M = h_T - h_H \qquad (2\text{-}4)$$

The constants in Equations (2-1) through (2-3) have been implemented as sensitivity coefficient array C9001 (see Appendix A).

The total fan cooler heat transfer rate Q_T is therefore

$$Q_T = Q_H + Q_M \qquad (2\text{-}5)$$

where

$$Q_H = h_H A_{eff}(T_{P,avg} - T_{S,avg}) \qquad (2\text{-}6)$$

$$Q_M = h_M A_{eff}(T_{P,avg} - T_{S,avg}) \qquad (2\text{-}7)$$

The average primary and secondary fluid temperatures, $T_{P,avg}$ and $T_{S,avg}$, respectively, are themselves functions of the primary and secondary fluid inlet temperatures, $T_{P,in}$ and $T_{S,in}$, the primary and secondary mass flow rates through the fan cooler, W_P and W_S, and the fan cooler heat transfer rates. Assuming that the average primary temperature decreases only in response to sensible heat transfer, while the average secondary temperature increases in response to the total heat transfer results in:

$$T_{P,avg} = T_{P,in} - \frac{Q_H}{2W_P c_{p,\,P}} \tag{2-8}$$

$$T_{S,avg} = T_{S,in} + \frac{Q_T}{2W_S c_{p,\,S}} \tag{2-9}$$

where $c_{p,\,P}$ and $c_{p,\,S}$ are specific heat capacities at constant pressure for the primary and secondary fluids. Noting that $Q_H/Q_T = h_H/h_T$, simple substitution of Equations (2-8) and (2-9) into Equations (2-4) through (2-7) gives

$$Q_T = h_T A_{eff}\left[T_{P.in} - T_{S,in} - \frac{Q_T}{2}\left(\frac{1}{W_S c_{p,\,S}} + \frac{h_H/h_T}{W_P c_{p,\,P}}\right)\right] \tag{2-10}$$

Solving for the total heat transfer rate Q_T, Equation (2-10) gives

$$Q_T = h_T A_{eff}\frac{T_{P,in} - T_{S,in}}{\left[1 + \frac{1}{2}\left(\frac{h_T}{W_S c_{p,S}} + \frac{h_H}{W_P c_{p,P}}\right)A_{eff}\right]} \tag{2-11}$$

The maximum condensation heat transfer rate is also limited to the water vapor inlet flow rate:

$$Q_{M,\max} = Y_{H2O} W_P h_{fg} \tag{2-12}$$

where Y_{H2O} is the water vapor *mass* fraction and h_{fg} is the latent heat of vaporization of water. If Q_M is limited to $Q_{M,\max}$, Q_H and Q_T are recalculated from Equations (2-5) and (2-10).

The effective surface area A_{eff} is calculated in MELGEN from the rated primary and secondary flows and temperatures (W_{PR}, W_{SR}, T_{PR}, and T_{SR}), from the total and convective heat transfer coefficients evaluated at the rated water vapor mole fraction (h_{TR} and h_{HR}), and from the cooler capacity Q_R at those conditions, using Equation (2-10):

$$A_{eff} = \frac{Q_R}{h_{TR}(T_{PR} - T_{SR}) - \frac{Q_R}{2}\left(\frac{h_{TR}}{W_{SR} c_{p,S}} + \frac{h_{HR}}{W_{PR} c_{p,P}}\right)} \tag{2-13}$$

Here, W_{PR} is the rated primary mass flow, related to the rated volumetric flow input by

$$W_{PR} \equiv \rho_{PR}\ \dot{V}_{PR} \tag{2-14}$$

Where the gas density, ρ_{PR}, is evaluated at T_{PR} and a pressure of one atmosphere (101325 Pa).

Note that, unlike the MARCH model, conditions actually used in the transient calculation in MELCOR may, in general, be different from rated flows and temperatures.

All mass and energy transfers calculated by the fan cooler model are communicated to the Control Volume Hydrodynamics (CVH) package through the standard interface provided for such interpackage transfers.

Fan coolers may be specified for any control volume. The user may optionally specify a separate discharge control volume for the fan cooler outlet air flow, in which case the cooler functions somewhat like a flow path with a constant volumetric flow (that is cooled or dehumidified) from the inlet volume to the discharge volume. Operation of the cooler may be tied to other facets of the calculation by use of a control function to switch the cooler on or off.

2.2 Mechanistic Model

The condensation or evaporation rate of a condensate on a surface is determined by the difference in the partial pressure of the condensable vapor across the gas boundary layer (see Figure 2.1) and can be adversely affected by the presence of noncondensible gases. The formulations of equations for the condensation mass flux across this boundary layer are derived below. Notice that properties characterizing the gas boundary layer ideally would be evaluated at the boundary layer temperature $(T_g + T_{if})/2$, where T_{if} is the gas surface interface temperature of the coil. However, to avoid nested iterations required to solve for T_{if} simultaneously with $T_{c,o}$, the gas properties are evaluated at an average temperature $T_{av} = (T_g + T_c)/2$, where T_g is the gas temperature for the row, and T_c is the coolant temperature for the row.

For each row of the fan cooler, the heat and mass transfer to the row of coils is governed by the heat transfer coefficient h, and the mass transfer coefficient K_g

The convective heat transfer coefficient is defined as

$$h_c = \frac{N_{nu} k_{av}}{d_c} \tag{2-15}$$

where d_C is the cooling coil diameter, k_{av} is the gas thermal conductivity at the average temperature, and the Nusselt number has the form

$$N_{Nu} = 0.33\, Re^{3/5}\, N_{Pr}^{1/3} \tag{2-16}$$

This Nusselt number is a Reynolds-Prandtl correlation for flow over horizontal tubes [4], and is applicable to turbulent flow over tube bundles if there are 10 or more transverse rows in the direction of air flow and the pitch-to-diameter ratios are between 1.25 and 1.5. It should also be noted that this correlation is derived for tube banks without fins. Therefore, the mechanistic model for the fan cooler should be used with caution and the results should, when possible, be compared to published cooler performance data.

The gas properties in the Reynolds number N_{Re} and the Prandtl number N_{Pr} are defined at the average temperature:

$$N_{Re} = \frac{d_c v_g \rho_{av}}{\mu_{av}} \tag{2-17}$$

where v_g is the gas velocity present at the row, ρ_v, is the gas density, and μ_{av} is the gas viscosity. Also,

$$N_{Pr} = \frac{c_{p,av}\mu_{av}}{k_{av}} \tag{2-18}$$

where $c_{p,av}$ the gas specific heat for the row.

The mass transfer coefficient is given by

$$K_g = \frac{N_{Sh}\ D_v\ P_g}{RT_{av}d_c P_{nm}} \tag{2-18}$$

where P_g is the total pressure and $P_{v,b}$ is the vapor partial pressure in the gas, $P_{v,if}$ is the vapor pressure at the atmosphere/film interface, and P_{nm} is the logarithmic mean pressure, given by

$$P_{nm} = \frac{\left[P_{v,if} - P_{v,b}\right]}{ln\left[\frac{P_g - P_{v,b}}{P_g - P_{v,if}}\right]} \tag{2-19}$$

The Sherwood number is correlated as

$$N_{Sh} = 0.33\, Re^{3/5}\, N_{SC}^{1/3} \tag{2-20}$$

and the Schmidt number is defined as

$$N_{Sc} = \frac{\mu_{av}}{\rho_{av} D_v} \tag{2-21}$$

The heat and mass transfer to the coil is calculated from the convective heat transfer coefficient h_C, the mass transfer coefficient K_g, and the coil-side heat transfer coefficient "fchntr," using a linearized approximation for the saturated vapor pressure as a function of temperature. Thus, ρ_g, μ_g, k_g, ν_g, and T_g are updated for the next row.

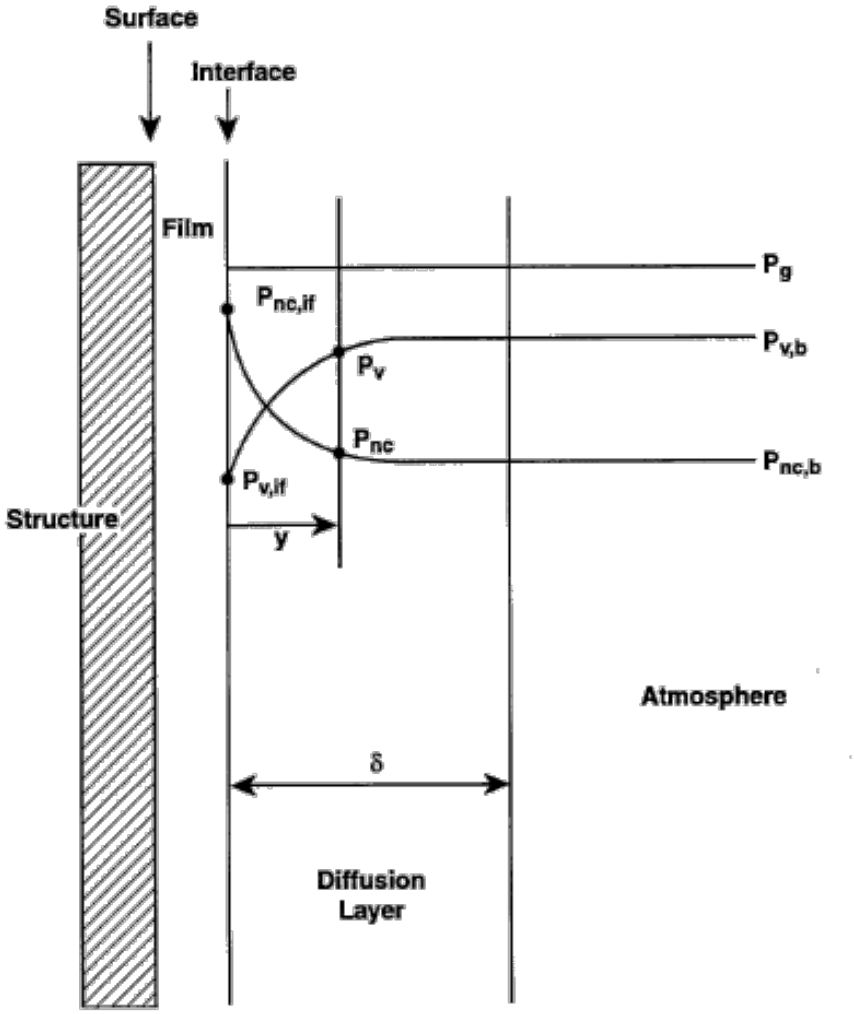


Figure 2.1 Effect of noncondensible gases on the condensation interface

3. Discussion and Development Plans

The simple MARCH model was examined as part of the MELCOR peer review [5]. This review found that use of the Oconee FSAR correlation for the total heat transfer coefficient and the MELCOR approach to partitioning it into a condensation component dependent on water vapor mole fraction and a constant sensible convection component to be deficient because they do not adequately represent the underlying physics. However, this model was deemed relatively unimportant for most PRA applications, since fan coolers either are assumed operational, in which case they have far more capacity than is needed to remove decay heat or are assumed inoperative.

However, for recovery scenarios investigated as part of accident management analyses, errors in calculating condensation rates would impact assessments of the dangers of de-inerting the containment atmosphere and causing burns. Concern was also expressed by the reviewers that the modeling limitations could become important for relatively low-capacity units (e.g., room coolers and non-safety grade fan coolers used for normal heat loads).

Appendix A: Sensitivity Coefficients

This section lists the sensitivity coefficients in the FCL package associated with various correlations and modeling parameters described in this reference manual.

Coefficient	Default Value	Units	EQUIVALENCE name	Equation
9001(1)	590.54	W/m^2-K	HSEN	2.1, 2.3
9001(2)	1.0	-	FSEN	2.3
9001(3)	0.26	-	FMLSCR	2.1, 2.2
9001(4)	3603.4	W/m^2-K	DHLAT1	2.1
9001(5)	2325.25	W/m^2-K	DHLAT2	2.2

References

1 R.O. Wooton, P. Cybulskis, and S.F. Quayle, MARCH 2 (Meltdown Accident Response Characteristics) Code Description and User's Manual, NUREG/CR-3988, BMI-2115 (August 1984).

2 K.K. Murata, D.C. Williams, J. Tills, et al., Code Manual for CONTAIN 2.0: A Computer Code for Nuclear Reactor Containment Analysis, NUREG/CR-6533, SAND97-1735.

3 Duke Power Company, Oconee Nuclear Station Units 1, 2, and 3: Final Safety Analysis Report (1987).

4 J.P. Holman, Heat Transfer, McGraw-Hill Book Co., New York, N.Y., 1968.

5 B. E. Boyack, et al., MELCOR Peer Review, LA-12240, Los Alamos National Laboratory (March 1992).

Fuel Dispersal Interactions (FDI) Package

This document describes in detail the various models incorporated in the Fuel Dispersal Interactions (FDI) package in MELCOR. A FDI sensitivity coefficient used to control the numerical order in which oxygen or steam is used to oxidize DCH metals was introduced in MELCOR 1.8.5. This parameter affects the amount of hydrogen that results from burning DCH materials in steam/oxygen atmospheres.

Details on input to the FDI Package can be found in the FDI Users' Guide.

Contents

1. Introduction

The Fuel Dispersal Interactions (FDI) Package in MELCOR calculates the behavior of debris in containment unless or until it is deposited in a cavity modeled by the MELCOR Cavity (CAV) package. Debris enters the package in basically two ways:

(1). If the Core (COR) package is active, debris enters the FDI package via the Transfer Process (TP) package after failure of the reactor vessel, or

(2). in the stand-alone high-pressure melt ejection (HPME) model, debris enters the FDI package through a user interface, which may be either tabular function input or input from an external data file (EDF) via the TP package.

Two types of phenomena are treated in the FDI package:

(1). low-pressure molten fuel ejection (LPME) from the reactor vessel and

(2). high-pressure molten fuel ejection from the reactor vessel (direct containment heating). There is currently no plan to model steam explosions within or outside the FDI package in MELCOR.

There is no fission product modeling associated with the FDI package, with one minor exception. In particular, there is no release of fission products from fuel debris modeled in the FDI package. In general, the only function performed by the FDI package with respect to radionuclide modeling is inventory transport. That is, if the FDI package transports fuel debris from one location to another, it calls the RadioNuclide (RN) package and instructs it to transport the fission products associated with the fuel debris in exactly the same way.

The one exception to the foregoing concerns decay heat associated with debris deposited on heat structures by the HPME model. The decay heat associated with deposited debris is treated in essentially the same way as the decay heat associated with fission product aerosols and vapors that settle/deposit on heat structures in the RN package modeling. The RadioNuclide Package Reference Manual discusses this modeling in detail. The decay heat associated with airborne debris in the HPME model and all debris during its short residence in the LPME model is ignored; the energy error associated with its omission should be quite small.

2. Detailed Models

2.1 General Information

The FDI package becomes active whenever debris material enters the package. Debris material typically enters the FDI package in one of three ways. In a reactor plant accident calculation, debris enters the FDI package via the TP package interface from the core (COR) package after failure of the reactor pressure vessel has been calculated. In a stand-alone direct containment heating (DCH) calculation, debris material is sourced into the FDI package either directly from tabular function user input or via the TP package interface to a user-provided external data file (EDF) containing the source. The Transfer Process Package Users' Guide and External Data File Package Users' Guide, along with the FDI Package Users' Guide, provide example input to illustrate the interfaces.

After the introduction of debris material, the FDI package classifies the ejection event as either a low- or a high-pressure melt ejection event on the basis of the ejection velocity passed through the TP package or a flag set by the user for stand-alone DCH calculations.

2.2 Low-Pressure Melt Ejection (LPME) Modeling

The heart of the LPME model that has been incorporated into MELCOR was developed by Corradini et al. [1] at the University of Wisconsin. In this model, heat is transferred from the molten debris to the water pool (if present in the associated control volume) as it breaks up and falls to the cavity floor. The heat transfer is normally dominated by radiation, but a lower bound determined by conduction through a vapor film (the Bromley model for film boiling) is also considered. All of the energy transfer from the molten debris is used to boil the pool water (i.e., no sensible heating is considered (e.g., a subcooled pool remains subcooled and its temperature does not change)). The LPME model does not consider oxidation of the metallic elements in the ejected debris. If no pool is present, material passes through FDI without any energy removal. At the cavity floor, the material is normally passed to the CAV package (CORCON) by way of the TP interface.

The first step in the LPME calculational sequence involves retrieving the variables describing the debris state entering the model at the beginning of each calculational cycle (timestep). The debris variables are passed from the COR package to the TP package prior to execution of the FDI package, so the values of the variables are current for the timestep. The variables retrieved from the TP package by the FDI package include the mass, composition and temperature of the debris ejected from the vessel during the timestep and the velocity and diameter of the ejection stream (see COR reference manual for a description of the calculation of these variables).

The second step in the LPME sequence is to determine the axial position of the head and tail of the ejected debris with respect to the FDI calculational volume. The user specifies

z_{TOP} and z_{BOT}, the elevation of the top and bottom of the calculational volume, respectively, (which typically are equal to the elevation of the bottom of the reactor vessel and the bottom of the reactor cavity). Then the positions of the head and tail of the ejected debris and its length are given by

$$z_{HEAD} = z_{TOP} - U_f \Delta t \quad (2\text{-}1)$$

$$z_{TAIL} = z_{HEAD} + m_f/(\pi D_{f0}^2 \rho_f/4) \quad (2\text{-}2)$$

$$z_{LEN} = z_{TAIL} - z_{HEAD} \quad (2\text{-}3)$$

where U_f, m_f, D_{f0} *and* ρ_f are the velocity, mass, initial diameter (determined by the COR package and equal to the diameter of reactor vessel breach, which may increase if hole ablation occurs) and density of the ejection stream, respectively. Any debris below elevation z_{BOT} accumulates on the cavity floor, and its mass is designated m_{FLR} and given by

$$m_{FLR} = m_f MAX\{0, MIN[1, (z_{BOT} - z_{HEAD})/z_{LEN}]\} \quad (2\text{-}4)$$

The portion of m_f that does not reach the floor remains in the FDI calculational volume until the next timestep and is designated m_{CAV}. If there was already mass in the volume (m_{CAV0} from the previous timestep), then it is added to m_{FLR} and deposited on the cavity floor on this timestep. If m_f is zero (i.e., if mass ejection from the vessel has ceased), then any pre-existing m_{CAV0} is transferred to m_{FLR}. If m_f is greater than zero, but z_{HEAD} is greater than z_{BOT}, then only pre-existing m_{CAV0} is deposited on the floor and given by

$$m_{FLR} = MIN(m_f, m_{CAV0}) \quad (2\text{-}5)$$

$$m_{CAV} = m_f + m_{CAV0} - m_{FLR} \quad (2\text{-}6)$$

In effect, this means that if mass is being ejected from the vessel but the timestep is too short for newly ejected debris to reach the cavity floor, then pre-existing debris that has not reached the cavity floor is deposited on the cavity floor at a rate equal to the vessel ejection rate, and the newly ejected debris takes the place of the deposited debris. However, as soon as vessel ejection ceases, then all remaining debris that has not reached the cavity floor is immediately deposited in the cavity floor in a single timestep.

After the mass of debris reaching the floor during the current timestep has been determined, heat transfer to water in the cavity is evaluated. Although the heat transfer occurs during the passage of the debris through the cavity pool, the actual heat transfer associated with a given packet of debris is not transferred to the pool until that packet is

deposited on the cavity floor. Debris that does not reach the floor during the current timestep does not participate in heat transfer to the water until a later timestep.

The rate of heat transfer from the debris to the water is determined primarily by the interfacial surface area, which is a function of the debris particle size. The particle size for molten debris particles descending through the cavity pool is given by a modified theoretical correlation for droplet breakup under hydrodynamic force. The original correlation as formulated by Chu [2] for a water/air system is

$$D_f(t) = D_{f0}\, exp(-C_1 \tau^{0.772} We^{0.246}) \quad (2\text{-}7)$$

We is the Weber number, which is defined by

$$We = \rho_c U_f^2 D_{f0}/\sigma_f \quad (2\text{-}8)$$

where ρ_c is the coolant density, $U_f, D_{f0}, and\ \sigma_f$ are the velocity, initial diameter and surface tension of the droplets, respectively. τ is the dimensionless time, which is defined by

$$\tau = (U_f t/D_{f0})(\rho_c/\rho_f)^{1/2} \quad (2\text{-}9)$$

where the time of descent, *t*, is zero when the debris is at the pool surface and increases as the debris descends through the pool. Constant C_1 is taken to be

$$C_1 = 0.171 - 0.149(\rho_c/\rho_f)^{1/2} \quad (2\text{-}10)$$

To provide an easily integrable form for analytic use in MELCOR, Chu's correlation is modified as follows

$$D_f(t) = D_{f0}\, exp(-C_1\ \tau\ We^{0.254}) \quad (2\text{-}11)$$

with constant C_1 taken to be

$$C_1 = 0.1232 - 0.149(\rho_c/\rho_f)^{1/2} \quad (2\text{-}12)$$

A comparison of Chu's correlation to this modified correlation for the water/air and corium/water systems reveals reasonable agreement [1]. Assuming constant velocity, U_f, Equation (2-11) can be converted to a function of the elevation of the pool surface, z_{POOL}, as shown below

$$D_f(z) = D_{f0}\, exp(-Z) \quad (2\text{-}13)$$

for $z_{BOT} \le z \le z_{POOL}$ where the variable Z is

$$Z = (C_1 We^{0.254}/D_{f0})(\rho_c/\rho_f)^{1/2}(z_{POOL} - z) \quad (2\text{-}14)$$

Equation (2-13) is valid only as long as the debris remains molten. After the debris solidifies, ($T_f < T_{sol,f}$ as determined by the solution of Equation (2-17) to follow), there is no further breakup, and the heat transfer area is constant.

Another important factor affecting the rate of heat transfer is the heat transfer regime. In the early stage of heat transfer from the debris, the debris temperature is very high; hence, radiation heat transfer would be the dominant heat transfer mechanism. As the debris temperature falls, eventually other mechanisms become important.

Although radiation and conduction through the vapor film occur in parallel, the model incorporated into MELCOR only considers the dominant mechanism at any given time. Hence, the model switches from radiation-dominated to conduction-dominated film boiling heat transfer when the debris temperature falls below the "regime transition temperature". The regime transition temperature, T_{TRAN}, is defined as the temperature at which the net radiation heat flux between the debris and pool is equal to the conduction-dominated film boiling heat flux from the debris to the pool and is given by the solution to the following equation

$$\sigma({T_{TRAN}}^4 - T_c^4) = h_{FB}(T_{TRAN} - T_{sat}) \quad (2\text{-}15)$$

where σ is the Stefan-Boltzmann constant (and the emissivity is assumed to be unity) and h_{FB} is the conduction-dominated film boiling heat transfer coefficient given by Bromley [3]

$$h_{FB} = (1/2)\ k_g\{k_g\mu_g D_f(T_g - T_{sat})/[\rho_g(\rho_c - \rho_g)\ g\ i_{fg}]\}^{-1/4} \quad (2\text{-}16)$$

where k_g and μ_g are the thermal conductivity and viscosity of the vapor film, respectively, i_{fg} is the latent heat of vaporization of water and g is the acceleration of gravity. To derive this equation, it was assumed that the vapor saturation temperature, T_{sat}, the debris diameter, D_f, and the vapor temperature, T_g, are constant. T_g is the arithmetic average of the debris and saturation temperatures. Equation (2-15) can be solved iteratively to yield T_{TRAN}, the heat transfer regime transition temperature.

The rate of change of the debris temperature, T_f, is given by

$$\rho_f c_{vf}(\pi D_f^3/6)U_f dT_f/dz = q_{f-c}(\pi D_f^2) \quad (2\text{-}17)$$

where c_{vf} is the specific heat capacity of the debris and q_{f-c} is the heat flux from the debris to the coolant. For $T_f > T_{sol,f}$, D_f is given by Equation (2-13); otherwise D_f remains equal to its value at the instant solidification begins. For $T_f > T_{TRAN}$, q_{f-c} is calculated assuming only radiative heat transfer; otherwise, q_{f-c} is calculated assuming only transition film boiling. Equation (2-17) can be integrated from $z = z_{POOL}$ to $z = z_{BOT}$ to yield $T_{f,BOT}$, the debris temperature at the bottom of the coolant pool.

Once the debris temperature at the bottom of the pool is known, the total amount of heat transferred to the pool is given by

$$Q_{f-c} = H_f(T_{f0}) - H_f(T_{f,BOT}) \tag{2-18}$$

where

$$H_f(T) = \sum_{i=1}^{NMAT} \left[m_{i,FLR} h_i(T) \right] \tag{2-19}$$

and $h_i(T)$ is the specific enthalpy of debris component *i* at temperature *T*. The mass of steam generated by the heat transfer is given by

$$m_{STEAM} = Q_{f-c}/(h_{SCV} - h_{POOL}) \tag{2-20}$$

where h_{SCV} is the specific enthalpy of saturated steam at the total pressure in the FDI control volume and h_{POOL} is the specific enthalpy of the water in the cavity pool. Note that all heat transfer is assumed to generate steam; hence, the pool temperature should not change. If m_{STEAM} exceeds the mass of coolant in the pool, then $H_f(T_{f,BOT})$ and $T_{f,BOT}$ are back-calculated to provide just enough heat transfer to vaporize the mass of coolant in the pool.

Following the calculation of steam generation, the increments to the pool and vapor masses and energies are passed to the CVH package, the debris deposited on the floor at temperature $T_{f,BOT}$ is passed to the CAV package, where core-concrete interactions are modeled, and the radionuclides associated with the debris passed to CAV are transferred from FDI to the radionuclide package.

2.3 High-Pressure Melt Ejection (HPME) Modeling

If the velocity of the molten debris ejected from the reactor vessel exceeds a critical value prescribed by sensitivity coefficient 4602 (with a default value of 10 m/s), or if the user

has invoked the stand-alone option for high-pressure melt ejection modeling, then the FDI is treated by the high-pressure model instead of the low-pressure model.

The parametric high-pressure model requires user input to control both the distribution of debris throughout the containment and the interaction of the hot debris with the containment atmosphere. The processes modeled include oxidation of the metallic components of the debris (Zircaloy, aluminum and steel) in both steam and oxygen, surface deposition of the airborne debris by trapping or settling and heat transfer to the atmosphere and deposition surfaces.

The HPME model does not include a mechanistic debris transport model; rather, the user specifies a set of debris destinations with a corresponding set of transport fractions that prescribe where the ejected debris is assumed to go. The debris destinations may include the atmosphere of any CVH control volume, the surface of any heat structure and cavities defined by the CAV package. The sum of the transport fractions over all the specified control volume atmospheres, heat structure surfaces and cavities must equal one. Transport of the ejected debris to its assumed destinations occurs instantaneously, with no interactions occurring between the point of ejection and the destination sites. As long as the HPME model is active (i.e., as long as the ejection velocity exceeds the LPME/HPME transition velocity prescribed by sensitivity coefficient 4602 or if the user has invoked the stand-alone HPME model), the ejected debris is partitioned among the destinations as specified by the transport fractions. When the ejection velocity falls below the LPME/HPME transition velocity for non-stand-alone applications, any debris subsequently ejected is passed to the LPME model, which uses LPME model input instead of the HPME transport model to determine the debris destination. However, debris that was transported to the HPME debris destinations before the model transition occurred continues to be treated by the HPME model.

Debris that is transported to cavity destinations is not treated further by the FDI package; rather, subsequent treatment is provided by the CAV package. As implemented in the HPME model, surface deposition of debris can occur in two distinct ways. Ejected debris that impacts structures prior to any significant interaction with the atmosphere is sourced directly to the destination surface via the user-specified transport fraction for that surface. This process is referred to as trapping in MELCOR. Alternatively, debris that interacts significantly with the atmosphere should be sourced to the appropriate control volume, in which a user-specified settling time constant determines the rate of deposition to the specified settling destination (either a heat structure surface or a cavity). This process is referred to as settling in MELCOR.

First-order rate equations with user-specified time constants for oxidation, heat transfer and settling are used to determine the rate of each process. Oxidation of airborne and deposited debris is only calculated if the debris temperature exceeds a minimum value, TOXMIN, which is adjustable through sensitivity coefficient 4609 and has a default value of 600 K. If a pool of water exists in the reactor cavity at the time of debris ejection, then the water is ejected into the droplet field (fog) of the atmosphere at a rate proportional to

the rate of injection of the debris into the pool. The proportionality constant is adjustable through sensitivity coefficient 4605 and has a default value of 10. This proportionality constant is strictly parametric and intended for exploratory purposes only. The rate of dispersal of the cavity water may be very important in determining containment loads, if interaction between the debris and cavity water is a primary contributor to the load. Excessive values of this coefficient may disperse the cavity water prematurely and limit subsequent interactions between ejected debris and cavity water, while deficient values excessively limit the overall interaction of debris and water. Consequently, it is strongly recommended that the effects of variations in the value of this sensitivity coefficient be examined both because of its inherent uncertainty, and because of the large impact it may have on containment loads. The HPME model does not consider any thermal interaction between the ejected debris and the water in the cavity pool such as that described above for the LPME model.

When the HPME model first initiates direct containment heating in a control volume, the FDI package requests a fallback of the cycle if the timestep exceeds the recommended start-up value prescribed by sensitivity coefficient 4607 (with a default value of 10^{-4}s). The start-up value should be reasonably small both to avoid numerical problems associated with excessive energy transfers to the CVH atmosphere per timestep and to capture the detail associated with the HPME phenomena, which occurs on a time scale comparable to the user-specified time constants for the phenomena. Experience has indicated that for most realistic scenarios, the rapid excursions in pressure and temperature caused by direct containment heating dictate the use of very small timesteps for several cycles following DCH initiation. See the input record EXEC_SOFTDTMIN in the Executive Package Users' Guide for help with this requirement.

The airborne masses of UO_2 and other materials that neither oxidize nor are the products of oxidation are described by the following first-order linear differential equation:

$$\frac{dm_{i,k}(t)}{dt} = -\frac{m_{i,k}(t)}{\tau_{ST,i}} + S_{i,k} \qquad (2\text{-}21)$$

where $m_{i,k}(t)$ is the mass of component *k* in control volume *i* at time *t*, $\tau_{ST,i}$ is the time constant for settling in control volume *i* and $S_{i,k}$ is the constant mass source rate of component *k* in control volume *i* associated with the high-pressure melt ejection process. The solution of Equation (2-21) is given by

$$m_{i,k}(t) = \left[m_{i,k}(t_0) - S_{i,k}\tau_{ST,i}\right] exp(\frac{dt}{\tau_{ST,i}}) + S_{i,k}\tau_{ST,i} \qquad (2\text{-}22)$$

where $m_{i,k}(t_0)$ is the mass at arbitrary initial time t_0 and *dt* is the difference between the final time, *t*, and time t_0.

The airborne masses of Zr, Al and steel (the only materials that are oxidized in the presence of oxygen or steam) are described by the following first-order linear differential equation:

$$\frac{dm_{i,k}(t)}{dt} = - \frac{m_{i,k}(t)}{\tau_{SO,i}} + S_{i,k} \quad (2\text{-}23)$$

where $\tau_{SO,i}$, the time constant for simultaneous oxidation and settling/trapping, is given by

$$\tau_{SO,i}^{-1} = \tau_{ST,i}^{-1} + \tau_{OX,i}^{-1} \quad (2\text{-}24)$$

and where $\tau_{OX,i}$, is the oxidation time constant in control volume *i*. The solution to Equation (2-23) is identical to Equation (2-22) except that $\tau_{ST,i}$ is replaced by $\tau_{SO,i}$.

The airborne masses of ZrO_2 and other materials that are products of oxidation reactions are given by

$$\frac{dm_{i,k}(t)}{dt} = -\frac{m_{i,k}(t)}{\tau_{ST,i}} + R\frac{m_{i,l}(t)}{\tau_{OX,i}} + S_{i,k} \quad (2\text{-}25)$$

where *R* is the mass of product *k* formed by the oxidation of a unit mass of reactant *l*. The solution of Equation (2-25) is

$$m_{i,k}(t) = \left[m_{i,k}(t_0) - C_1 - C_2\right] exp(-dt/\tau_{ST,i}) + C_2\, exp(-dt/\tau_{SO,i}) + C_1 \quad (2\text{-}26)$$

where

$$C_1 = (S_{i,k} + RS_{i,l}\tau_{SO,i}/\tau_{OX,i})\tau_{ST,i}$$

and

$$C_2 = R\left[S_{i,l}\tau_{SO,i} - m_{i,l}(t_0)\right]$$

The HPME model contains two options for oxidation modeling. These may be selected independently for each control volume. The first is the sequential oxidation option, in which the order of oxidation is Zr, Al, then steel (typical metallic elements associated with reactor cores and/or simulation experiments). This is invoked by specifying a positive value for the oxidation time constant, $\tau_{OX,i}$. The second option is simultaneous oxidation of the metals, which is invoked by specifying a negative value of $\tau_{OX,i}$, in which case the time constant is equal to the absolute value of $\tau_{OX,i}$. Under normal conditions where the metallic constituents exist in a more or less well-mixed state, the sequential oxidation

option is recommended because it is more realistic. Elements with higher oxidation potentials tend to be preferentially oxidized unless some kinetic limitation exists.

In the sequential oxidation model, a separate oxidation rate is first calculated for each metal independently of all others, with the given value of $\tau_{OX,i}$. Then the mass of metal B consumed is converted into an equivalent mass of metal A, where metal A is assumed to oxidize in preference to metal B, until all of metal A is consumed. Hence, steel (and Inconel, which is included in the steel mass in the FDI package) is not consumed until all the Zr and Al have been consumed, and Al is not consumed until the Zr is exhausted. This implies that the effective time constant for metal A oxidation when metal B is present may be significantly shorter than $\tau_{OX,i}$. The actual values of the effective oxidation time constants are used to determine the end of timestep airborne mass inventories in Equation (2-23) and Equation (2-26) above.

Both oxidation options are constrained by the availability of oxygen or steam. Oxidation is apportioned between steam and oxygen by their relative mole fractions in the atmosphere. This change was invoked in MELCOR 1.8.5; previously, the oxygen would react first, followed by steam only after the oxygen had been consumed. Although this assumption probably reflects the relative oxidation potential of oxygen versus steam, it does not consider diffusion transport in the atmosphere and generally resulted in insufficient hydrogen generation during DCH. The relative oxidation effectiveness of oxygen versus steam can be adjusted through sensitivity coefficient 4610. The oxidation ratio is proportional to the moles of oxygen divided by the moles of oxygen plus steam:

$$R_{OX} = \frac{WX_{O_2}}{WX_{O_2} + X_{st}} \qquad (2\text{-}27)$$

where W is the weighting factor given by sensitivity coefficient 4610, X_{O_2} is the moles of oxygen, and X_{st} is the moles of steam in the atmosphere. Making W a large number weights the oxygen moles and give the previous “oxygen first” behavior. If there is insufficient oxidant to support the calculated rates of oxidation for zirconium and iron, then the zirconium has priority. The oxidation reactions proceed at the start of timestep values of debris temperature in each control volume, and any hydrogen formed by the steam reaction enters the atmosphere at that temperature.

The temperature of the airborne debris is affected by debris sources, oxidation and heat transfer from the debris to the atmosphere. The temperature of the atmosphere, T_{gas}, is assumed to remain constant and equal to the beginning of timestep value obtained from the CVH package database. This explicit coupling between FDI and CVH may limit the timestep size during energetic transients, as discussed below. The enthalpy of the airborne debris is given by the solution of the following simple equation:

$$\frac{dH_i(t)}{dt} = \dot{E}_{ox,i}(t) - \dot{Q}_{GAS,i}(t) + S_{H,i} \quad (2\text{-}28)$$

where $\dot{E}_{OX,i}(t)$ is the rate of heat generation by the oxidation reaction, $S_{H,i}$ is the enthalpy source rate associated with the HPME source, and the rate of heat transfer to the gas is approximated as:

$$\dot{Q}_{GAS,i}(t) = \frac{Q_g i(t)}{\tau_{HT,i}} = \frac{H_i(T_{dbr}) - H_i(T_{gas})}{\tau_{HT,i}} \quad (2\text{-}29)$$

where $Q_{g,i}(t)$ is the enthalpy available for transfer to the gas, $\tau_{HT,i}$ is the user-specified time constant for heat transfer from the airborne debris to the atmosphere in control volume *i*, $H_i(T_{dbr})$ is the enthalpy content of the debris at its actual temperature, T_{dbr}, and $H_i(T_{gas})$ is the enthalpy content of the debris in equilibrium with the gas at temperature T_{gas}. The solution to Equation (2-28) is given by:

$$H_i(t) = H_i(t_0) + E_{OX,i}(t) - Q_{GAS,i}(t) \quad (2\text{-}30)$$

where $H_i(t_0)$ is the enthalpy of the debris after the addition of the integrated enthalpy source $S_{H,i}dt$ and after adjustments to its composition associated with the oxidation reactions, where $E_{OX,i}(t)$ is the oxidation enthalpy generated between times t_0 and t, and where $Q_{GAS,i}(t)$ is the amount of heat transferred to the gas between times t_0 and t. $Q_{GAS,i}(t)$ is given by

$$Q_{GAS,i}(t) = \int_{t_0}^{t_0+dt} \left[Q_{g,i}(t)/\tau_{HT,i}\right]\, dt \quad (2\text{-}31)$$

where the available enthalpy $Q_{g,i}(t)$ increases as a result of oxidation and the addition of high-temperature debris source material and decreases as enthalpy is transferred to the gas. $Q_{g,i}(t)$ satisfies

$$\frac{dQ_{g,i}(t)}{dt} = -\frac{Q_{g,i}(t)}{\tau_{HT,i}} + \left[Q_{SRC,i}/dt + \left(E_{OX,i} + Q_{OX,i}\right)/dt\right] \quad (2\text{-}32)$$

$$Q_{SRC,i} = S_{H,i}(T_{src}) - S_{H,i}(T_{gas})$$

is the available source enthalpy and

$$Q_{OX,i} = H_{OX,i}(T_{dbr}) - H_{OX,i}(T_{gas})$$

is the available enthalpy created by composition adjustments during oxidation. The solution to Equations (2-31) and (2-32) is:

$$Q_{GAS,i}(t) = Q_{OLD,i}\left[1 - exp\left(-dt/\tau_{HT,i}\right)\right] + \left[Q_{SRC,i}/dt + \left(E_{OX,i} + Q_{OX,i}\right)/dt\right] \left\{dt - \tau_{HT,i}\left[1 - exp\left(-dt/\tau_{HT,i}\right)\right]\right\} \quad (2\text{-}33)$$

where $Q_{OLD,i} = H_i\left(T(t_0)\right) - H_i\left(T_{gas}\right)$ is the initial available enthalpy.

The inclusion of the HPME source terms in Equations (2-21) through (2-33) reduces some timestep dependencies that would arise if the sources were added prior to the calculation of oxidation, heat transfer and settling/trapping. After the total enthalpy at the advanced time, t, is determined, it is compared to the enthalpy corresponding to a maximum permissible temperature, H_{MAX}. If $H_i(t)$ exceeds H_{MAX}, then Equation (2-30) is solved for $Q_{GAS,i}(t)$ with $H_i(t)$ set equal to H_{MAX} as follows:

$$Q_{GAS,i}(t) = H_i(t_0) + E_{OX,i}(t) - H_{MAX} \quad (2\text{-}34)$$

so that the heat transferred to the gas is increased sufficiently to limit the advanced time debris temperature to the maximum prescribed value, T_{MAX}. T_{MAX} is given by

$$T_{MAX} = MAX\left(T_{gas}, T_{dbr}(t_0), T_{dbr}(t^{'}),\ T4603\right) \quad (2\text{-}35)$$

where T_{gas} is the gas temperature, $T_{dbr}(t_0)$ is the debris temperature at the beginning of the timestep, $T_{dbr}(t')$ the debris temperature after addition of new source material to the initial inventory and C4603 is the temperature limit prescribed by sensitivity coefficient C4603, which normally exceeds the other arguments in the max function of Equation (2-35). The default value of C4603 is approximately equal to the boiling temperature of UO_2—temperatures much in excess of this value would likely result in very rapid fragmentation of debris droplets and significantly increased droplet-to-gas heat transfer.

After an advanced time and temperature for the airborne debris have been determined, the projected change in the CVH atmosphere temperature as a result of direct containment heating during the timestep is calculated. If the change exceeds a value prescribed by sensitivity coefficient C4604 (with a default value of 500 K), then the FDI package requests a fallback with a decreased timestep. This feature provides control over numerical problems associated with excessive energy transfers to the CVH atmosphere. If the value of sensitivity coefficient 4604 is set too high, it is possible that the CVH package encounters numerical difficulties that cannot be resolved by CVH fallbacks. In practice, the default value was found to prevent numerical problems in CVH without excessively limiting the timestep.

Following the determination of the advanced time temperature for the airborne debris, the advanced time mass equations, Equations (2-21) through (2-26), are used to determine how much material is removed from the atmosphere by settling/trapping. The settled material and its energy content are removed from the airborne inventory and deposited on the appropriate surface specified by user input. After the settling calculation has been performed, the advanced time total airborne mass in each control volume is determined by summing over all components. If the advanced time total airborne mass is insignificant compared to the total mass of material sourced into the control volume atmosphere over the duration of the DCH event, then all of the remaining airborne mass in the control volume is immediately deposited on the appropriate settling surface and a message is issued to notify the user that direct containment heating has ceased in that particular control volume. The ratio used to determine when the airborne mass has become insignificant is adjustable through sensitivity coefficient 4606 and has a default value of 0.001. This implies that only 0.1 percent of the mass source is prematurely deposited, which was judged to be a reasonable compromise between the demands of accuracy and calculational effort.

Deposited Debris

The mass of material *k* on surface *i* at time *t* is given by

$$m_{i,k}(t) = m_{i,k}(t_0) + S'_{i,k}dt \tag{2-36}$$

where

$$S'_{i,k} = S_{i,k} + \sum_j \int_{t_0}^{t_0+dt} \left[m_{j,k}(t')/\tau_{ST,j}\right] dt'/dt \tag{2-37}$$

and $S_{i,k}$ is the constant mass source rate of component *k* to surface *i* from trapping. The second term on the right-hand side of Equation (2-37) accounts for settling to the surface, where the sum is over all control volumes that have surface *i* as the user-specified settling surface, and $m_{j,k}(t)$ and $\tau_{ST,j}$ are the airborne mass of component *k* in control volume *j* and the settling time constant in control volume *j*, respectively.

For UO_2 and other materials not associated with oxidation, the settling term is given by

$$\begin{aligned}\int_{t_0}^{t_0+dt} m_{j,k}(t')dt'/\tau_{ST,j} &= m_{j,\ell}(t_0)\left[1 - exp(-dt/\tau_{ST,j})\right] \\ &+ S_{j,k}\left\{dt - \tau_{ST,j}\left[1 - exp(-dt/\tau_{ST,j})\right]\right\}\end{aligned} \tag{2-38}$$

For metals that oxidize, the settling term is given by

$$\int_{t_0}^{t_0+dt} m_{j,k}(t')dt'/\tau_{ST,j} = (\tau_{SO,j}/\tau_{ST,j})\big(m_{j,k}(t_0)\big[1-exp(-dt/\tau_{SO,j})\big] + S_{j,k}\big\{dt-\tau_{SO,j}\big[1-exp(-dt/\tau_{SO,j})\big]\big\}\big) \quad (2\text{-}39)$$

which reduces to Equation (2-38) if $\tau_{OX,i} >> \tau_{ST,j}$, because in that case $\tau_{SO,j} \approx \tau_{ST,j}$ as shown by Equation (2-24). For oxidation products, the settling term is given by

$$\begin{aligned}\int_{t_0}^{t_0+dt} m_{j,k}(t')dt'/\tau_{ST,j} &= m_{j,k}(t_0)\big[1-\exp(-dt/\tau_{ST,j})\big]+S_{j,k}\{dt-\tau_{ST,j}\\ &[1-\exp(-dt/\tau_{ST,j})]\}+R\{(\tau_{SO,j}/\tau_{OX,j})[m_{j,l}(t_0)+S_{j,l}(dt-\tau_{SO,j}\\ &-\tau_{ST,j})]-[m_{j,l}(t_0)-S_{j,l}\tau_{ST,j}]\exp(-dt/\tau_{ST,j})+([\tau_{SO,j}/\tau_{ST,j}]\\ &[m_{j,l}(t_0)-S_{j,l}\tau_{SO,j}]\exp(-dt/\tau_{SO,j})\}\end{aligned} \quad (2\text{-}40)$$

where material *l* is the metal from which the oxide is formed and *R* is the mass of product *k* formed by the oxidation of a unit mass of material *l*.

The energy of the deposited debris is calculated with equations almost identical to Equations (2-28) through (2-35) except the source term $S_{H,i}$ also includes the enthalpy associated with debris settling. It is assumed that the enthalpy of the settled debris is equal to the end of timestep value calculated with Equation (2-30). The settled mass with the end of step enthalpy is applied to the deposition surface during the timestep at a constant rate as implied by Equation (2-36). The other difference between the treatment of the energy of airborne and deposited debris concerns heat transfer. As discussed above, the user specifies a time constant for heat transfer from the airborne debris to the atmosphere. However, for heat transfer from deposited debris to the structure, a different approach is taken. Because the CVH package does not recognize the deposited debris temperature as the effective surface temperature, in order to effectively simulate the heat transfer from the hot debris to the CVH pool and/or atmosphere associated with the surface, it is necessary to couple the debris temperature tightly to the HS surface temperature that CVH does recognize.

The debris temperature and HS surface temperature are tightly coupled if the effective heat transfer coefficient from the debris to the surface, h_{SRF}, is large compared to the heat transfer from the surface to the first interior node in the structure, which is given by $k_{HS,1}/\Delta x_{HS,1}$ (structure thermal conductivity divided by the node thickness). In order to generate a large value of h_{SRF}, a very small time constant equal to the minimum of half the surface oxidation time constant and a value of 0.001 s is used to calculate the amount of heat transfer from the debris to the deposition surface using the analog of Equation (2-33) for heat transfer to surfaces. The value obtained is then used to determine h_{SRF} as follows:

$$h_{SRF} = Q_{SRF,i}(t)/(A_{SRF}\Delta T dt) \quad (2\text{-}41)$$

where A_{SRF} is the surface area of the structure, ΔT is the difference between the beginning of timestep debris temperature and the structure surface temperature and Q_{SRF} is the value obtained from the analog of Equation (2-33). This value almost always exceeds the value of $k_{HS,1} / \Delta x_{HS,1}$. In fact, the value of h_{SRF} may be large enough to induce oscillations in the structure surface temperature because of the explicit coupling between FDI and HS packages. Therefore, a limit is placed on the value of h_{SRF}. If h_{SRF} exceeds a maximum value, $h_{SRF,\max}$, specified by sensitivity coefficient 4608 (default value 1000. W/m^2-K), then the value of Q_{SRF} is reduced by the ratio $h_{SRF,\max} / h_{SRF}$ to limit it to the value consistent with $h_{SRF,\max}$. Whenever, the Q_{SRF} is limited by $h_{SRF,\max}$ the direction of heat transfer (i.e., debris-to-surface or surface-to-debris) is compared to the direction from the previous timestep. If the direction is alternating, this probably indicates that the surface temperature has been driven into an oscillation about the debris temperature because the timestep exceeds the stability limit associated with the explicit coupling between the FDI and HS packages. In such cases, FDI requests a system fallback with the timestep reduced by a factor of one half. Normally, the value of $h_{SRF,\max}$ should be chosen large enough to promote rapid equilibration of the debris and surface temperatures, yet not so large as to induce instability in the surface temperature for reasonable values of the timestep. Users should refer to the HS Reference Manual for a further discussion of stability/accuracy concerns associated with structure nodalization and timestep size.

If the MELCOR RadioNuclide (RN) package is active, then FDI calls RN1 of the RN Package anytime fuel is moved so that the associated radionuclides can be moved simultaneously. Furthermore, the decay heat associated with the radionuclides is deposited in the appropriate location.

3. Sensitivity Coefficients

For convenient reference, the sensitivity coefficients for the FDI package are summarized below, taken from the FDI Reference manual.

Sensitivity Coefficients	Definition
4602	Vessel ejection velocity at transition between high- and low-pressure ejection modeling.
	(default = 10., units = m/s, equiv = none)
4603	Airborne debris temperature above which oxidation energy is deposited directly in the atmosphere—approximate vaporization point.
	(default = 3700., units = K, equiv = none)
4604	Maximum change in the temperature of the CVH atmosphere permitted without a timestep cut.
	(default = 500., units = K, equiv = none)
4605	Ratio of the mass of water ejected from a pool into the reactor cavity atmosphere to the mass of the debris injected from the vessel into the cavity pool.
	(default = 10., units = none, equiv = none)
4606	Ratio of the current airborne debris mass to the integrated airborne debris mass source in a control volume below which the mass is deposited onto the settling surface associated with the control volume—deactivates DCH when the remaining airborne mass becomes insignificant.
	(default = 0.001, units = none, equiv = none)
4607	Initial timestep size for HPME initiation.
	(default = 0.0001, units = s, equiv = DTHPME)
4608	Maximum debris-to-surface heat transfer coefficient.
	(default = 1000., units = W/m^2-K, equiv = HTCMAX)
4609	Minimum temperature for oxidation.
	(default = 600., units = K, equiv = TOXMIN)
4610	Oxygen-steam oxidation weighting factor.
	(default = 1., units = none, equiv = WGTO2)
4620	Convergence criteria for the FDI equation of state.
	(1) Tolerance (relative) for enthalpy. (default = 1.0E-06, units = dimensionless, equiv = TOLENH) (2) Tolerance (relative) for temperature. (default = 2.0E-07, units = dimensionless, equiv = TOLTMP)

4. Discussion and Development Plans

The simple direct containment heating model described above in Section 2.3 is not intended to predict all details of DCH events from first principles. Nodalization requirements would be much greater than normal MELCOR models. Rather, it is intended to allow users to evaluate the overall effect of varying the relative rates of the most important processes controlling DCH loads.

HPME model results are sensitive to the relative values of $\tau_{OX,i}, \tau_{HT,i}$ and $\tau_{ST,i}$ specified by the user for each control volume. Reasonable values for these time constants can be obtained in basically two ways. First, results from detailed codes such as CONTAIN can be used to obtain appropriate values; or, second, reasonable assumptions concerning particle sizes and velocities in conjunction with simplified hand calculations can yield a range of time constants in the correct range. In most cases, this second method should be adequate for parametric PRA studies. Specified time constants of less than 10^{-6} s are reset to that value to avoid potential numerical problems associated with vanishing time constants. For time scales of interest, a time constant of 10^{-6} s implies an essentially instantaneous process (i.e., instantaneous complete oxidation, instantaneous thermal equilibration with the atmosphere or instantaneous settling).

Users are cautioned that the absence of mechanistic debris transport in the HPME model currently limits the scope of phenomena that may be investigated. Specifically, decoupling the debris transport from the vessel blowdown precludes accurately investigating effects associated with the coherence between the debris and steam ejection. If the severity of the DCH threat is primarily limited by the amount of thermal and chemical energy available in the ejected debris, then the model should prove useful. However, if the threat is primarily limited by the amount of steam that has an opportunity to interact with the airborne debris, then the model may fail to capture the important phenomena and can underpredict the DCH load. The user should suspect that this condition may exist whenever the following two conditions hold:

(1). Most of the debris is specified to <u>not</u> reach the main volume of the containment.

(2). In the cavity and/or subcompartment volumes that are specified to receive most of the debris, maximum gas temperatures approach the initial debris temperature and/or oxidant concentrations (O_2 + H_2O) fall to low levels during the time period that airborne debris concentrations are relatively high.

References

1. M. H. Kim, M. D. Oh, G. A. Moses, and M. L. Corradini, “MELCOR Fuel Dispersion Interactions Model,” University of Wisconsin, Reactor Safety Research, Nuclear Engineering Department document (November 1986).

2 . C. C. Chu, M. L. Corradini, “One-Dimensional Transient Model for Fuel-Coolant Fragmentation and Mixing,” Proc. Int’l Mtg. on Thermal Reactor Safety, San Diego CA (February 1986).

3 . L. A. Bromley, “Heat Transfer in Stable Film Boiling,” Chemical Engineering Progress, Vol. 46, No. 5, pp. 221-227, 1950.

Heat Structure (HS) Package

The MELCOR Heat Structure (HS) package calculates heat conduction within an intact, solid structure and energy transfer across its boundary surfaces. The modeling capabilities of heat structures are general and can include pressure vessel internals and walls, containment structures and walls, fuel rods with nuclear or electrical heating, steam generator tubes, piping walls, etc.

This document provides detailed information about the models, solution methods, and timestep control that are utilized by the HS package. Section 1 is an introduction to heat structure modeling and the calculation procedure. Section 2 provides details on the heat and mass transfer models. The solution methods utilized are discussed in Section 3, and timestep control is summarized in Section 4.

Information that is necessary to execute the HS package with other packages in the MELCOR code is found in the HS Users' Guide.

Contents

List of Figures

List of Tables

1. Introduction

The Heat Structure (HS) package calculates heat conduction within an intact solid structure and energy transfer across its boundary surfaces into control volumes. This document is the reference manual for the HS package. It contains the following information for this package:

(1) detailed models,

(2) solution methods, and

(3) timestep control.

This section describes the modeling of a heat structure in the MELCOR code and provides a discussion of the calculation procedure that is used to obtain the temperature distribution and energy transfer for each heat structure and the calculation procedure for its interactions with other packages.

A heat structure is an intact solid structure that is represented by one-dimensional heat conduction with specified boundary conditions at each of its two boundary surfaces. The modeling capabilities of heat structures are general and can include pressure vessel internals and walls, containment structures and walls, fuel rods with nuclear or electrical heating, steam generator tubes, and piping walls.

Figure 1.1 illustrates a heat structure between two control volumes. The heat structure is inclined at some angle with respect to the vertical and is partially immersed. Although the geometry shown here is rectangular, a heat structure may have a rectangular, cylindrical, spherical, or hemispherical geometry.

The heat structure in Figure 1.1 is nodalized with N temperature nodes. The nodalization is specified by user input and may be non-uniform (i.e., the distance between temperature nodes need not be the same). Node 1 is the temperature node at the left boundary surface for a rectangular geometry or at the inside boundary surface for a cylindrical, spherical, or hemispherical geometry. Node N is the temperature node at the right boundary surface for a rectangular geometry or at the outside boundary surface for other geometries.

The region between two adjacent temperature nodes is called a mesh interval. Each mesh interval may contain a different material. The material in each mesh interval is specified by user input. The Material Properties package provides thermal properties for each material through an interface with the HS package. Most materials commonly found in PWRs and BWRs are included in the Material Properties package default database, and properties for materials that are not included can easily be defined through Materials Properties package user input (refer to the MP package documentation).

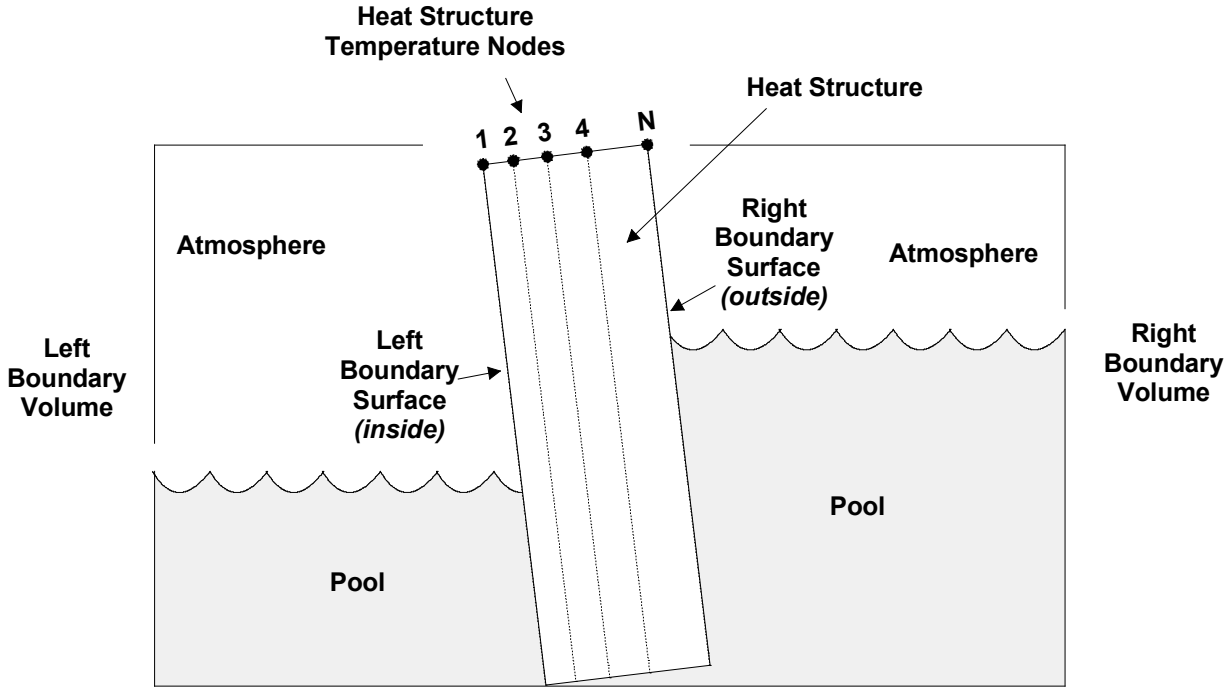


Figure 1.1 Heat Structure in a Control Volume

An internal power source may be specified for a heat structure. Its spatial dependence is specified by user input and may vary for each mesh interval. Its time dependence is given by a user-specified tabular function or control function.

Each heat structure has two boundary surfaces—left and right for rectangular geometries or inside and outside for cylindrical, spherical, or hemispherical geometries. At each boundary surface one of the following boundary conditions is specified:

(1) symmetry (adiabatic)

(2) convective with calculated heat transfer coefficient,

(3) convective with calculated heat transfer coefficient and a specified surface power source function,

(4) convective with specified heat transfer coefficient function,

(5) specified surface temperature function, and

(6) specified surface heat flux function.

If a convective boundary condition is selected for a boundary surface, a control volume must be specified as its boundary volume. Furthermore, the entire boundary surface must

fit within its boundary volume—that is, the bottom of the surface (user value HSALT) must equal or exceed the elevation of the bottom of the control volume specified in the CVH user input, and the top of the surface (calculated from HSALT and the surface length and orientation) must not exceed the elevation of the top of the control volume. No boundary volume is permitted for a symmetry or specified surface temperature boundary condition and a boundary volume for the specified heat flux boundary condition is a user option.

If a boundary volume is specified for a surface, then some additional data are required through user input. For each boundary surface with a boundary volume, these data are its

(1) surface area,

(2) characteristic length (the dimension used in calculating the Reynolds, Grashof, Nusselt, and Sherwood numbers),

(3) axial length (length of structure along boundary surface, used to determine pool fraction),

(4) type of flow over the surface (internal or external; used in calculating the Nusselt number), and

(5) critical pool fractions for pool and atmosphere heat transfer.

The pool fraction of a heat structure boundary surface is the fraction of its surface area in the pool of its boundary volume. Pool fractions and critical pool fractions permit a weighting of heat and mass transfer to the boundary volume atmosphere and pool. These are discussed in detail in Section 2.4.

If a convective boundary condition with calculated heat transfer coefficient is specified, then an extensive set of correlations is available for calculating natural or forced convection to the pool and atmosphere. Pool boiling heat transfer is calculated if the temperature of a heat structure surface is above the boundary volume saturation temperatures by utilizing correlations for nucleate boiling, critical heat flux, film boiling, and transition boiling.

Radiative heat transfer from a heat structure surface to the boundary volume pool is calculated during boiling. Radiative heat transfer can also be specified between a heat structure surface and the boundary volume atmosphere. Note, however, that radiation heat transfer to the atmosphere occurs only if the atmosphere contains water vapor (steam) and/or carbon dioxide; all other gases are considered to be non-absorbing by MELCOR. Two options, an equivalent band model and a gray gas model, are currently available. Radiation between user-specified pairs of surfaces may also be modeled, as described in Section 2.6.2.2. Radiative heat transfer between the COR components and HS structures is discussed in the COR package documentation.

Mass transfer between a heat structure surface and the boundary volume atmosphere is modeled using correlations or expressions for calculating mass flux. Models include condensation and evaporation in the presence of noncondensibles with an appropriate limit for pure steam, and flashing in any environment. Liquid films on heat structure surfaces are also modeled so that condensate transferred from the boundary volume atmosphere and liquid deposited by other packages can be treated. An optional film tracking model is available to track condensate film drainage from structure to structure. The film tracking model is activated when the user defines one or more network(s) of connected structures, such as stacked cylindrical sections to represent the steam generator tubes and/or cylindrical shells capped by a hemisphere to represent a containment dome. The user also specifies a drainage pattern for each network, which consists of drainage destinations and fractions for the drainage from each structure in the network. Drainage from a structure surface may be partitioned between three destination types:

(1) the surface of one or more additional structures in the network,

(2) “rain” that is passed to the MELCOR Containment Sprays (SPR) package via the Transfer Process (TP) package, and/or

(3) the pool of the CVH volume associated with the surface.

The user may also designate an external source of water for any structure in the network via tabular function input or a control function. External sources are primarily intended to allow the user to model the source for a passive containment cooling system or some such similar cooling device. When the film tracking model is active, the film thickness is calculated as a function of the condensate flow rate throughout the network.

Mass transfer affects the temperature distribution within a heat structure by its energy flux at the surface. This energy flux due to mass transfer is included in the boundary conditions for the conduction calculations, and film/atmosphere interfacial temperatures are calculated simultaneously with the structure node temperatures. The volume occupied by liquid films affects the virtual volume tracked by the CVH package, and the presence of liquid films also affects the rate and accumulation of radionuclides deposited on the surfaces by the RN package (see RN documentation). Decay heat from deposited radionuclides is treated as a power source at the surface in the equation for the surface temperature.

Finite-difference equations are used to advance the temperature distribution of a heat structure in time during MELCOR execution or to obtain its steady-state temperature distribution during MELGEN execution if specified by user input. These equations are obtained from an integral form of the one-dimensional heat conduction equation and boundary condition equations utilizing a fully implicit numerical method. The finite-difference approximation is a tridiagonal system of N equations (or N + 1 or N + 2 if there is a liquid film on one or both surfaces of the structure) for a heat structure with N

temperature nodes (or N + 1 or N + 2 temperature nodes if there is liquid film on one or both surfaces of the structure). The solution of this system is obtained using the standard solution algorithm for a tridiagonal system of linear equations.

A degassing model is provided for the release of gases from materials that are contained in heat structure mesh intervals. Input may be provided, for example, to represent the release of water vapor or carbon dioxide from concrete as its temperature increases. The HS package calculates a constant gas release rate over the degassing temperature range and modifies the thermal properties over this range to account for the energy associated with the gas production and release. The degassing model is also used in a modified form to treat ice condensers.

Communication of mass and energy changes to other packages is achieved through well-defined interfaces.

The remainder of the reference manual amplifies this calculation procedure. An enumeration and description of all models employed in the HS package calculations are included in Section 2. The solution methods used by the HS package are discussed in Section 3. Section 4 elaborates on the timestep control use by this package.

References for the HS Package Reference Manual follows Appendix A that contains information on the sensitivity coefficients used in the HS package.

2. Detailed Models

The modeling of a heat structure in the MELCOR Code and the calculation procedure for the HS package are discussed in Section 1. This section provides a detailed description of the models that are utilized by the HS package in the calculation procedure.

Heat conduction within a heat structure is modeled by the heat conduction equation in one spatial dimension. This equation and the specification of boundary conditions constitute a well-defined mathematical problem for the temperature distribution of a heat structure. However, the generality of boundary conditions, the inclusion of surface power sources and mass transfer at each boundary surface, temperature-dependent thermal properties, spatial-dependent materials, and the variety of geometries preclude the possibility of analytic solutions for the temperature distribution. Therefore, the HS package utilizes numerical methods for the determination of the temperature distribution for each heat structure. The description of detailed models in the HS package begins in Section 2.1 with a presentation of the finite-difference equations that approximate the heat conduction equation within a heat source. The finite-difference equations that approximate the heat conduction equation at the boundary surfaces are presented in Section 2.2.

The finite-difference equations of Sections 2.1 and 2.2 require specification or calculation of the following:

(1) power sources,

(2) pool fractions,

(3) thermal properties,

(4) heat transfer,

(5) mass transfer, and

(6) liquid film modeling.

Sections 2.3 through 2.8 provide the detailed models that specify these items.

Knowledge of the temperature distribution of a heat structure permits the calculation of its stored energy. The definition of stored energy of a heat structure is given in Section 2.9 within the context of the approximations of the HS package.

The thermal interactions between heat structures and control volumes result in the transfer of mass and energy between the CVH and HS packages. The HS package calculates such transfers between modules for the following:

(1) heat flux,

(2) liquid film evaporation and condensation, and

(3) degassing.

The detailed modeling of these phenomena in the HS package is discussed in Sections 2.6, 2.7, and 2.10. The COR package calculates heat transfer from the core to the bounding heat structures and passes the resulting energy transfers to the appropriate heat structures through an interface with the HS package (see COR package documentation for further details).

2.1 Finite-Difference Equations for Interior

The equation that governs conduction heat transfer in the interior of a heat structure is the one-dimensional heat conduction equation. This equation has the form

$$C_p \frac{\partial T}{\partial t} = \frac{1}{A} \frac{\partial}{\partial x} \left(k \; A \frac{\partial T}{\partial x} \right) + U \qquad (2\text{-}1)$$

where

C_p = volumetric heat capacity (product of heat capacity at constant pressure and density) [kJ/m^3•K]

T = temperature [K]

$\frac{\partial}{\partial t}$ = partial derivative with respect to time

A = heat transfer area [m2]

k = thermal conductivity [W/m-K]

$\frac{\partial}{\partial x}$ = partial derivative with respect to spatial variable

U = volumetric power [W/m^3]

The heat conduction equation is a parabolic partial differential equation. The HS package must solve it with boundary and initial conditions to determine the temperature distribution at each point in a heat structure. Sections 2.1.1 and 2.1.2 discuss the finite-difference approximation of Equation (2-1) in the interior of a heat structure.

2.1.1 Nodalization at Interior Temperature Nodes

The finite-difference approximation of the heat conduction equation requires a spatial partitioning of the heat structure into a finite number of temperature nodes. Temperature nodes must be located at the boundary surfaces and at interfaces between different materials. Additional nodes may be located at arbitrary locations within individual materials.

The region between two adjacent temperature nodes is called a mesh interval. For rectangular geometries, the node locations are relative to the node at the left boundary; for cylindrical geometries, they are relative to the axis of the cylinder; and for spherical or hemispherical geometries, they are relative to the center of the sphere. The location of the temperature nodes increases in a monotonic manner from the node at the left or inside boundary surface.

Figure 2.1 illustrates the nodalization of the interior of a heat structure near the *n*-th temperature node. This figure contains three temperature nodes and the mesh intervals for which they are the boundary points. For a rectangular geometry, the HS volume that is depicted in Figure 2.1 is part of a rectangular solid; for a cylindrical geometry, it is part of a cylindrical shell; and for a spherical or hemispherical geometry, it is part of a spherical shell. The quantities represented in Figure 2.1 are:

N = number of temperature nodes in heat structure

n = interior node number (2, 3, …, N-1)

X = location of temperature node

ΔX_n = $X_{n+1} - X_n$, length of n-th mesh interval

k = thermal conductivity of material

C_p = volumetric heat capacity of material

U = volumetric power source

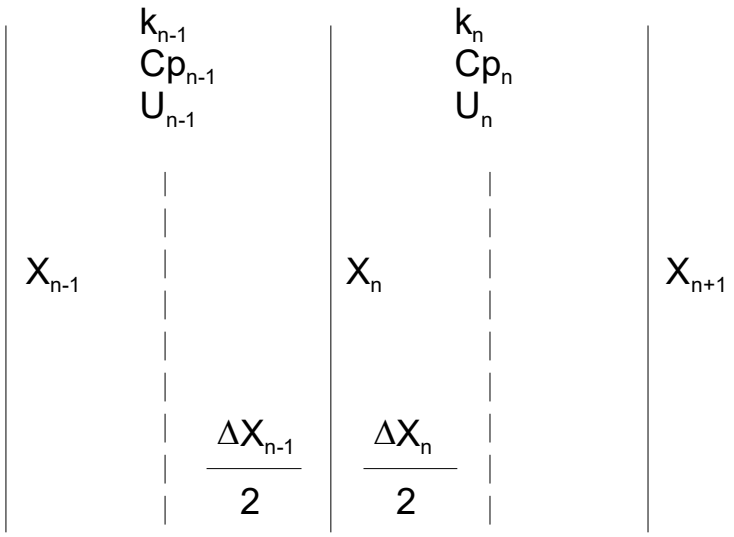


Figure 2.1 Nodalization in Interior of a Heat Structure

This figure also shows thermal properties and volumetric power sources in the mesh intervals adjacent to the n-th temperature node. These quantities are present in the finite-difference equations and are discussed in Sections 2.3 and 2.5.

To allow a more general representation of the equations and to consolidate expressions that define the numerical approximation, the following geometrical quantities are used [1]:

HSLn = left surface weight for n-th temperature node

HVLn = left volume weight for n-th temperature node

HSRn = right surface weight for n-th temperature node

HVRn = right volume weight for n-th temperature node

Table 2.1 Surface and Volume Weights in Interior

Rectangular Geometries	**Equation**
$HSL_n = 1/\Delta X_{n-1}$	(2-2)
$HVL_n = \Delta X_{n-1}/2$	(2-3)
$HSR_n = 1/\Delta X_n$	(2-4)
$HVR_n = \Delta X_n/2$	(2-5)
Cylindrical Geometries	**Equation**
$HSL_n = 2\pi\left(X_n - \Delta X_{n-1}/2\right)/\Delta X_{n-1}$	(2-6)
$HVL_n = \pi\left[X_n^2 - \left(X_n - \Delta X_{n-1}/2\right)^2\right]$	(2-7)
$HSR_n = 2\pi\left(X_n + \Delta X_n/2\right)/\Delta X_n$	(2-8)
$HVR_n = \pi\left[\left(X_n + \Delta X_n/2\right)^2 - X_n^2\right]$	(2-9)
Spherical Geometries	**Equation**
$HSL_n = 4\pi\left(X_n - \Delta X_{n-1}/2\right)^2/\Delta X_{n-1}$	(2-10)
$HVL_n = (4\pi/3)\left[X_n^3 - \left(X_n - \Delta X_{n-1}/2\right)^3\right]$	(2-11)
$HSR_n = 4\pi\left(X_n + \Delta X_n/2\right)^2/\Delta X_n$	(2-12)
$HVR_n = (4\pi/3)\left[\left(X_n + \Delta X_n/2\right)^3 - X_n^3\right]$	(2-13)
Hemispherical Geometries	**Equation**
$HSL_n = 2\pi(X_n - \Delta X_{n-1}/2)^2 / \Delta X_{n-1}$	(2-14)
$HVL_n = (2\pi/3)\left[X_n^3 - \left(X_n - \Delta X_{n-1}/2\right)^3\right]$	(2-15)
$HSR_n = 2\pi(X_n + \Delta X_n/2)^2 / \Delta X_n$	(2-16)
$HVR_n = (2\pi/3)\left[\left(X_n + \Delta X_n/2\right)^3 - X_n^3\right]$	(2-17)

The surface and volume weights in the interior of a heat structure are defined in Table 2.1. The interior temperature nodes correspond to n = 2, 3 …, N-1. The weights for n = 1 and N are defined in Section 2.2.1.

The surface and volume weights may be interpreted by considering for each geometry the rectangular solids, cylindrical shells, and spherical shells that are bordered by a temperature node and have thicknesses equal to half the length of the mesh intervals adjacent to this node. For all geometries, each surface weight has a factor that is the reciprocal of the length of the appropriate mesh interval. These weights appear in the gradient terms of the difference equations. For rectangular geometries, the other factors in the surface weight and the volume weight are the surface area and volume per unit area of one of the solids, respectively. For cylindrical geometries, they are the surface area and volume per unit axial length of one of the shells; and for spherical or hemispherical geometries, they are the surface area and volume of one of the shells. By definition, $HSL_{n+1} = HSR_n$ for all geometries, which ensures conservation.

2.1.2 Difference Approximation at Interior Nodes

The finite-difference equations are obtained from an integral form of the heat conduction equation. Consider multiplying Equation (2-1) by the area term and integrating the result over a heat structure. This integral equals the sum of integrals each of which is evaluated over a solid that is bounded by the dashed lines in Figure 2.1. The finite-difference approximation at the n-th interior temperature node is obtained from the integral of this equation over the solid that is bounded by these dashed lines. This approximation has the form

$$G_n\left(T_n^{m-1} - T_n^m\right)/\Delta t_m = k_{n-1}HSL_n\left(T_{n-1} - T_n\right) + k_nHSR_n\left(T_{n+1} - T_n\right) + \left(U_{n-1}HVL_n + U_nHVR_n\right) \quad (2\text{-}18)$$

where

T_n^m = temperature of n-th node at time t_m

G_n = $Cp_{n-1}\,HVL_n + Cp_n\,HVR_n$

U_n = volumetric power for n-th mesh interval

Δt_m = timestep for m-th computational cycle

$_n$ = quantity at n-th temperature node or mesh interval

^{m} = quantity at time t_m

$^{m+1}$ = quantity at time $t_m + \Delta t_m$

The time superscript for most of the terms in this equation is omitted. If all are *m*, then the finite-difference formulation is fully explicit. If all are *m* + 1, then the formulation is fully implicit. The fully implicit method is used by the HS package, so

$$G_n^{m+1}\,(T_n^{m+1} - T_n^m)/\Delta t_m = d_n^{m+1} \tag{2-19}$$

where

d_n^i = right side of Equation (2-18) at time t_i

For steady-state initialization calculations, the appropriate difference equation is

$$d_n = 0 \tag{2-20}$$

2.1.3 Finite-Difference Equations at Interior Temperature Nodes

The finite-difference equation at each interior temperature node is obtained by expanding Equation (2-19) or (2-20) and collecting the temperature terms at the *m* + 1 time level on the left. This equation is

$$A_{n-1}^{m+1}T_{n-1}^{m+1} + B_n^{m+1} + C_n^{m+1}T_{n+1}^{m+1} = D_n^{m+1} \tag{2-21}$$

where, in addition to previously defined quantities,

$$A_n^{m+1} = -k_{n-1}^{m+1}HSL_n\Delta t_m$$

$$C_n^{m+1} = -k_n^{m+1}HSR_n\Delta t_m$$

$$B_n^{m+1} = -A_n^{m+1} - C_n^{m+1} + a\,G_n^{m+1}$$

$$D_n^{m+1} = a\,G_n^{m+1}T_n^m + \left(U_{n-1}^{m+1/2}HVL_n + U_n^{m+1/2}HVR_n\right)\Delta t_m$$

a = 1 for transient calculations

= 0 for steady-state calculations

n = 2, 3, …, N-1

The value of the power, *U*, is evaluated as the average of old and new time values in order to more accurately reflect the desired input energy. The result of applying Equation (2-21) to a heat structure with *N* temperature nodes is a tridiagonal system of *N*-2 equations for the interior temperature nodes. (Note: editorial corrections have been made to coefficients in Equation (2-21)).

2.2 Finite-Difference Equations at Boundary Surfaces

The numerical calculation of the temperature distribution of a heat structure not only requires a finite-difference approximation of the heat conduction equation at interior temperature nodes, but also a finite-difference approximation of this equation and the boundary condition at each boundary surface.

There are two basic cases to consider at the surfaces of a structure:

(1) the case when there is no liquid film on the surface, and

(2) the case when there is a liquid film on the surface.

If there is no liquid film, then a boundary condition is applied to the structure surface and used to calculate the structure surface temperature. If a liquid film exists, then an additional mesh interval, consisting of the film bounded by the structure surface temperature node on the inside and the film/atmosphere interfacial temperature node on the outside, is defined, and a conduction equation for the film/atmosphere interfacial temperature is added to the set of *N* equations for the structure node temperatures. In this case, the equation for the structure surface (i.e., the structure/film interface) temperature is similar to the equations for the temperatures at the interior nodes, except that the half mesh interval on the outside consists of half of the liquid film instead of structure material. Hence, if there is no liquid film on either surface of the structure, the tridiagonal set consists of *N* equations (*N*-2 for interior nodes and 2 for the two surface node temperatures), while the set consists of *N*+1 (or *N*+2) temperature equations, if there is a liquid film on one (or both) surfaces of the structure.

Only certain types of boundary conditions are permitted if mass transfer (liquid film condensation/evaporation) is to be treated; film formation is prohibited if an adiabatic, specified surface heat flux or specified surface temperature boundary condition is imposed.

In the discussion that follows, it is to be understood that the boundary condition is applied at the film/atmosphere interface and not the structure/film interface if a liquid film exists on the surface of the structure. The general form of the boundary condition at the surface of a heat structure is

$$\alpha T + \beta \frac{dT}{dN} = \gamma \qquad (2\text{-}22)$$

where

α = first boundary condition coefficient

β = second boundary condition coefficient

γ = third boundary condition coefficient

T = temperature of surface

$\frac{dT}{dN}$ = gradient of temperature in direction of outward normal

This expression is implicit in the surface temperature, which is determined iteratively. All variables in this expression that are part of the heat structure package database (structure temperatures and properties that are functions of the structure temperature) are treated implicitly during the iteration procedure. Variables from other MELCOR packages (CVH temperatures and energy deposited by other packages) must be treated explicitly because of the explicit coupling between all MELCOR packages. All permitted boundary conditions can be put into this form as shown below.

2.2.1 Boundary Condition Coefficients

2.2.1.1 Symmetry (Adiabatic)

The symmetry boundary condition is represented by

$$\frac{dT}{dN} = 0 \tag{2-23}$$

For this boundary condition, the boundary condition coefficients are

$\alpha = 0$

$\beta = 1$

$\gamma = 0$

2.2.1.2 Convective (Calculated or Specified Heat Transfer Coefficients)

The convective boundary condition is represented by

$$-k\frac{dT}{dN} + S = (h_{atm} + h_{atmr})\left(1 - x_{pool}\right)(T - T_{atm}) + h_{pool}x_{pool}\left(T - T_{pool}\right) \tag{2-24}$$

where

k = thermal conductivity [W/m-K]

S = surface energy flux (flowing into heat structure)

h_{atm} = atmosphere heat transfer coefficient

h_{atmr} = atmosphere radiative heat transfer coefficient

x_{pool} = fraction of surface in pool of boundary volume

T_{atm} = temperature of atmosphere in boundary volume

h_{pool} = pool heat transfer coefficient

T_{pool} = temperature of pool in boundary volume

For these boundary conditions, the coefficients are

$$\alpha = h_{pool} x_{pool} + (h_{atm} + h_{atmr})(1 - x_{pool})$$

$$\beta = k$$

$$\gamma = h_{pool} x_{pool} T_{pool} + (h_{atm} + h_{atmr})(1 - x_{pool}) T_{atm} + S$$

2.2.1.3 Specified Surface Heat Flux

For specified heat flux at the surface, the boundary condition is represented by

$$-k \frac{dT}{dN} = q'' \tag{2-25}$$

where

q'' = specified heat flux at surface (positive out)

For these boundary conditions, the coefficients are

$$\alpha = 0$$

$$\beta = k$$

$$\gamma = -q''$$

2.2.1.4 Specified Surface Temperature

The boundary condition for a specified surface temperature is represented by

$T = T_{surf}$ = specified surface temperature (2-26)

For these boundary conditions, the coefficients are

$\alpha = 1$

$\beta = 0$

$\gamma = T_{surf}$

2.2.2 Nodalization at Boundary Temperature Nodes

Figure 2.2 illustrates the geometry of a heat structure near the surface temperature nodes of a heat structure. This figure contains two temperature nodes and the mesh intervals for which they are the boundary points at both surfaces. It also depicts the condensate films that may or may not be present on each boundary surface. For all geometries, the volumes that are depicted in this figure are as described in Section 2.1.1. The quantities represented in Figure 2.2 are:

N = number of temperature nodes in heat structure

X = location of temperature node

ΔX_n = $X_{n+1} - X_n$, length of n-th mesh interval

k = thermal conductivity of material

c_p = volumetric heat capacity of material

U = volumetric power

S = surface power

δ_f = thickness of liquid film

m_f = mass of liquid film

h_f = specific enthalpy of liquid film

Cp specific heat of liquid film

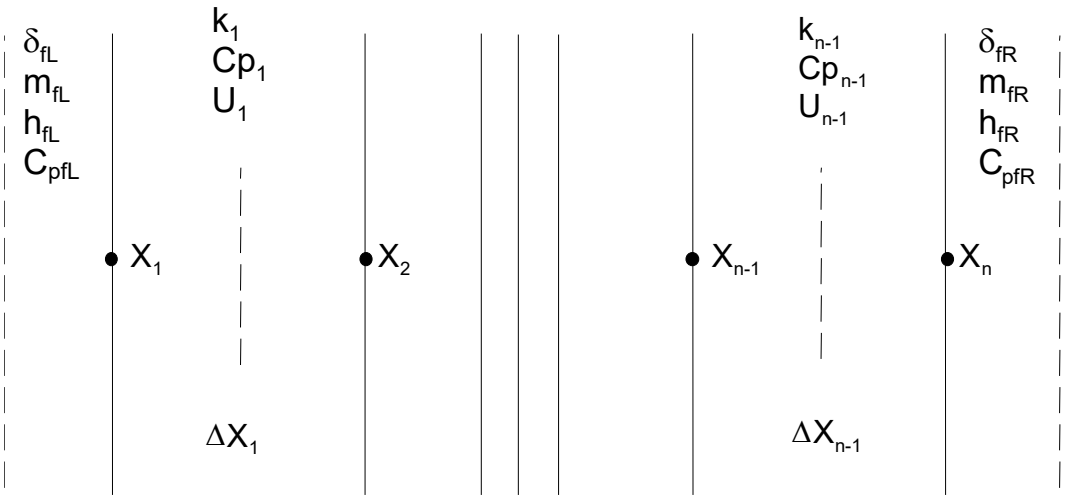


Figure 2.2 Nodalization at Boundary Surfaces of a Heat Structure

Table 2.2 Surface and Volume Weights at Boundary Surfaces

Rectangular Geometries	**Equation**
$HSL_1 = 1$	(2-27)
$HVL_1 = 0$	(2-28)
$HSR_1 = 1/\Delta X_1$	(2-29)
$HVR_1 = \Delta X_1/2$	(2-30)
$HSL_N = 1/\Delta X_{N-1}$	(2-31)
$HVL_N = \Delta X_{N-1}/2$	(2-32)
$HSR_N = 1$	(2-33)
$HVR_N = 0$	(2-34)
Cylindrical Geometries	**Equation**
$HSL_1 = 2\pi X_1$	(2-35)
$HVL_1 = 0$	(2-36)
$HSR_1 = 2\pi(X_1 + \Delta X_1/2)/\Delta X_1$	(2-37)
$HVR_1 = \pi\left[(X_1 + \Delta X_1/2)^2 - {X_1}^2\right]$	(2-38)

$HSL_N = 2\pi(X_N - \Delta X_{N-1}/2)/\Delta X_{N-1}$	(2-39)
$HVL_N = \pi\left[X_N^{\,2} - \left(X_N - \Delta X_{N-1}/2\right)^2\right]$	(2-40)
$HSR_N = 2\pi X_N$	(2-41)
$HVR_N = 0$	(2-42)
Spherical Geometries	**Equation**
$HSL_1 = 4\pi X_1^{\,2}$	(2-43)
$HVL_1 = 0$	(2-44)
$HSR_1 = 4\pi(X_1 + \Delta X_1/2)^2/\Delta X_1$	(2-45)
$HVR_1 = (4\pi/3)\left[(X_1 + \Delta X_1/2)^3 - X_1^{\,3}\right]$	(2-46)
$HSL_N = 4\pi(X_N - \Delta X_{N-1}/2)^2/\Delta X_{N-1}$	(2-47)
$HVL_N = (4\pi/3)\left[X_N^{\,3} - (X_N - \Delta X_{N-1}/2)^3\right]$	(2-48)
$HSR_N = 4\pi X_N^{\,2}$	(2-49)
$HVR_N = 0$	(2-50)
Hemispherical Geometries	**Equation**
$HSL_1 = 2\pi X_1^{\,2}$	(2-51)
$HVL_1 = 0$	(2-52)
$HSR_1 = 2\pi(X_1 + \Delta X_1/2)^2/\Delta X_1$	(2-53)
$HVR_1 = (2\pi/3)\left[(X_1 + \Delta X_1/2)^3 - X_1^{\,3}\right]$	(2-54)
$HSL_N = 2\pi(X_N - \Delta X_{N-1}/2)^2/\Delta X_{N-1}$	(2-55)
$HVL_N = (2\pi/3)\left[X_N^{\,3} - \left(X_N - \Delta X_{N-1}/2\right)^3\right]$	(2-56)
$HSR_N = 2\pi X_N^{\,2}$	(2-57)
$HVR_N = 0$	(2-58)

The figure also shows thermal properties and volumetric power sources in the mesh intervals adjacent to the boundary nodes. These quantities are present in the finite-difference equations and are discussed in Sections 2.3 and 2.5.

The surface and volume weights are also defined at the boundary surfaces. The definitions are given in Table 2.2 for the case when no liquid film exists on either surface. The surface and volume weights for the case involving liquid films are similar, except there is one additional temperature node on each side that has a liquid film, and the mesh interval (with a thickness equal to that of the liquid film, δ_f) between the additional node (on the outside) and the structure surface node (on the inside) contains the liquid film. The surface and volume weights may be interpreted as discussed in Section 2.1.1 except (a) the left volume weight at the left (inside) temperature node and the right volume weight at the right (outside) temperature node are zero and (b) the left surface weight at the left (inside) temperature node and the right surface weight at the right (outside) boundary surface are the areas of these respective surfaces.

2.2.3 Difference Approximation at Boundary Nodes

The finite-difference equations at boundary nodes are obtained from an integral form of the heat conduction equation. The finite-difference approximation at the boundary temperature nodes is obtained from the integral of Equation (2-1) (multiplied by the area term) over the solid that is bounded by the film surface and the dashed line in Figure 2.2.

2.2.3.1 Finite-Difference Equation at Left (Inside) Boundary

By using Equation (2-22) to eliminate the spatial derivative term, the finite-difference approximation has the following form at the left (inside) boundary surface:

$$G_1^{m+1}\left(T_1^{m+1} - T_1^m\right)/\Delta t_m = \left[k_1\left(\gamma_L - \alpha_L T_1\right)/\beta_L\right] HSL_1 + k_1\left(T_2 - T_1\right) HSR_1 + U_1 HVR_1 - m_{f,L}^m \Delta h_{f,L}/f \qquad (2\text{-}59)$$

(Note: editorial corrections have been made to coefficients in Equation (2-59)). If there is no liquid film on that surface (i.e., there was and is no film, or there was film that completely evaporated or was transferred to the pool associated with the boundary volume). If there is a liquid film, then the equations for the film surface temperature and structure surface temperature are:

$$G_{f,L}^{m+1}\left(T_{f,L}^{m+1}-T_{f,L}^{m}\right)/\Delta t_m = HSR_{f,L}H_{f,L}\left(T_1-T_{f,L}\right)$$
$$+HSL_{f,L}\left[\left(1-x_{pool,L}\right)\cdot\left(H_{atm,L}+H_{atmr,L}\right)\cdot\left(T_{atm,L}-T_{f,L}\right)+x_{pool,L}H_{pool,L}\cdot\left(T_{pool,L}-T_{f,L}\right)\right] \quad (2\text{-}60a)$$
$$+\left[\left(1-x_{pool,L}\right)A_L/f\right]\cdot\left[\left(h_{V,L}-h_{f,L}\right)\cdot\max\left(0,\dot{m}_{C,L}\right)+\left(h_{V,L}-\overline{h}_{f,L}\right)\cdot\min\left(0,\dot{m}_{C,L}\right)\right]$$

$$\left(G_1^{m+1}+G_{f,L}^{m+1}\right)\left(T_1^{m+1}-T_1^{m}\right)/\Delta t_m = HSR_1k_1\left(T_2-T_1\right)-HSR_{f,L}H_{f,L}\left(T_1-T_{f,L}\right)$$
$$+\left[\left(1-x_{pool,L}\right)A_L/f\right]\left(h_{f,L}-\overline{h}_{f,L}\right)\max\left(0,\dot{m}_{c,L}\right)+U_1HVR_1 \quad (2\text{-}60b)$$

where

G_1^{m+1} = $C_{p1}^{m+1}HVR_1$

f = geometry factor

= surface area of heat structure for rectangular geometries

= axial length of heat structure for cylindrical geometries

= 1.0 for spherical and hemispherical geometries

m_f^m = (old) mass of film evaporated (or transferred) this timestep

Δh_f = specific enthalpy added to m_f^m before its removal

= latent heat of vaporization if film evaporated

= 0.0 if film was transferred to pool

HSR_f = HSR evaluated at film mass median surface

HSL_f = HSL evaluated at film/atmosphere interface

A = structure surface area

x_{pool} = fraction of boundary surface in pool of boundary volume

H_{atm} = convective heat transfer coefficient to atmosphere

H_{atmr} = radiative heat transfer coefficient to atmosphere

H_{pool} = convective heat transfer coefficient to pool

$H_{f,l}$ = convective/conductive heat transfer coefficient through film

T_{atm} = temperature of atmosphere in boundary volume

T_{pool} = temperature of pool in boundary volume

m_f = mass of film

$\dot{m}_c$ = condensation/evaporation mass flux (+/- for cond/evap)

c_{pf} = specific heat capacity of film

G_f^{m+1} = $0.5\, m_f^{m+1} c_{pf}^{m+1} / f$

h_v = specific enthalpy of vapor in boundary volume

$h_{f,L}$ = specific enthalpy of film evaluated at film/atmosphere interfacial temperature

$h_{f,1}$ = specific enthalpy of film evaluated at structure/film interfacial temperature

$\bar{h}_f$ = 0.5 x ($h_{f,L}$ + $h_{f,1}$)

$_L$ = quantity at left (inside) boundary surface

$_i$ = quantity at *i*-th temperature node or mesh interval

^{m} = quantity at time t_m

$^{m+1}$ = quantity at time $t_m + \Delta t_m$

Equation (2-60) does not apply when $\beta = 0$. This corresponds to the specified surface temperature boundary condition, in which case $T_1^{m+1} = T_{surf,L}^{m+1}$ and the presence of surface films and mass transfer is not permitted.

The time superscript for each term on the right side of Equation(2-60) is omitted. If all are *m*, then the finite-difference formulation is fully explicit. If all are *m*+1, then the formulation is fully implicit. The fully implicit numerical method is used by the HS package. Equation (2-60) is used for both steady-state (MELGEN) and transient (MELCOR) calculations, except that for steady-state calculations the old time values (*m*) are overwritten with the new time values (*m*+1) after each iteration. Hence, when convergence is achieved in MELGEN, temperatures that do not change with time (steady-state) have been determined. In MELGEN the timestep size is given by the value of sensitivity coefficient C4051(3), which is 10^5 s by default.

The finite-difference equation(s) at the left (inside) boundary temperature node(s) are obtained by expanding Equation (2-60) and collecting the temperature terms at the *m*+1 time level on the left. Equation (2-60) reduces to

$$B_1^{m+1}T_1^{m+1} + C_1^{m+1}T_2^{m+1} = D_1^{m+1} \qquad \text{(2-61a)}$$

where for β_L not zero,

$$B_1^{m+1} = G_1^{m+1} - C_1^{m+1} + \left(\alpha_L^{m+1}k_1^{m+1}HSL_1\Delta t_m\right)/\,\beta_L^{m+1}$$

$$C_1^{m+1} = -k_1^{m+1}HSR_1\Delta t_m$$

$$D_1^{m+1} = G_1^{m+1}T_1^m + \left(\gamma_L^{m+1}k_1^{m+1}HSL_1\Delta t_m\right)/\,\beta_L^{m+1} + U_1^{m+1/2}HVR_1\Delta t_m - m_{f,L}^m\,\Delta h_{f,L}/f$$

and for β_L zero,

$$B_1^{m+1} = 1$$

$$C_1^{m+1} = 0$$

$$D_1^{m+1} = \gamma_L^{m+1} = T_{surf,L}^{m+1}$$

Equation(2-60a) and(2-60b) reduce to

$$B_L^{m+1}T_L^{m+1} + C_L^{m+1}T_1^{m+1} = D_L^{m+1} \qquad \text{(2.61b)}$$

$$A_1^{m+1}T_{f,L}^{m+1} + B_1^{m+1}T_1^{m+1} + C_1^{m+1}T_2^{m+1} = D_1^{m+1} \qquad \text{(2.61c)}$$

where

$$B_L^{m+1} = G_{f,L}^{m+1} - C_L^{m+1} + HSL_{f,L}\left[\left(1-x_{pool,L}\right)\left(H_{atm,L}^{m+1} + H_{atmr,L}^{m+1}\right) + x_{pool,L}H_{pool,L}^{m+1}\right]\Delta t_m$$

$$C_L^{m+1} = -H_{f,L}^{m+1}HSR_{f,L}\Delta t_m$$

$$D_L^{m+1} = G_{f,L}^{m+1}T_{f,L}^m + HSL_{f,L}\left[\left(1-x_{pool,L}\right)\left(H_{atm,L}^{m+1} + H_{atmr,L}^{m+1}\right)T_{atm,L} + x_{pool,L}H_{pool,L}^{m+1}T_{pool,L}\right]\Delta t_m + \left[\left(1-x_{pool,L}\right)A_L/f\right]\left[\left(h_{v,L} - h_{f,L}^{m+1}\right)\max\left(0,\dot{m}_{c,L}^{m+1}\right) + \left(h_{v,L} - \overline{h}_{f,L}^{m+1}\right)\min\left(0,\dot{m}_{c,L}^{m+1}\right)\right]\Delta t_m$$

$$A_1^{m+1} = C_L^{m+1}$$

$$B_1^{m+1} = G_1^{m+1} + G_{f,L}^{m+1} - A_1^{m+1} - C_1^{m+1}$$

$$C_1^{m+1} = -k_1^{m+1} HSR_1 \Delta t_m$$

$$D_1^{m+1} = \left(G_1^{m+1} + G_{f,L}^{m+1}\right) T_1^m + \left[\left(1 - x_{pool,L}\right) A_L / f\right] \left[\left(h_{f,L}^{m+1} - \bar{h}_{f,L}^{m+1}\right) \max\left(0, \dot{m}_{c,L}^{m+1}\right)\right] \Delta t_m + U_1^{m+1/2} HVR_1 \Delta t_m$$

2.2.3.2 Finite-Difference Equation at Right (Outside) Boundary

The finite-difference equation(s) at the right (outside) boundary surface are exactly analogous to those at the left (inside) boundary surface. The subscripts 1, 2 and L are merely replaced by subscripts *N*, *N*-1 and *R*; *HSL* and *HSR* are reversed; *HVL* and *HVR* are reversed; and matrix elements *A* and *C* are reversed.

2.3 Power Sources

Power sources are included in the calculation of the temperature distribution of each heat structure. These sources include the following:

(1) internal power source,

(2) surface power source, and

(3) energy transferred by other packages.

These items are discussed in Sections 2.3.1 through 2.3.3, respectively.

2.3.1 Internal Power Sources

The internal power source is included in the temperature evolution equations as volumetric power terms in each mesh interval. User input specifies the spatial distribution of the power source for each heat structure that contains an internal power source. These data are used to calculate the fraction of power from a tabular function or control function that is applied to each mesh interval. The HS package calculates the volumetric power terms, U, that appear in the equations. For the *n*-th mesh interval,

$$U_n = x_{P,n} P_{\text{int}} / \left(HVL_{n+1} + HVR_n\right) / f \qquad (2\text{-}62)$$

where

n = 1,2,...,N-1

N-1 = number of mesh intervals

x_P = fraction of power from tabular function that is applied to this mesh interval (user input)

P_{int} = average of power from internal source tabular function at old and new times or power from a specified control function for transient calculations and time zero value for initialization calculations

HVL = left (inside) volume weights defined in Sections 2.1.1 and 2.2.2

HVR = right (outside) volume weights defined in Sections 2.1.1 and 2.2.2

f = geometry factor with the following values for different geometries

= surface area of heat structure for rectangular geometries

= axial length of heat structure for cylindrical geometries

= 1.0 for spherical and hemispherical geometries

2.3.2 Surface Power Sources

For a convective boundary condition with calculated heat transfer coefficients, a surface power source may be specified by a user-input control function or a tabular function of time. This source is included in Equations (2-24) and (2-25) as energy fluxes that are added to the boundary condition coefficients, γ, in Equation (2-24). The HS package calculates these fluxes, q''_{surf}, prior to calculating the temperature distribution of each heat structure. For each boundary surface, these terms are

$$q''_{surf} = P_{surf} / A_{surf}$$

where

q''_{surf} = energy flux at boundary surface from surface power source

P_{surf} = average of surface power from tabular function at old and new times for transient calculations and zero for initialization calculations

A_{surf} = area of boundary surface

2.3.3 Energy Transferred by Other Packages

The energy that is transferred to a heat structure surface by other packages is obtained from an array in the HS package database whose elements are updated using an interface subroutine that can be called any package. This energy includes, for example, the radiant energy from a core cell, conduction from debris deposited on the structure surface by the high-pressure-melt-ejection (HPME) model of the FDI package or the decay-heat energy of radionuclides deposited on a heat structure surface. This energy is included in Equations (2-24) and (2-25) as energy fluxes that are added to the boundary condition coefficients γ in Equation (2-24). The HS package calculates these fluxes, q''_{ext}, prior to calculating the temperature distribution of each heat structure. For each boundary surface, these terms are

$$q''_{ext} = E_{ext} / A_{surf} / \Delta t \qquad (2\text{-}63)$$

where

q''_{ext} = energy flux at boundary surface from energy that is transferred by other packages

E_{ext} = energy that is transferred to boundary surface by other packages since the previous call to the HS package for transient calculations and zero for initialization calculations

A_{surf} = area of boundary surface

Δt = computational timestep

For each boundary surface, the surface energy flux term S in Equation (2-24) used in determining the boundary coefficient γ is the sum of q''_{surf} and q''_{ext} obtained from the above equations.

2.4 Pool Fractions

When a heat structure with a convective boundary condition is in contact with a CVH volume containing either single-phase liquid or vapor, the implementation of the boundary condition is straightforward. However, if the surface is partially submerged, then it is necessary to partition the heat transfer between the pool and the atmosphere as described in Section 2.2.1.2. In this case, heat transfer is partitioned on the basis of a calculated fraction of the heat structure surface that is submerged as depicted in Figure 1.1. This fraction is called the pool fraction with a range of 0 to 1. This section describes how the pool fraction is calculated for each type of geometry and the controls available to the user.

There are two input parameters for each surface, CPFPL and CPFAL, which allow the user to disable heat transfer to the pool and/or atmosphere as a function of the pool fraction. The range of each is 0 to 1. Heat transfer to the pool is calculated only when the pool fraction exceeds the critical pool fraction CPFPL. Similarly, heat and mass transfer to the atmosphere occur only when the pool fraction falls below the critical pool fraction CPFAL. Note that CPFPL and CPFAL are completely independent. Furthermore, disabling heat transfer to either phase does not affect heat transfer to the other phase directly. Also, when permitted, the heat transfer rates are independent of the values of CPFPL and CPFAL. When heat transfer from either phase is permitted, it occurs over the fraction of the surface area that is in contact with that phase (as given by x_{pool} for the pool and 1-x_{pool} for the atmosphere in Equation (2-24). When heat transfer to either phase is disabled it is as though there is a perfectly insulating layer at the interface.

The primary use of this input feature is to prohibit simultaneous heat transfer to both pool and atmosphere when such an occurrence generates unrealistic results. The most common situation to be avoided occurs when a vertical structure is in contact with both a cool liquid pool and a hot atmosphere. In this case, if the heat structure can communicate with both phases simultaneously, the relatively large heat transfer coefficient to the pool pulls the one-dimensional structure surface temperature down to a value much closer to the pool temperature than the atmosphere temperature. Consequently, heat transfer from the atmosphere to the structure is much greater than should be expected. The net effect is an artificially large heat transfer from the atmosphere to the pool via the structure surface. This situation can be avoided by specifying equal values of CPFPL and CPFAL so that structure only communicates to either the pool or atmosphere.

For situations in which the pool and atmosphere temperatures in the boundary control volume are nearly the same and heat transfer to both phases is expected to be significant, the user should enable simultaneous communications with both phases by specifying a value of zero for CPFPL and one for CPFAL. This option is often used for steam generator heat structures. This should also be used for horizontal floors and ceiling for which the pool fraction is specially modified as described in Section 2.4.1.

A value of 0.0 for CPFPL, or of 1.0 for CPFAL, can lead to numerical problems because heat transfer may be calculated to an arbitrarily small fluid mass if the heat structure extends to the bottom or top of the control volume, respectively. To avoid this potential problem, bounds are imposed on the user-input values so that CPFPL $\geq$ 0.02 and CPFAL $\leq$ 0.98. These bounds are contained in sensitivity coefficient array 4071.

If CPFPL is greater than CPFAL, there is a dead band with no communication to either the pool or atmosphere. This unlikely situation is not currently treated as a fatal input error.

If CPFPL is less than CPFAL, there is a band with simultaneous heat transfer to both pool and atmosphere.

If CPFPL is equal to CPFAL, heat transfer switches from one phase to the other as the pool fraction crosses the critical value. In the special case where the calculated pool fraction is exactly equal to the common value of CPFPL and CPFAL, communication is to the pool if the value is greater than or equal to 0.5 and to the atmosphere otherwise.

Because of the potential for serious problems when unequal values of CPFPL and CPFAL are specified, MELGEN generates a warning message to alert the user to the potential for unrealistic results. Nevertheless, if simultaneous heat transfer to both the pool and atmosphere is unlikely to cause serious problems, then CPFPL and CPFAL should be chosen to permit simultaneous heat transfer (i.e., set CPFPL = 0 and CPFAL = 1).

The pool fraction for a surface is set to 0.0 if its lowest point is above the pool and set to 1.0 if it is completely immersed in the pool. If the pool/atmosphere interface is very close to the top or bottom of a heat structure surface (less than the maximum film thickness), the pool fraction is set to 1.0 or 0.0, respectively. For all other situations, the expressions given below are evaluated for the pool fraction. Sections 2.4.1 through 2.4.4 present these expressions for rectangular, cylindrical, spherical, and hemispherical geometries, respectively.

2.4.1 Rectangular Geometry

The pool fraction for a surface with a rectangular geometry is given by

$$x_{pool} = Z / L \cos(\alpha) \qquad (2\text{-}64)$$

where

Z = depth of pool in boundary volume of this surface relative to the altitude of the lowest point on this surface

L = axial length of this surface

α = angle between this surface and the vertical

For horizontal surfaces, such as floors and ceilings, where $\cos(\alpha) = 0$ in Equation (2-64), the pool fraction is defined to vary from zero to one as the pool surface ascends through a vertical distance of the maximum of liquid film thickness (see Section 2.8) and 10^{-6}m, which eliminates a step change in pool fraction.

2.4.2 Cylindrical Geometry

The following quantities are used for defining the pool fraction for a surface with a cylindrical geometry:

R = radius of cylinder containing the surface

L = axial length of this surface

α = angle between this surface and the vertical

a = vertical projection of cylinder diameter, $2R\sin(\alpha)$

b = vertical projection of cylinder axial length, $L\cos(\alpha)$

Z = depth of pool in boundary volume of this surface relative to the altitude of the lowest point on this surface

The pool fraction for a vertical surface, $\cos(\alpha)=1$, with a cylindrical geometry is given by

$$x_{pool} = Z/L \tag{2-65}$$

The pool fraction for a horizontal surface, $\cos(\alpha)=0$, with a cylindrical geometry is given by

$$x_{pool} = \Theta/\pi \tag{2-66}$$

where

$$\Theta = cos^{-1}[(R-Z)/R],$$

is in radians and is π for $Z > 2R$.

The pool fraction for a cylinder inclined at an angle α between vertical and horizontal is given as follows:

$$X_{pool} = R\sin(\alpha)\left[TERML(Z) + TERMR(Z)\right]/b \tag{2-67}$$

where $TERML(Z)$ and $TERMR(Z)$ are functions of Z that are derived by considering whether or not the pool surface intersects the bottom and/or top flat surfaces of the cylinder. Defining

$$XL = 1 - Z/\left[R\sin(\alpha)\right] \tag{2-68}$$

it can be shown that, if the pool intersects the bottom surface (XL > -1), then

$$TERML = \frac{1}{\pi}\left\{\left[1 - XL^2\right]^{1/2} - \frac{XL}{\cos(XL)}\right\} \tag{2-69}$$

Otherwise, the bottom surface is completely submerged, and

$$TERML = -XL \tag{2-70}$$

Similarly, defining

$$XU = (a + b - z)/[R\sin(\alpha)] - 1 \tag{2-71}$$

It can be shown that

$$TERMR = \frac{1}{\pi}\left\{\frac{XU}{\cos(XU)} - \left[1 - XU^2\right]^{1/2}\right\} \tag{2-72}$$

when the surface intersects the top end ($XU < 1$); otherwise,

$$TERMR = 0 \tag{2-73}$$

2.4.3 Spherical Geometry

The pool fraction for a surface with a spherical geometry is given by

$$x_{pool} = Z / 2R \tag{2-74}$$

2.4.4 Hemispherical Geometry

The pool fraction for a surface with a hemispherical geometry is given by

$$x_{pool} = Z / R \tag{2-75}$$

2.5 Thermal Properties

The conduction equations require the thermal conductivity and volumetric heat capacity (product of heat capacity and density) of the material in each mesh interval. These thermal properties are discussed in Section 2.5.1. Their modification for a degassible material is discussed in Section 2.5.2.

2.5.1 Thermal Conductivity and Volumetric Heat Capacity

The thermal conductivity, heat capacity, and density of the material in each mesh interval are obtained as a function of temperature using an interface with the Material Properties

package. They are obtained for the material in each mesh interval at a temperature that is the average of the temperatures of the nodes that are boundaries of this mesh interval.

2.5.2 Modifications for Degassible Materials

The volumetric heat capacity of a degassible material whose temperature is in the degassing temperature range is increased by an amount equal to the product of the heat of reaction and the source density divided by the degassing temperature range. The volumetric heat capacity is therefore replaced by

$$C_p + \Delta h_R \rho_{gas} / \Delta T_{gas} \tag{2-76}$$

where

C_p = volumetric heat capacity from Material Properties package, kJ/m^3•K

Δh_R = heat of reaction of gas source, kJ/kg

ρ_{gas} = density of gas in source, kg/m^3

ΔT_{gas} = degassing temperature range of gas source, K

This modification accounts for the energy that is required to produce and release the gas.

2.6 Heat Transfer

The methods of calculating heat transfer at a heat structure surface are discussed in this section for the following:

(1) specified temperature boundary conditions

(2) specified heat flux boundary conditions

(3) convective boundary conditions

If the temperature of a surface is specified by a tabular function, the heat flux is calculated from a finite-difference approximation that expresses the surface heat flux in terms of temperatures at the surface and adjacent nodes and quantities known in the interior of the heat structure. For the left boundary surface, this heat flux is given by

$$q''_L = \left[-k_1^{m+1}\left(T_1^{m+1} - T_2^{m+1}\right) HSR_1 + U_1 HVR_1 - Cp_1^m \frac{T_1^{m+1} - T_1^m}{\Delta t_m} HVR_1 \right] \Big/ HSL_1 \tag{2-77}$$

For the right boundary surface, this heat flux is

$$q''_R = \left[-k_N^{m+1}\left(T_N^{m+1} - T_{N-1}^{m+1}\right)HSL_N + U_N HVL_N - Cp_N^m \frac{T_N^{m+1} - T_N^m}{\Delta t_m} HVL_N \right] \Big/ HSR_N \quad (2\text{-}78)$$

where

k^m = thermal conductivity of heat structure at time t_m

C_p^m = volumetric heat capacity of heat structure at time t_m

HSL = left (inside) surface weight defined in Section 2.2.1

HSR = right (outside) surface weight defined in Section 2.2.1

HVL = left (inside) volume weight defined in Section 2.2.1

HVR = right (outside) volume weight defined in Section 2.2.1

U^m = volumetric power source at time t_m

T^m = node temperature at time t_m

T^{m+1} = node temperature at time t_{m+1}

Δt = timestep size, $t_{m+1} - t_m$

If the surface heat flux is specified by a tabular function, the heat flux is known from the value of the tabular function.

If a convective boundary condition is specified, the heat flux is the product of the heat transfer coefficient and the temperature difference between the surface (film surface, if a liquid film is present) and the atmosphere or pool of the boundary volume. The heat transfer coefficient is either calculated or provided by a tabular function of time or temperature.

If a convective boundary condition with calculated heat transfer coefficients is specified, then correlations are available for the following heat transfer regimes:

(1) atmosphere natural convection

(2) atmosphere forced convection

(3) pool natural convection

(4) pool forced convection

(5) pool boiling

The HS package calculates convective heat transfer between a heat structure and the boundary volume atmosphere whenever the pool fraction at a boundary surface is less

than or equal to its critical pool fraction for atmosphere heat transfer. Atmosphere heat transfer occurs through a gas boundary layer and, if condensate is present on the surface, through a liquid layer between the surface and the boundary layer. Radiative heat transfer also can occur between a heat structure and the boundary volume atmosphere. For this case, radiation and convection for the structure surface (or the film surface, if a liquid film exists) occur in parallel with one another (and in series with conduction/convection through the liquid film to the structure surface).

Heat transfer through the gas boundary layer is accounted for by a heat transfer coefficient obtained from correlations for natural or forced convection heat transfer. User input must specify whether an internal flow or external flow correlation is to be used when calculating atmosphere heat transfer coefficients for each boundary surface. These correlations are given in Section 2.6.1 for atmosphere heat transfer. Section 2.6.1.1 describes the modeling of heat transfer through liquid films when the film tracking model is inactive. The modeling of heat transfer through liquid films flowing over structures included in user-defined film tracking networks is discussed in Section 2.6.1.2.

Radiative heat transfer between a heat structure surface and the boundary volume atmosphere is modeled in either of two ways. The user has the option of employing the equivalent band model or the gray gas model for radiative heat transfer. These models are presented in Section 2.6.2.

At any surface, an arbitrary, user-specified nonnegative scaling factor may be applied to the calculated convective and radiative heat transfer coefficients to the atmosphere. The user may also apply a separate, arbitrary nonnegative scaling factor to the condensation/evaporation mass transfer coefficient. Users are cautioned that the application of significantly different heat and mass transfer scaling factors at the same surface may lead to nonphysical results and numerical problems. The scaling factors are provided primarily for conducting sensitivity studies associated with uncertainties related to surface fouling, local fluid effects, etc.

The HS package calculates heat transfer between a heat structure and the boundary volume pool whenever the pool fraction at a boundary surface is greater than or equal to its critical pool fraction for pool heat transfer. Pool heat transfer can be by natural convection, forced convection, or pool boiling. The HS package uses an extensive set of correlations for natural or forced convection pool heat transfer. User input must again specify whether an internal flow or external flow correlation is to be used when calculating pool heat transfer coefficients for each boundary surface. These correlations are exhibited in Section 2.6.3.

Pool boiling heat transfer is calculated at a surface if its pool fraction is greater than the critical pool fraction and its temperature is greater than the saturation temperature of its boundary volume (at total pressure). In calculating pool boiling heat transfer, the HS package uses a set of correlations for nucleate boiling, critical heat flux, minimum film boiling, and stable film boiling. Radiative heat transfer between a surface and the pool of

its boundary volume is calculated during stable film and transition boiling. Correlations for pool boiling heat transfer as well as models for pool radiation heat transfer are discussed in Section 2.6.4.

The HS package obtains the boiling heat transfer coefficient at a boundary surface as the quotient of the boiling heat flux and the difference between the temperature of this surface and the saturation temperature of its boundary volume (at total pressure).

The correlation of experimental heat transfer data is usually accomplished with dimensionless variables that are obtained by dimensional analysis or physical reasoning. These variables include:

Reynolds number (Re) = $\rho V L_c / \mu$

Prandtl number (Pr) = $\mu c_p / k$

Grashof number (*Gr*) = $g \beta \Delta t L_c^3 \rho^2 / \mu^2$

$g \left| \rho_{srf} - \rho \right| L_c^3 \rho / \mu^2$ (during condensation)

Nusselt number (*Nu*) = $h L_c / k$

Rayleigh number (*Ra*) = *Gr*•*Pr*

where

ρ = density of atmosphere (pool), kg/m^3

ρ_{srf} = density of atmosphere evaluated at film surface temperature, kg/m^3

V = velocity of atmosphere (pool), m/s

L_c = characteristic length of surface, m

μ = viscosity of atmosphere (pool), kg/m•s

c_p = heat capacity at constant pressure of atmosphere (pool), J/kg•K

k = thermal conductivity of atmosphere (pool), W/m•K

g = acceleration of gravity, m/s^2

β = volume coefficient of expansion of atmosphere (pool), K^{-1}

Δt = magnitude of difference between temperatures of surface and atmosphere (pool), K

h = atmosphere (pool) heat transfer coefficient, W/m^2•K

The HS package uses these variables for selecting the appropriate heat transfer correlation or in expressing the functional form of the correlation for all heat transfer regimes except pool boiling. The pool boiling correlations are not expressed in a dimensionless form.

2.6.1 Atmosphere Convection Heat Transfer

Natural, forced, or mixed convection heat transfer to the atmosphere is determined at a surface by the following criteria:

Region	Criteria	Equation
Natural Convection	$Re^2 < 1.0\ Gr$	(2-79)
Forced Convection	$Re^2 > 10.0\ Gr$	(2-80)
Mixed Convection	$1.0\ Gr \leq Re^2 \leq 10.0\ Gr$	(2-81)

where

Re = Reynolds number for atmosphere

Gr = Grashof number for atmosphere

Ra = Rayleigh number for atmosphere

The constants in Equations (2-79) through (2-81) are implemented as sensitivity coefficient array C4060.

The atmosphere natural convection heat transfer correlations have the following form:

$$Nu = C\,Ra^{m} + D \qquad (2\text{-}82)$$

where

Nu = Nusselt number

Ra = Rayleigh number

C,m,D = constants dependent on flow condition and geometry

The constants *C*, *m*, and *D* in Equation (2-82) have been implemented as sensitivity coefficient arrays C4101 – C4112 and are presented in Table 2.3 for the various flow conditions and geometries.

Table 2.3 Constants for HS Package Heat Transfer Correlations: Atmosphere

Region	Type of Flow		Geometry	(1)	(2)	(3)	(4)	Ref	SC Array
ATMOSPHERE									
Atmosphere	Natural Convection	Internal	Laminar	Rectangular	0.046	1/3	0	-	[1]	C4101
				Cylindrical	0.046	1/3	0	-	[1]	C4102
				Spherical	0.228	0.226	0	-	[1]	C4103
			Turbulent	Rectangular	0.046	1/3	0	-	[1]	C4104
				Cylindrical	0.046	1/3	0	-	[1]	C4105
				Spherical	0.228	0.226	0	-	[1]	C4106
		External	Laminar	Rectangular	0.59	0.25	0	-	[1]	C4107
				Cylindrical	0.59	0.25	0	-	[1]	C4108
				Spherical	0.43	0.25	2.0	-	[1]	C4109
			Turbulent	Rectangular	0.10	1/3	0	-	[1]	C4110
				Cylindrical	0.10	1/3	0	-	[1]	C4111
				Spherical	0.43	0.25	2.0	-	[1]	C4112
	Forced Convection	Internal	Laminar	Rectangular	8.235	0	0	0	[1]	C4113
				Cylindrical	48/11	0	0	0	[1]	C4114
				Spherical	48/11	0	0	0	[1]	C4115
			Turbulent	Rectangular	0.023	0.8	1/3	0	[2]	C4116
				Cylindrical	0.023	0.8	1/3	0	[2]	C4117
				Spherical	0.023	0.8	1/3	0	[2]	C4118
		External	Laminar	Rectangular	0.664	0.5	1/3	0	[2]	C4119
				Cylindrical	0.664	0.5	1/3	0	[2]	C4120
				Spherical	0.60	0.5	1/3	2.0	[2]	C4121
			Turbulent	Rectangular	0.037	0.8	1/3	0	[2]	C4122
				Cylindrical	0.037	0.8	1/3	0	[2]	C4123
				Spherical	0.60	0.5	1/3	2.0	[2]	C4124

The atmosphere forced convection heat transfer correlations have the following form:

$$Nu = C\,\mathrm{Re}^{m}\,\mathrm{Pr}^{n} + D \qquad (2\text{-}83)$$

where

Nu = Nusselt number

Re = Reynolds number

Pr = Prandtl number

C,m,n,D = constants dependent on flow condition and geometry

The constants C, m, n, and D in Equation (2-83) have been implemented as sensitivity coefficient arrays C4113 – C4124 and are presented in Table 2.3 for the various flow conditions and geometries.

The Nusselt number in the mixed convection regime is a linear interpolation between the Nusselt numbers for the natural and forced convection regimes, based on the ratio Re^2/Gr. That is,

$$Nu_{mixed} = \left[\left(\mathrm{Re}^2/Gr - 1\right)/9\right]\left[Nu_{forced} - Nu_{natural}\right] + Nu_{natural} \qquad (2\text{-}84)$$

The constants in Equation (2-84) are, of course, derived from the sensitivity coefficients that define the limits of natural and forced convection. If the values of these coefficients do not define a proper transition—specifically if the upper limit for natural convection, C4060(1) (default value 1.0) is negative or is greater than or equal to the lower limit for forced convection, C4060(2) (default value 10.0)—no mixed convection regime is considered. Instead, convection heat transfer to the atmosphere is assumed to be given by the greater of the values defined by the natural and forced convection correlations. This simple and often-used treatment may be specified in MELCOR by deliberate modification of the sensitivity coefficients.

Laminar or turbulent natural convection heat transfer to the atmosphere is determined at a surface by the following criteria:

Region	Criteria	Equation
Laminar Natural Convection	$Ra < 10^9$	(2-85)
Turbulent Natural Convection	$Ra > 10^{10}$	(2-86)
Transition between Laminar and Turbulent Natural Convection	$10^9 \leq Ra \leq 10^{10}$	(2-87)

The constants in Equations (2-85) through (2-87) are implemented as sensitivity coefficient arrays C4061 – C4063 for rectangular, cylindrical, and spherical (hemispherical) geometries.

Laminar or turbulent forced convection heat transfer to the atmosphere is determined at a surface by the following criteria:

Region	Criteria	Equation
Laminar Forced Convection	$Re < 3x10^5$ (rectangular)	(2-88a)
	$Re < 2x10^3$ (cylindrical/spherical)	(2.88b)
Turbulent Forced Convection	$Re > 6x10^5$ (rectangular)	(2-89a)
	$Re > 1x10^4$ (cylindrical/spherical)	(2.89b)
Transition between Laminar and Turbulent Forced Convection	$3\times10^5 \le Re \le 6\times10^5$ (rectangular)	(2-90a)
	$2\times10^3 \le Re \le 1\times10^4$ (cylindrical/spherical)	(2.90b)

The constants in Equations (2-88) through (2-90) are implemented as sensitivity coefficient arrays C4064 – C4066 for rectangular, cylindrical, and spherical (hemispherical) geometries.

The Nusselt number in the transition region is a linear interpolation between the Nusselt numbers for the laminar and turbulent regimes. The interpolation is based on the Rayleigh number for natural convection and the Reynolds number for forced convection. An example is the Nusselt number in the transition region for forced convection with rectangular geometries:

$$Nu_{transition} = \left[\left(Re - 3x10^5\right)/3x10^5\right]\left[Nu_{turbulent} - Nu_{laminar}\right] + Nu_{laminar} \qquad (2\text{-}91)$$

The constants in Equation (2-91) are, of course, derived from the sensitivity coefficients that define the limits of laminar and turbulent convection. If the values of these coefficients do not define a proper transition—specifically, if the upper limit for laminar convection, C406m(1) is negative or is greater than or equal to the lower limit for turbulent convection, C406m(2)—no transition regime is considered. Instead, convection heat transfer to the atmosphere is assumed to be given by the greater of the values defined by the laminar and turbulent convection correlations. This simple and often-used treatment may be specified in MELCOR by deliberate modification of the sensitivity coefficients.

2.6.1.1 Conduction/Convection through Liquid Films (film tracking inactive)

Liquid film modeling is discussed in detail in Section 2.8. Heat transfer through a liquid film is accounted for by a heat transfer coefficient, H_f, which is used in Equations (2-59) and (2-60a). The value of H_f used when the structure is included in a user-defined film tracking network is discussed in Section 2.6.1.2. When film tracking is inactive, the value of H_f used is the greater of two values:

(1) a value obtained from a steady-state correlation appropriate for the geometry and film conditions (zero is used if no film exists) and

(2) the quotient of the thermal conductivity of the liquid and the transient film thickness. Thus, the liquid film heat transfer coefficient is given by

$$H_f = max\left(H_{f,corr}, k_f/\delta_f\right) \tag{2-92}$$

where

k_f = thermal conductivity of liquid film, W/m•K

δ_f = liquid film thickness, m

and $H_{f,corr}$ is a function of surface geometry and film flow conditions. Laminar or turbulent heat transfer through the condensate film is determined by the following criteria:

Laminar if $Re_f < Re_{LOW,m}$
Turbulent if $Re_f > Re_{HIGH,m}$
Transition if $Re_{LOW,m} \leq Re_f \leq Re_{HIGH,m}$

where

Re_f = Reynolds number for the film flow

The laminar heat transfer coefficient through the film, $h_{f,l}$, is given by

$$h_{f,l} = \left(k_f / L\right) Nu_{f,l}$$

where the laminar film Nusselt number, $Nu_{f,l}$, is given by

$$Nu_{f,l} = C_{l,m}\left\{g\ \rho_f\left(\rho_f - \rho_v\right)h_{fg}L^3 \sin(\theta)/\left[\mu_f k_f\left(T_{sat} - T_{srf}\right)\right]\right\}^{el,m}$$

The turbulent heat transfer coefficient through the film, $h_{f,t}$, is given by

$$h_{f,t} = \left\{k_f /\left[\left(\mu_f / \rho_f\right)^2 / g\right]^{et1,m}\right\} Nu_{f,t}$$

where the turbulent film Nusselt number, $Nu_{f,t}$ is given by

$$Nu_{f,t} = \left(\mathrm{Re}_f^{et2,m} + C_{t,m}\,\mathrm{Re}_f^{et3,m}\,\mathrm{Pr}_f^{et4,m}\right)^{et5,m}$$

The transition heat transfer coefficient through the film, $h_{f,tr}$, is given by linear interpolation of Re_f as

$$h_{f,tr} = h'_{f,l} + \left[h'_{f,t} - h'_{f,l}\right] \cdot \left[\mathrm{Re}_f - \mathrm{Re}_{LOW,m}\right] / \left[\mathrm{Re}_{HIGH,m} - \mathrm{Re}_{LOW,m}\right]$$

In each of these equations,

k_f = thermal conductivity of film

L = characteristic length of surface

ρ_f = density of film

ρ_v = density of vapor

g = acceleration of gravity

h_{fg} = latent heat of vaporization corrected for sensible heat

$\left|h_{fg} + 0.68\, c_{p,f}\left(T_f - T_{srf}\right)\right|$

$c_{p,f}$ = specific heat capacity of film

T_f = temperature of film/atmosphere interface

T_{srf} = temperature of film/structure interface

μ_f = viscosity of film, kg/m•s

θ = angle between horizontal and structure surface or axis (cyl.)

$h'_{f,l}$ = $h_{f,l}$ evaluated with Re_f = C42m0(1)

$h'_{f,t}$ = $h_{f,t}$ evaluated with Re_f = C42m0(2)

and

m = 1 for upward-facing rectangular geometries

= 2 for horizontal cylindrical geometries

= 3 for spherical or hemispherical geometries

$\mathrm{Re}_{LOW,m}$, $\mathrm{Re}_{HIGH,m}$ and the minimum permissible value of $\sin(\theta)$ ($\cos(\theta)$ for cylindrical geometry) have been implemented as sensitivity coefficients C42m0. $C_{l,m}$ and *el*,*m* have been implemented as sensitivity coefficients C42m1, and $C_{t,m}$ and *etn*,*m* have been implemented as sensitivity coefficients C42m2.

For downward-facing rectangular geometries, the laminar/turbulent transition criteria are given by:

Laminar if $Ra_f < Ra_{TRAN}$
Turbulent, otherwise

where

Ra_f = Rayleigh number for the film flow

The heat transfer coefficient through the film is given by

$$h_f = \left(k_f / \{\sigma_f / [g(\rho_f - \rho_v)\cos(\theta)]\}^{1/2}\right) Nu_f$$

where the film Nusselt number is given by

$$Nu_f = C_{l,4}\left[\max\left(Ra_{MIN}, Ra_f\right)\right]^{el,4}$$

for laminar flow, and by

$$Nu_f = C_{t,4}\left[\min\left(Ra_{MAX}, Ra_f\right)\right]^{et,4}$$

for turbulent film flow. Ra_{TRAN}, Ra_{MIN}, Ra_{MAX} and the minimum value of $\cos(\theta)$ have been implemented as sensitivity coefficients C4213, $C_{l,4}$ and el,4 have been implemented as sensitivity coefficients C4214 and $C_{t,4}$ and et,4 have been implemented as sensitivity coefficients C4215.

Early in its formation, the transient film thickness determines the rate of heat transfer; while its steady-state value is limited by the greater of the correlation value or $k_f / \delta_{\max}$ where $\delta_{\max}$ is the user-specified maximum film thickness discussed in Section 2.8.1 below. Note, that because the film convective heat transfer correlations are functions of the flow conditions, and the flow conditions are a function of the rate of heat transfer, the convective heat transfer coefficient through the film must be determined iteratively as part of the overall solution for the temperature profile through a heat structure and its associated films. This is implied by use of the new time superscript (m+1), e.g., $H_{f,L}^{m+1}$.

2.6.1.2 Conduction/Convection through Liquid Films (Film Tracking Active)

Section 2.8.2 discusses the film tracking model. This section only describes the correlations used to evaluate heat transfer through films being treated by the film tracking model. The correlations used are the same for all geometries and treat both laminar and turbulent film flow conditions. Laminar or turbulent heat transfer through the condensate film is determined by the following criteria:

Laminar if $Re_f < Re_{LOW}$
Turbulent if $Re_f > Re_{HIGH}$
Transition if $Re_{LOW} \leq Re_f \leq Re_{HIGH}$

where

Re_f = Reynolds number for the film flow

The laminar heat transfer coefficient through the film, $h_{f,l}$, is given by

$$h_{f,l} = k_f / \max\left(\delta_{f,l}, \delta_{\min}\right)$$

where the laminar film thickness, $\delta_{f,l}$, is obtained from the film tracking model solution, and $\delta_{\min}$ is the user-adjustable minimum film thickness implemented as sensitivity coefficient C4251(1) (see Section 2.8.1). The turbulent heat transfer coefficient through the film, $h_{f,t}$, is given by

$$h_{f,t} = \left\{k_f / \left[\left(\mu_f / \rho_f\right)^2 / \left(g \cdot \sin(\theta)\right)\right]^{1/3}\right\} Nu_{f,t}$$

where the turbulent film Nusselt number, $Nu_{f,t}$, is given by

$$Nu_{f,t} = \left(Re_f^{et1} + C_t\, Re_f^{et2}\, Pr_f^{et3}\right)^{et4}$$

The transition heat transfer coefficient through the film, $h_{f,tr}$, is given by linear interpolation of Re_f as

$$h_{f,tr} = h'_{f,l} + \left[h'_{f,t} - h'_{f,l}\right] \cdot \left[Re_f - Re_{LOW}\right] / \left[Re_{HIGH} - Re_{LOW}\right]$$

In each of these equations,

k_f = thermal conductivity of film, W/m-K

ρ_f = density of film, kg/m^3

g = acceleration of gravity, m/s^2

μ_f = viscosity of film, kg/m•s

θ = angle between horizontal and structure surface or axis (cyl.)

$h'_{f,l}$ = $h_{f,l}$ evaluated with Re_f=C4253(5)

$h'_{f,t}$ = $h_{f,t}$ evaluated with Re_f=C4253(6)

Re_{LOW}, Re_{HIGH}, C_t, et1, et2, et3, and et4 have been implemented as sensitivity coefficients c4253(5), c4253(6), c4253(8), c4253(7), c4253(9), c4253(10), c4253(11), respectively.

2.6.2 Radiative Heat Transfer

Simple models are available to determine the energy exchanges between a heat structure surface and the surrounding atmosphere and between the surfaces of heat structures. These are discussed below.

2.6.2.1 Atmosphere Radiative Heat Transfer

In addition to the convective boundary condition options, radiative heat transfer between the surface and the boundary volume atmosphere can be specified. Two options are currently available. They are:

(1) Equivalent band model, and

(2) Gray gas.

The equivalent band model is based on work by Edwards et al. [3] [4] in which the total radiation properties can be used to adequately calculate radiative heat transfer without resorting to a band model. The equivalent band equation is:

$$q_{EB} = \begin{Bmatrix} \sigma(F_g T_g^4 - F_{gw} T_w^4) \text{ for } T_g \neq T_w \\ 0 \text{ for } T_g = T_w \end{Bmatrix} \quad (2\text{-}93)$$

where

σ = Stefan-Boltzmann constant

F_g = $\varepsilon_w \varepsilon_{g1} / (1 - \rho_w \tau_{gb})$

F_{gw} = $\varepsilon_w \alpha_{gw1} / (1 - \rho_w \tau_{gbw})$

τ_{gb} = $(\varepsilon_{g2} - \varepsilon_{g1}) / \varepsilon_{g1}$

τ_{gbw} = $(\alpha_{gw2} - \alpha_{gw1}) / \alpha_{gw1}$

and ε and α are the emissivity and absorptivity, respectively. Subscripts *g*, *w*, *gw*, 1, and 2 refer to the gas, the wall, the gas at the wall temperature, one path length, and two path lengths. The values of the gas emissivity (ε_g) and absorptivity (α_{gw}), obtained from the model in CONTAIN [5], are functions of the gas composition, including the pressure of

water vapor, CO, and CO_2 as well as the radiation path length, which is user specified. The wall emissivity ε_w, is given by user input, $\varepsilon_{w,user}$, that is overwritten if a liquid film is present. The emissivity of the film-covered wall becomes

$$\varepsilon_w = 1 - \rho_f - \rho_w \tau_f / (1 - \rho_f \rho_w) \tag{2-94a}$$

where

$$\tau_f = \exp(-1000\delta) \tag{2-94b}$$

$$\rho_f = (1 - \varepsilon_{H2O})(1 - \tau_f) \tag{2-94c}$$

$$\rho_w = 1 - \varepsilon_{w,user} \tag{2-94d}$$

$$\varepsilon_{H2O} = 0.96$$

and δ is the film thickness in meters.

The gray gas model equation is:

$$q_{GG} = \sigma\left[(1/\varepsilon_g) + (1/\varepsilon_w) - 1\right]^{-1}\left(T_g^4 - T_w^4\right) \tag{2-95}$$

where the gas emissivity is calculated for one path length.

2.6.2.2 Structure-to-Structure Radiation Heat Transfer

Structure-to-structure radiation can be calculated by a simple gray surface model. This optional model assumes that the radiative exchange between pairs of surfaces is independent and decoupled from the exchanges involved with other surfaces or with the intervening atmosphere. This permits sequential processing for each pair and does not require the use of iterative or simultaneous solution techniques. The net radiative heat loads for the surfaces of each heat structure are entered explicitly into the surface nodal energy balances similar to the method described in Section 2.3.2 for surface power sources.

An arbitrary number of heat structure surface pairs may be defined by the user and radiative exchange calculated between the surfaces of each pair by the following relationship:

$$q_{12} = \frac{\sigma\left(T_1^4 - T_2^4\right)}{\frac{1-\varepsilon_1}{\varepsilon_1 A_1} + \frac{1}{A_1 F_{12}} + \frac{1-\varepsilon_2}{\varepsilon_2 A_2}} \tag{2-96}$$

where

q_{12} = radiative energy transfer rate from surface 1 to surface 2 (W)
σ = Stefan-Boltzmann constant (W/m^2•K^4)
T_1 = surface 1 temperature (K)
T_2 = surface 2 temperature (K)
ε_1 = surface 1 emissivity
ε_2 = surface 2 emissivity
A_1 = surface 1 area (m^2)
A_2 = surface 2 area (m^2)
F_{12} = view factor from surface 1 to surface 2

The emissivities may be computed by a default relation (from the COR package for oxidized steel surfaces) or may be computed by evaluation of user-specified real-valued control functions. A modification to account for the presence of a water film on either surface is applied and is the same as that described in Section 2.6.2.1 for radiative exchanges with the atmosphere. The areas used in the above equation correspond to the uncovered portions above the swollen liquid level of the adjacent CVH control volume. Radiative energy exchange between the surfaces of a pair is not calculated (i.e., q_{12} is set to 0.0) when:

(1) either of the surfaces is covered by a pool,

(2) either of the surface emissivities is determined to be zero, or

(3) the input view factor is zero.

2.6.2.3 Radiation Enclosure Model

A radiation enclosure consists of two or more surfaces that envelope a region of space for which radiation transfer occurs among those surfaces. The space between these surfaces may or may not be filled with a participating medium, for which the gas may absorb, emit, and scatter radiation emitted by the surfaces. Each surface is assumed to be isothermal, opaque, diffuse, and gray, and are characterized by uniform radiosity. Consequently, the absorptivity (α) of a surface is equal to the emissivity (ε) and the sum of the absorptivity and reflectivity (ρ) is 1.0.

$$\varepsilon_i = \alpha_i = 1 - \rho_i \qquad (2\text{-}97)$$

Reciprocity is also assumed between surface pairs and it is assumed the sum of the view factors from a surface to all surfaces in the enclosure, is equal to 1.0. and a surface may also radiate to itself.

$$\sum_{i=1}^{N} VF_{i,j} = 1.0 \qquad (2\text{-}98)$$

The surface radiosity is defined as the total heat flux that departs from an area (reflected and emitted) and is labeled J_i:

$$J_i = \rho_i \cdot G_i + \varepsilon_i E_{b,i} \qquad (2\text{-}99)$$

where

G_i= radiation flux incident on surface I from radiation from all other surfaces,

E_{bi}= blackbody emissive power of surface i, σT_i^4

The radiation flux incident on a surface is calculated by summing the radiosity from all surfaces, reduced by multiplying by the configuration view factor between surfaces and the transmissivity through the intermediate gas and also includes the emission from the gas:

$$G_i = \sum_{j}^{N} \left[A_j \cdot F_{j,i} \cdot \tau_{j,i} \cdot J_j \right] / A_i + \varepsilon_m E_{bm} \qquad (2\text{-}100)$$

The net heat transfer rate from the i-th surface is the difference between the radiosity and the incident radiation, multiplied by the area of surface i, A_i:

$$q_i = A_i \left(J_i - G_i \right) \qquad (2\text{-}101)$$

The reciprocity relationship can be used to eliminate the surface areas in these equations. Combining the above two equations gives an equation for the radiosity at a surface, J_i in terms of the radiosity at all surfaces and blackbody emissive power:

$$J_i = \left(1 - \varepsilon_i\right) \cdot \sum_{j}^{N} \left[F_{ij} \cdot \tau_{j,i} \cdot J_j \right] + \varepsilon_i \cdot \sigma \cdot T_i^{\,4} + \rho_i \varepsilon_m E_{bm} \qquad (2\text{-}102)$$

The transmissivity and the emissivity of the gas is calculated using the existing COR package routine for steam. This matrix equation is solved for each Ji, which can then be used to calculate the heat transfer from each surface (See Section 2.1.1 for the COR package).

2.6.3 Pool Convection Heat Transfer

Natural, forced, or mixed convection heat transfer to the pool is determined at a surface by the following criteria:

Region	Criteria	Equation
Natural Convection	$\mathrm{Re}^2 < 1.0Gr$	(2-103)
Forced Convection	$\mathrm{Re}^2 > 10.0Gr$	(2-104)
Mixed Convection	$1.0Gr \leq \mathrm{Re}^2 \leq 10.0Gr$	(2-105)

where

Re = Reynolds number for pool

Gr = Grashof number for pool

Ra = Rayleigh number for pool

The constants in Equations (2-103) through (2-105) are implemented as sensitivity coefficient array C4080.

The pool natural convection heat transfer correlations have the following form:

$$Nu = C\,Ra^m + D \qquad (2\text{-}106)$$

where

Nu = Nusselt number

Ra = Rayleigh number

C,m,D = constants dependent on flow condition and geometry

The constants *C*, *m*, and *D* in Equation (2-106) have been implemented as sensitivity coefficient arrays C4151 – C4162 and default values are presented in Table 2.3 for the various flow conditions and geometries.

The pool forced convection heat transfer correlations have the following form:

$$Nu = C\,\mathrm{Re}^m\,\mathrm{Pr}^n + D \qquad (2\text{-}107)$$

where

Nu = Nusselt number

Re = Reynolds number

Pr = Prandtl number

C,m,n,D = constants dependent on flow condition and geometry

The constants *C*, *m*, *n*, and *D* in Equation (2-107) have been implemented as sensitivity coefficient arrays C4163 – C4174 and are presented in Table 2.3 for the various flow conditions and geometries.

Table 2.4 Constants for HS Package Heat Transfer Correlations: Pool

POOL									
			Rectangular	0.046	1/3	0	-	[1]	C4151
		Laminar	Cylindrical	0.046	1/3	0	-	[1]	C4152
			Spherical	0.028	0.226	0	-	[1]	C4153
	Internal								
			Rectangular	0.046	1/3	0	-	[1]	C4154
		Turbulent	Cylindrical	0.046	1/3	0	-	[1]	C4155
	Natural Convection		Spherical	0.228	0.226	0	-	[1]	C4156
			Rectangular	0.59	0.25	0	-	[1]	C4157
		Laminar	Cylindrical	0.59	0.25	0	-	[1]	C4158
			Spherical	0.43	0.25	2.0	-	[1]	C4159
	External								
			Rectangular	0.10	1/3	0	-	[1]	C4160
		Turbulent	Cylindrical	0.10	1/3	0	-	[1]	C4161
			Spherical	0.43	0.25	2.0	-	[1]	C4162
	Pool								
			Rectangular	8.235	0	0	0	[1]	C4163
		Laminar	Cylindrical	48/11	0	0	0	[1]	C4164
			Spherical	48/11	0	0	0	[1]	C4165
	Internal								
			Rectangular	0.023	0.8	1/3	0	[2]	C4166
		Turbulent	Cylindrical	0.023	0.8	1/3	0	[2]	C4167
	Forced Convection		Spherical	0.023	0.8	1/3	0	[2]	C4168
			Rectangular	0.664	0.5	1/3	0	[2]	C4169
		Laminar	Cylindrical	0.664	0.5	1/3	0	[2]	C4170
			Spherical	0.60	0.5	1/3	2.0	[2]	C4171
	External								
			Rectangular	0.037	0.8	1/3	0	[2]	C4172
		Turbulent	Cylindrical	0.037	0.8	1/3	0	[2]	C4173
			Spherical	0.60	0.5	1/3	2.0	[2]	C4174

The Nusselt number in the mixed convection regime is a linear interpolation between the Nusselt numbers for the natural and forced convection regimes, based on the ratio Re^2/Gr. This is the same method employed for atmosphere heat transfer, and an example is shown in Section 2.6.1. As with atmosphere heat transfer, the sensitivity coefficients defining the limits of natural and forced convection (sensitivity coefficient array C4080) may be chosen to eliminate the mixed convection regime for the pool in favor of use of the maximum of natural and forced convection heat transfer.

Laminar or turbulent natural convection heat transfer to the pool is determined at a surface by the following criteria:

Region	Criteria	Equation
Laminar Natural Convection	$Ra < 10^9$	(2-108)
Turbulent Natural Convection	$Ra > 10^{10}$	(2-109)
Transition between Laminar and Turbulent Natural Convection	$10^9 \leq Ra \leq 10^{10}$	(2-110)

The constants in Equations (2-108) through (2-110) are implemented as sensitivity coefficient arrays C4081 – C4083 for rectangular, cylindrical, and spherical (hemispherical) geometries.

Laminar or turbulent forced convection heat transfer to the pool is determined at a surface by the following criteria:

Region	Criteria	Equation
Laminar Forced Convection	$Re < 3x10^5$ (rectangular)	(2-111a)
	$Re < 2x10^3$ (cylindrical/spherical)	(2.111b)
Turbulent Forced Convection	$Re > 6x10^5$ (rectangular)	(2-112a)
	$Re > 1x10^4$ (cylindrical/spherical)	(2.112b)
Transition between Laminar and Turbulent Forced Convection	$3\times10^5 \leq Re \leq 6\times10^5$ (rectangular)	(2-113a)
	$2\times10^3 \leq Re \leq 1\times10^4$ (cylindrical/spherical)	(2.113b)

The constants in Equations (2-111) through (2-113) are implemented as sensitivity coefficient arrays C4084 – C4086 for rectangular, cylindrical, and spherical (hemispherical) geometries.

The Nusselt number in the transition region is a linear interpolation between the Nusselt numbers for the laminar and turbulent regimes. The interpolation is based on the Rayleigh number for natural convection and the Reynolds number for forced convection. This is the same method employed for atmosphere heat transfer, and an example is shown in Section 2.6.1. As with atmosphere heat transfer, the sensitivity coefficients defining the limits of laminar and turbulent convection (sensitivity coefficient arrays C408m) may be chosen to eliminate the transition regime for the pool in favor of use of the maximum of laminar and turbulent convection heat transfer.

2.6.4 Pool Boiling Heat Transfer

If a heat structure is submerged in a pool or a film is present and the heat structure surface temperature, T_{surf}, is greater than the saturation temperature, T_{sat}, at the total control volume pressure, pool boiling heat transfer from the heat structure is assumed. Using the heat structure surface temperature and various liquid properties, the logic for choosing the appropriate pool boiling regime is given by:

Nucleate boiling (Rohsenow) is calculated if

$$q''_{nb} \text{ (Rohsenow)} \leq q''_{chf} \text{ (Zuber)}$$

Film boiling (modified Bromley) is calculated if

$$q''_{film} \text{ (modified Bromley)} \geq q''_{mfilm} \text{ (Zuber)}$$

where

q''_{nb} = nucleate boiling heat flux given by Equation (2-114), W/m^2

q''_{chf} = critical heat flux given by Equation (2-116), W/m^2

q''_{film} = film boiling heat flux give by Equation (2-118), W/m^2

q''_{mfilm} = minimum film boiling heat flux given by Equation (2-117), W/m^2

If neither of these conditions is met, the surface is in transition boiling and a linear interpolation of the surface temperature is used to determine the heat flux at that temperature.

For all the above cases, once a heat flux has been determined, an effective heat transfer coefficient is evaluated as the ratio of heat flux over the difference between the surface and pool temperatures. This heat transfer coefficient is used as the boundary heat transfer coefficient in the solution of the heat conduction equations.

2.6.4.1 Nucleate Boiling

The nucleate boiling heat flux is obtained through the Rohsenow relation [6]

$$\left[\frac{c_{pl}\left(T_{surf}-T_{sat}\right)}{h_{fg}}\right]=C_{sf}\left[\frac{q''_{nb}}{\mu h_{fg}}\left(\frac{\sigma}{g(\rho_l-\rho_v)}\right)^{1/2}\right]^n \mathrm{Pr}^m \qquad (2\text{-}114)$$

where

q''_{nb} = nucleate boiling heat flux, W/m^2

c_{pl} = heat capacity of liquid at T_{sat}, J/kg•K

T_{surf} = temperature of surface, K

T_{sat} = saturation temperature in boundary volume, K

C_{sf} = constant determined empirically for different surfaces and fluids (default = 0.013)

μ = dynamic viscosity of liquid at T_{avg}, kg/m•s

h_{fg} = latent heat in boundary volume of this surface, J/kg

σ = surface tension at T_{avg}, N/m

g = acceleration of gravity, m/s^2

ρ_l = density of liquid at T_{sat}, kg/m^3

ρ_v = density of vapor at T_{sat}, kg/m^3

n = constant (default = 0.33)

Pr = Prandtl number of liquid in boundary volume

m = constant (default = 1.0)

T_{avg} = $(T_{surf} + T_{sat}) / 2$, K

The constants C_{sf}, m, and n in Equation (2-114) have been implemented as sensitivity coefficient array C4180.

The surface tension of water is given as a function of temperature by

$$\sigma = 0.2358\left(1 - 0.625\,T_R\right)T_R^{1.256} + c \qquad (2\text{-}115)$$

where

σ = surface tension, N/m

T = temperature, K

T_R = 1 – T / 647.3

c = constant (default = 0.0)

The constants in Equation (2-115), including c, have been implemented as sensitivity coefficient array C4000.

2.6.4.2 Critical Heat Flux

The critical heat flux is given by

$$q''_{chf} = 0.18\rho_v h_{fg}[\sigma(\rho_l - \rho_v)g/\rho_v^2]^{1/4}[\rho_l/(\rho_l + \rho_v)]^{1/2} \quad (2\text{-}116)$$

where

q''_c = critical heat flux, W/m^2

ρ_v = density of vapor at T_{sat}, kg/m^3

ρ_l = density of liquid at T_{sat}, kg/m^3

h_{fg} = latent heat in boundary volume, J/kg

g = acceleration of gravity, m/s^2

σ = surface tension at T_{avg}, N/m

T_{avg} = (T_{surf} + T_{sat}) / 2, K

T_{sat} = saturation temperature in boundary volume, K

T_{surf} = temperature of this surface, K

The constants in Equation (2-116) have been implemented as sensitivity coefficient array C4181. Zuber gives a leading coefficient of 0.131, while 0.18 is the value suggested by Rohsenow, see Reference [5].

2.6.4.3 Minimum Film Boiling Heat Flux

The minimum film boiling heat flux is given by Zuber [5] as

$$q''_{mfilm} = 0.09\rho_v h_{fg}\left[\sigma(\rho_l - \rho_v)g/\rho_l^2\right]^{1/4}[\rho_l/(\rho_l + \rho_v)]^{1/2} \quad (2\text{-}117)$$

where

q''_{mfilm} = minimum film boiling heat flux, W/m^2

The constants in Equation (2-117) have been implemented as sensitivity coefficient array C4182.

2.6.4.4 Stable Film Boiling

The film boiling heat flux is given by Bromley [5] as

$$q''_{film} = 0.943\left[\rho_v(\rho_l - \rho_v)g\ k_v^3\left(h_{fg} + (1/2)c_{pv}\Delta T\right)/\mu_v L_c\right]^{1/4}\Delta T^{0.75} \qquad (2\text{-}118)$$

where

q''_{film} = film boiling heat flux, W/m^2

L_c = characteristic length of this surface, m

ΔT = $T_{surf} - T_{sat}$, K

T_{surf} = temperature of this surface, K

T_{sat} = saturation temperature in boundary volume, K

g = acceleration due to gravity, m/s^2

h_{fg} = latent heat in boundary volume, J/kg

ρ_l = density of liquid at T_{sat}, kg/m^3

ρ_v = density of vapor at T_{sat}, kg/m^3

c_{pv} = heat capacity of vapor at T_{sat}, J/kg•K

T_{avg} = (T_{surf} + T_{sat}) / 2, K

μ_v = dynamic viscosity of vapor at T_{avg}, kg/m•s

k_v = thermal conductivity of vapor at T_{avg}, W/m•K

The constants in Equation (2-118) have been implemented as sensitivity coefficient array C4183.

2.6.4.5 Transition Boiling

If transition boiling occurs at a surface, the heat flux is calculated as follows. First the surface temperatures at critical heat flux and minimum film boiling are calculated from

$$T_c = T_{sat} + (q''_c \Delta T^3 / q''_{NB})^{1/3} \qquad (2\text{-}119)$$

$$T_{mfilm} = T_{sat} + \left[q''_{mfilm}\Delta T^{0.75} / \left(q''_{film} + q''_{rad}\right)\right]^{4/3} \qquad (2\text{-}120)$$

where

T_{sat} = saturation temperature in boundary volume, K

ΔT = $T_{surf} - T_{sat}$, K

q''_c = critical heat flux given by Equation (2-116), W/m^2

q''_{NB} = nucleate boiling heat flux given by Equation (2-114), W/m^2

q''_{mfilm} = minimum film boiling heat flux given by Equation (2-117), W/m^2

q''_{film} = film boiling heat flux given by Equation (2-118), W/m^2

q''_{rad} = radiation to pool heat flux given by Equation (2-122), W/m^2

The constants in Equations (2-119) and (2-120) are sensitivity coefficients 4180(4) and 4183(3), respectively.

With these temperatures known, the transition boiling heat flux is then obtained by logarithmic interpolation between the critical heat flux and the minimum film boiling heat flux based on ($T - T_{sat}$) values and includes the radiation heat flux. Therefore, after simplification, the transition boiling heat flux is given by

$$q''_{tran} = \exp\left(\left[\ln\left(q''_c\right)\left(\ln\Delta T_{surf} - \ln\Delta T_{mfilm}\right) + \ln\left(q''_{mfilm}\right)\left(\ln\Delta T_c - \ln\Delta T_{surf}\right)\right] / \left[\ln\Delta T_c - \ln\Delta T_{mfilm}\right]\right) + q''_{rad} \quad (2\text{-}121)$$

where

q''_{tran} = transition boiling heat flux, W/m^2

T_c = critical temperature, K

ΔT_{surf} = T_{surf} - T_{sat}, K

ΔT_{mfilm} = $T_{mfilm} - T_{sat}$, K

ΔT_c = $T_c - T_{sat}$, K

q''_{rad} = radiation heat flux calculated by Equation (2-122), W/m^2

2.6.4.6 Radiation During Boiling

Radiation heat transfer between a surface and the boundary volume pool is calculated during stable film and transition boiling. The radiation to pool heat flux is given by

$$q''_{rad} = C\ \sigma\left(T^4_{surf} - T^4_{pool}\right) \quad (2\text{-}122)$$

where

q''_{rad} = radiation to pool heat flux, W/m^2

T_{surf} = temperature of surface, K

T_{pool} = temperature of pool in boundary volume, K

σ = Stefan-Boltzmann constant, 5.669×10^{-8} W/m^2•K^4

The constant C in Equation (2-122) defaults to 1.0 and has been implemented as sensitivity coefficient array C4184.

2.6.5 Energy Transfer to Control Volumes

The energy that is transferred from a heat structure surface to the boundary volume pool is:

$$\Delta Q^m_{pool} = q''^m_{pool} x_{pool} A \, \Delta t_m \tag{2-123}$$

Likewise, the energy that is transferred from a heat structure surface to the boundary volume atmosphere is:

$$\Delta Q^m_{atm} = q''^m_{atm} (1 - x_{pool}) A \, \Delta t_m \tag{2-124}$$

where

ΔQ^m_{pool} = energy transferred between heat structure surface and pool between times t_{m-1} and t_m, J

ΔQ^m_{atm} = energy transferred between heat structure surface and atmosphere between times t_{m-1} and t_m, J

q''^m_{pool} = heat flux to pool at time t_m, W/m^2

q''^m_{atm} = heat flux to atmosphere at time t_m, W/m^2

A = heat structure boundary surface area, m^2

x_{pool} = fraction of boundary surface in pool of boundary volume

Δt_m = timestep size ($t_m - t_{m-1}$), s

These time-surface integrals are evaluated at each boundary surface to determine the total energy transferred between each heat structure and its respective boundary volume atmosphere and pool. These integrals are used to update the energy communication arrays for the CVH package.

2.6.6 Helical Steam Generator Heat Transfer Correlations

The heat transfer correlation for subcooled water flow within the tubes, given forced-convection, is shown in Equation (2-125).

$$\mathrm{h} = \frac{1}{41}\left(\frac{k}{d_i}\right) Pr^{0.4}\, Re^{5/6} \left(\frac{d_i}{D_c}\right)^{1/12} \left[1 + \frac{0.061}{\{\, Re\, (\, d_i/D_c)^{2.5}\}^{1/6}}\right] \qquad (2\text{-}125)$$

where

k = heat conductivity (W/m - K)

d_i = inner diameter of a steam generator tube (m)

D_c = twisting diameter of helical steam generator tube = 1.35 m

The heat transfer for two-phase flow inside helical steam generator tubes are calculated by the following equations. The initial state represents the annular flow with nucleate boiling. As the film thins, the flow regime changes to evaporating film condition. Heat transfer coefficient for micro-convection by boiling is represented by h_b and h_c represents the heat transfer coefficient for macro-convection by flow. The two terms with the values of F and S are used for calculating the final heat transfer coefficient.

$$h_b = 0.00122 \left[\frac{k_f^{0.79}\, C_{pf}^{0.45}\, \rho_f^{0.49}}{\sigma^{0.25}\, \mu_f^{0.29}\, h_{fg}^{0.24}\, \rho_g^{0.24}}\right] \Delta T_{sat}^{0.24}\, \Delta P_{sat}^{0.75} \qquad (2\text{-}126)$$

$$h_c = 0.023 \left(\frac{k_f}{d_i}\right) (1-x)^{0.8}\, Re^{0.85}\, Pr^{0.4} \left(\frac{d_i}{D_c}\right)^{0.1} \qquad (2\text{-}127)$$

where,

k_f = (water) heat conductivity (W/m - K)

C_{pf} = (water) heat capacity (function of temperature and pressure) (J/kg – K)

ρ_f = (water) density (function of temperature and pressure) (J/kg – K)

σ = (water) surface tension (N/m)

μ_f = (water) viscosity (N s/m^2)

h_{fg} = latent heat of vaporization (J/kg)

ρ_g = (steam) density (kg/m^3)

x = flow quality (-), steam mass flow rate divided by total mass flow rate

ΔT_{sat} = $T_w - T_{sat}$ (K)

ΔP_{sat} = $P_{sat}^{Tsurf} - P_{sat}$ (Pa)

Re = Reynolds number (-), $Re_f = (\rho_f v_f d_i)/\mu_f$

Pr = Prandtl number (-), $Pr_f = (C_{pf}\ \mu_f)/k_f$

Function F is defined by Martinelli`s parameter as follows:

$$X_{tt} = \left(\frac{\rho_g}{\rho_f}\right)^{0.5} \left(\frac{\mu_f}{\mu_g}\right)^{0.1} \left(\frac{1}{x} - 1\right)^{0.9} \quad (2\text{-}128)$$

$$F = 2.35\left(\frac{1}{X_{tt}} + 0.213\right)^{0.736} \quad (2\text{-}129)$$

Reynolds number in two phase flow and function S are defined as follows:

$$Re_{tp} = \frac{G\ (1-x)\ d_i}{\mu_f}\ F^{1.25}\ (10^{-4}) \quad (2\text{-}130)$$

where,

G = mass flux, $\alpha\rho_g v_g + (1-\alpha)\rho_f v_f$

α = void fraction

and,

$$S = \begin{cases} (\ 1\ + 0.12\ (Re_{tp})^{1.14}\ )^{-1}, & Re_{tp} < 32.5 \\ \dfrac{G\ (1-x)\ d_i}{\mu_f}\ F^{1.25}\ (10^{-4}), & 32.5 \le Re_{tp} \le 70.0 \\ 0.0797, & 70.0 < Re_{tp} \end{cases} \quad (2\text{-}131)$$

Consequently, the heat transfer coefficient in a two-phase condition is calculated by F, S, h_b and h_c as follows:

$$h = S \cdot h_b\ + F \cdot h_c \quad (2\text{-}132)$$

The heat transfer coefficient in Equation (2-132) is calculated when the surface temperature is greater than the saturation temperature of the coolant. Then, the higher heat transfer coefficient between Equation (2-125) and Equation (2-132) is used.

The heat transfer coefficient for superheated steam in a forced-convection condition is calculated in Equation (2-133). Steam properties are used.

$$h = \frac{1}{26.2}\left(\frac{k}{d_i}\right)\frac{Pr}{(Pr^{2/3} - 0.074)}Re^{4/5}\left(\frac{d_i}{D_c}\right)^{0.1}[1 + \frac{0.098}{\{Re(d_i/D_c)^2\}^{0.2}}] \qquad (2\text{-}133)$$

The following constant multipliers are applied to the helical steam generator heat transfer correlations:

Region	Default Value	SC
Subcooled Convection	1.0	C4186(1)
Two-Phase Convection	1.0	C4186(2)
Superheated Convection	1.0	C4186(3)

2.7 Mass Transfer

Condensation occurs on a structure surface if its temperature is below the dew point of the associated atmosphere, and mass transfer from that surface has been enabled through user input (see the description for record HS_LB in the HS Users' Guide). The dew point is the saturation temperature corresponding to the partial pressure of steam in the bulk atmosphere of the boundary volume (obtained from the CVH database). Evaporation from an existing film on a heat structure surface occurs if the surface temperature of the film exceeds the dew point. (A model to treat film flashing at the structure/film interface, when the temperature exceeds the boiling temperature, has not been activated because it has been unnecessary.)

In nearly pure steam environments, the rate of condensation is limited only by heat transfer through the structure, i.e. by the ability of the structure to dissipate the latent heat of vaporization that is released by condensation. Hence, in nearly pure steam environments, the rates of condensation and evaporation self-adjust to whatever values are required to maintain the saturation temperature at the film/atmosphere interface.

As noncondensibles are introduced into the condensing steam, their accumulation near the film surface from local steam depletion tends to inhibit the flow of fresh steam to the film surface and restricts the rate of condensation. Consequently, when the ratio of the steam partial pressure to the total pressure in the boundary volume (obtained from the CVH database) falls below a user-prescribed threshold, VPFRAC (also sensitivity coefficient 4200 with a default value of 0.9995), a mass transfer rate limitation is imposed on the rate of condensation. Experimental evidence indicates that the value of VPFRAC (below which diffusion rate limitations to condensation mass transfer become significant) can depend on the degree of turbulence. As the turbulence decreases, the value of VPFRAC should be increased to account for the inhibiting effect of even very small amounts of noncondensibles in a stagnant environment. Conversely, in a well-mixed system, the value of VPFRAC may have to be reduced to avoid artificially limiting the

condensation rate. It is suggested that the user vary the value of VPFRAC in sensitivity studies, if uncertainty in the rate of condensation is of much concern.

The mass transfer rate limitation is a function of the diffusion mass transfer coefficient, which is calculated at a heat structure boundary surface whenever the surface is exposed to the atmosphere of its boundary volume. This coefficient is related to the atmosphere Nusselt number through a heat transfer analogy and is calculated by a Sherwood number correlation involving the Nusselt, Prandtl, and Schmidt numbers. This correlation is presented in Section 2.7.1.

The mass transfer rate-limited expression for condensation or evaporation at a surface exposed to a noncondensible-bearing atmosphere is formulated using a mechanistic approach that models the diffusion of a condensable vapor through a gas layer that contains noncondensible gases. Section 2.7.2 discusses this expression. If the surface temperature is greater than the critical temperature, 647.2 K, diffusion mass transfer is not calculated. However, the diffusion mass transfer coefficient is still calculated using Equation (2-139) since the radionuclide package requires this quantity.

2.7.1 Sherwood Number for Diffusion Mass Transfer

The mass transfer coefficient is related to the atmosphere Nusselt number by a heat and mass transfer analogy. In addition to the use of the Nusselt, Reynolds, and Prandtl numbers, the HS package uses the following dimensionless variables for its mass transfer calculations:

Schmidt number (Sc) = $\mu/(\rho D)$

Sherwood number (Sh) = $h_D L_c/D$

where

μ = dynamic viscosity of atmosphere at average of surface and atmosphere temperatures, kg/m•s

ρ = density of atmosphere, kg/m^3

D = diffusivity, m^2/s

h_D = mass transfer coefficient, m/s

L_c = characteristic length or dimension of surface, m

A Sherwood number correlation is used to calculate a diffusion mass transfer coefficient. The correlation is

$$Sh = C\,Nu^{a}\,Sc^{b}\,\mathrm{Pr}^{d} \tag{2-134}$$

where

Nu = Nusselt number

Pr = Prandtl number

The constants *C, a, b*, and *d* have been implemented as sensitivity coefficient array C4201. The default values are:

C = 1.0

a = 1.0

b = 1/3

d = -1/3

The mass transfer coefficient is then obtained by

$$h_D = F_m Sh\ D/L_c \tag{2-135}$$

where F_m is an arbitrary, nonnegative scaling factor (with a default value of 1.0) that may be specified by the user at any surface. Refer to Section 2.6 for further discussion of this scaling factor and a caution concerning its use.

2.7.2 Condensation and Evaporation with Noncondensibles

The principal expression for condensation or evaporation mass flux at a surface exposed to an atmosphere with a significant partial pressure of noncondensible gases (i.e., $P_{stm} < VPFRACxP_{tot}$) is formulated using a mechanistic approach that models the diffusion of a condensable vapor through a gas layer that contains noncondensible gases [5]. The condensation mass flux is given by:

$$\dot{m}_c = h_D \rho_v \, ln\left(\Delta P_{srf}/\Delta P_{atm}\right) \tag{2-136}$$

where

$\dot{m}_c$ = mass flux at this surface, kg/m^2•s

h_D = mass transfer coefficient, m/s

ρ_v = density of vapor at $T_{sat}(P_{tot})$, kg/m^3

ΔP_{srf} = P_{tot} - P_{srf}, Pa

ΔP_{atm} = P_{tot} - P_{stm}, Pa

P_{tot} = total control volume pressure, Pa

P_{srf} = saturation pressure of steam at the surface temperature, Pa

P_{stm} = steam partial pressure in the control volume, Pa

Because Equation (2-136) is singular when the P_{srf} reaches P_{tot} it is necessary to bound the rate of evaporation as the surface temperature reaches T_{sat} (P_{tot}). This is done by using a flashing heat transfer coefficient to limit the rate of evaporation as follows:

$$\dot{m}_e = h_e \, min\left(0, T_{dew} - T_{srf}\right)/h_{fg} \tag{2-137}$$

$$\dot{m} = max(\dot{m}_c, \dot{m}_e) \tag{2-138}$$

where

h_e = flashing heat transfer coefficient, W/m^2•K

h_{fg} = latent heat of vaporization for steam, J/kg

T_{dew} = control volume dew point temperature, K

T_{srf} = surface temperature, K

and h_e has been implemented as sensitivity coefficient C4202, with a default value of 5 x 10^5 W/m^2•K.

2.7.3 Mass-Energy Transfer to Control Volumes

The mass that is transferred between the surface of a heat structure and the atmosphere of its boundary volume between times $t_{m\text{-}1}$ and t_m is the value of the integral from $t_{m\text{-}1}$ to t_m of the product of the mass flux and the area of the surface that is exposed to the atmosphere:

$$\Delta m^m = \dot{m} \, A\left(1 - x_{pool}\right)\Delta t_m \tag{2-139}$$

where

Δm^m = mass transferred between heat structure surface and atmosphere between times $t_{m\text{-}1}$ and t_m, kg

$\dot{m}$ = mass flux at heat structure surface, kg/m^2•s

A = heat structure boundary surface area, m^2

x_{pool} = fraction of boundary surface in pool of boundary volume

Δt_m = system timestep size ($t_m - t_{m-1}$), s

Mass transfer is not considered if the pool fraction is greater than the critical pool fraction CPFAL for the structure as defined in Section 2.4.

The time-surface integral of the mass flux is evaluated during each computational cycle for each surface to determine the total mass of the liquid on each heat structure boundary surface. Its value is constrained so that no more steam is condensed than is present in its boundary volume and no more liquid is evaporated or flashed than is present on the surface.

If more than a user-specified fraction (sensitivity coefficient C4203(2), with a default value of 90%) of the steam in a control volume is condensed during a computational cycle, then remedial action is taken. If the current timestep size is greater than a user-specified value (sensitivity coefficient C4203(1), with a default value of -1. s), then the HS package requests that the computational cycle be repeated with a smaller timestep size to eliminate the excessive condensation. The requested timestep size is equal to the current value times the ratio of the maximum amount of steam that may condense divided by the actual, excessive amount that would have condensed without the requested fallback. If the current timestep size is less than the value prescribed by sensitivity coefficient C4203(1), then the condensation flux (mass transfer rate) on each surface associated with this boundary volume is reduced by the same factor that would have been applied to the timestep size with the fallback option. The HS (not the entire MELCOR cycle) calculation is then repeated with the modified mass transfer rates. The fallback option is the default and recommended option because it does not alter the mechanistically calculated condensation rates. The scaling option may falsify the solution and should be avoided, if possible. Excessive condensation is a result of violating a timestep size constraint imposed by the explicit coupling between the HS and CVH packages (and is akin to the material Courant timestep limit). In some situations, it may be possible to avoid excessive condensation by re-nodalizing the problem to reduce the ratio of the surface area for condensation to the volume of steam available for condensation.

The liquid mass that is transferred to a heat structure surface by other packages is obtained from an array in the HS package database whose elements are updated using an interface routine that can be called by any package.

For each heat structure surface, the mass and energy transfer are calculated for the steam that was condensed from or added to its boundary volume atmosphere and the liquid deposited in the boundary volume pool. The results of these calculations are used to update the mass and energy communication arrays for the CVH package. If the heat structure surface is part of a film tracking network, then the film thickness is determined dynamically as a function of the film flow rate and the drainage from the surface is partitioned between the boundary volume pool, the boundary volume fog and the surfaces

of other heat structures in the user-specified network. The film tracking model is discussed in Section 2.8.2.

2.8 Liquid Film Modeling

The mass, thickness, and specific enthalpy of a liquid film on a heat structure boundary surface are first determined during the initialization procedure in MELGEN execution. Calculation of these properties is also carried out during MELCOR execution. The models that are used to determine liquid film properties are described in Section 2.8.1. Section 2.8.2 describes the film tracking model, which is based on the model in CONTAIN [4] and is user-activated to track film drainage over a user-specified network of connected structure surfaces.

2.8.1 Film Models

During MELGEN and MELCOR execution, the mass of a liquid film on a heat structure boundary surface is determined from

(1) calculation of the mass that is transferred between this surface and its boundary volume by condensation, evaporation or draining,

(2) the liquid mass that is transferred to this surface by other packages, and

(3) the liquid mass that is transferred to this surface by external sources (tabular function or control function) or film drainage from other heat structure surfaces, if the surface is part of a user-defined film tracking network.

The mass of the liquid film and the film surface and structure surface temperatures enable its thickness and specific enthalpy to be determined. The film equations are nodalized so that half of the film mass is associated with the film/structure interfacial node and the other half is associated with the atmosphere/film interfacial node. Therefore, the average specific enthalpy of the film is given by $0.5 \cdot [h_f (T_{s,srf}) + h_f (T_{f,srf})]$, where $h_f (T)$ is the specific enthalpy of the film at temperature T, $T_{s,srf}$ is the film/structure interfacial temperature and $T_{f,srf}$ is the atmosphere/film interfacial temperature.

For structures that are not part of a film tracking network, the condensate and deposited liquid is permitted to accumulate on a surface until the film thickness reaches a maximum. If the liquid mass is sufficiently large that the film thickness exceeds this maximum, then the excess liquid is deposited in the pool of the boundary volume of the surface. The maximum thickness of a liquid film on a surface is determined in one of two ways:

(1) for geometries for which the convective heat transfer coefficient through the film (see Section 2.6.1.1) is obtained from a correlation as a function of the Reynolds number of the film flow, the Reynolds number is also used to obtain the film thickness from the correlations used by the film tracking model (see Section 2.8.2 below) or

(2) for all other geometries the maximum film thickness is obtained from a user-adjustable value. The user-adjustable value, $\delta_{\max}$, has been implemented as sensitivity coefficient C4251(2) with a default value of 5 x 10^{-4} m.

2.8.2 Film Tracking Model

For structures that are part of a film tracking network, the film thickness on a surface is determined iteratively as a function of the Reynolds number of the film flow rate as follows. First, the Reynolds number of the film flow is given by

$$Re_f = 2\,(\dot{m}_{in} + \dot{m}_{out})/(w\ \mu_f) \tag{2-140}$$

where $\dot{m}_{in}$ is the mass inflow rate (kg/s) from film drainage to the surface from other surfaces in the network and water deposited on the surface by other MELCOR packages, $\dot{m}_{out}$ is the mass outflow rate (kg/s) from film drainage from this surface (which is to be determined iteratively), w is the width of this surface and μ_f is the bulk viscosity of the film. As an initial guess $\dot{m}_{out}$ is set equal to zero. The film thickness as a function of Re_f is given by the following correlation

$$\begin{aligned}
\delta_f &= C_{f,l} \cdot \delta^* \cdot \mathrm{Re}_f^{ef,l}, \text{ if } \mathrm{Re}_f < \mathrm{Re}_{LAM} \\
&= C_{f,t} \cdot \delta^* \cdot \mathrm{Re}_f^{ef,t}, \text{ if } \mathrm{Re}_f > \mathrm{Re}_{TURB} \\
&= \text{determined by interpolation between limits at } \mathrm{Re}_{LAM} \text{ and } \mathrm{Re}_{TURB}, \text{otherwise} \\
\delta^* &= \left[(\mu_f/\rho_f)^2/(g \cdot \sin\theta)\right]^{1/3}
\end{aligned} \tag{2-141}$$

where the constants $C_{f,x}$, exponents ef,x and limits Re_x (where x represents laminar or turbulent) in Equation (2-141) have been implemented as sensitivity coefficients C4253, and ρ_f and θ are the film density and angle of inclination of the surface from horizontal, respectively. The film thickness can also be determined from the conservation of film mass as

$$\delta_f = \left[m_{f,0} + (\dot{m}_{in} + \dot{m}_c - \dot{m}_{out}) \cdot \Delta t\right]/(\rho_f A_{srf}) \tag{2-142}$$

where $m_{f,0}$ is the film mass at the start of the timestep Δt, $\dot{m}_c$ is the condensation rate (a negative value indicates evaporation) and A_{srf} is the surface area. Equation (2-142) has been presented for the case of rectangular geometry; the equations for cylindrical and spherical geometry are different because the film thickness is related to film volume differently.

For given values of $m_{f,0}$, $\dot{m}_{in}$, and $\dot{m}_c$, Equations (2-141) and (2-142) can be solved simultaneously by iterating on the value of $\dot{m}_{out}$ to determine consistent values of δ_f and $\dot{m}_{out}$. Note, however, that if the value of δ_f given by Equation (2-141) with $\dot{m}_{out}$ = 0 exceeds the value of δ_f given by Equation (2-142) with $\dot{m}_{out}$ = 0, then the film thickness cannot possibly achieve the steady-state value consistent with Equation (2-141) during the given timestep. When a steady-state value consistent with Equation (2-141) is impossible for a timestep, $\dot{m}_{out}$ is set equal to zero and Equation (2-142) is used to determine δ_f.

A user-specified minimum film thickness, $\delta_{\min}$, has been implemented as sensitivity coefficient C4251(1) to prevent film flow when the film thickness is less than the specified value (default value is 10^{-9} m). This can be used to inhibit film flow on rough surfaces until a reasonably thick film is established. Hence, when the solution to Equations (2-141) and (2-142) is less than $\delta_{\min}$, then δ_f in Equation (2-142) is set equal to $\delta_{\min}$ to determine the value of $\dot{m}_{out}$, if a positive value is possible; otherwise, δ_f is equal to the value obtained from Equation (2-142) with $\dot{m}_{out}$ set equal to zero.

The outflow (drainage) from the film tracking solution, $\dot{m}_{out}$, is partitioned between the CVH pool associated with the surface, “rain” passed to the SPR package via the TP package and the other drainage surfaces associated with the given surface through the user-specified film tracking network.

2.9 Stored Energy of a Heat Structure

The total stored energy of each heat structure, including surface films, is initialized during MELGEN execution. The stored energy of the structure itself is obtained by integrating the product of the volumetric heat capacity weight and the absolute temperature over the volume of the heat structure. The energy of the films is added to that total to arrive at a total structure energy storage. Therefore, the initial stored energy is

$$E^0 = f \sum_{i=1}^{N} G_i^m T_i^0 + m_{film,L}^0 \, h_{film,L}^0 + m_{film,R}^0 h_{film,R}^0 \qquad (2\text{-}143)$$

where

E^0 = initial stored energy of heat structure, J

f = geometry factor with the following values for different geometries,

= surface area of heat structure for rectangular geometries, m^2

= axial length of heat structure for cylindrical geometries, m

= 1.0 for spherical and hemispherical geometries.

G_i^0 = volumetric heat capacity weight

= $C_{p,i}^0$ HVR for $i = 1$

= $C_{p,i-1}^0 HVL_i + C_{p,j}^0 HVR_i$ for $i = 2, \ldots, N-1$

= $C_{p,i-1}^0 HVL_i$ for $i = N$

$C_{p,i}^0$ = initial volumetric heat capacity of mesh interval i, $J/m^3 \bullet K$

HVL_i = left (inside) volume weight for mesh interval i, defined in Sections 2.1.1 and 2.2.1

HVR_i = right (outside) volume weight for mesh interval i, defined in Sections 2.1.1 and 2.2.1

T_i^0 = initial temperature of node i, K

T_{i-1}^0 = initial temperature of node $i-1$, K

$m_{film,L}^0$ = initial mass of film on left boundary surface, kg

$h_{film,L}^0$ = initial specific enthalpy of film on left boundary surface, J/kg

$m_{film,R}^0$ = initial mass of film on right boundary surface, kg

$h_{film,R}^0$ = initial specific enthalpy of film on right boundary surface, J/kg

During MELCOR execution, the change in the stored energy of each heat structure is calculated every cycle. This is obtained by integrating the product of the volumetric heat capacity weight and the change in temperature between times t_{m-1} and t_m over the volume of the heat structure, and including the energy change of the surface films. Therefore,

$$\Delta E^m = f \sum_{i=1}^{N} G_i^m \left(T_i^m - T_i^{m-1} \right) + m_{film,L}^m h_{film,L}^m - m_{film,L}^{m-1} h_{film,L}^{m-1} + m_{film,R}^m h_{film,R}^m - m_{film,R}^{m-1} h_{film,R}^{m-1} \quad (2\text{-}144)$$

where

ΔE^m = change in stored energy of heat structure between times t_{m-1} and $t_{m,j}$

G_i^m = volumetric heat capacity weight

= $C_{p,i}^{m} HVR_i$ for $i = 1$

= $C_{p,i-1}^{m} HVL_i + C_{p,i}^{m} HVR_i$ for $i = 2, \ldots, N-1$

= $C_{p,i-1}^{m} HVL_i$ for $i = N$

$C_{p,i}^{m}$ = volumetric heat capacity of mesh interval i at time t_m, $J/m^3 \bullet K$

T_i^m = temperature of node i at time t_m, K

T_1^{m-1} = temperature of node i at time $t_{m\text{-}1}$, K

$m_{film,L}$ = mass of film on left boundary surface, kg

$h_{film,L}$ = specific enthalpy of film on left boundary surface, kg

$m_{film,R}$ = mass of film on right boundary surface, kg

$h_{film,R}$ = specific enthalpy of film on right boundary surface, kg

m = denotes quantity of time t_m

$^{m+1}$ = denotes quantity at time $t_m + \Delta t_m$

2.10 Degassing Model

The HS package degassing model assumes that the gas release occurs uniformly over the degassing temperature range. The contribution to the degassing rate for each mesh interval whose temperature exceeds the previously attained maximum is the product of the source density, volume of the mesh interval, and the fraction of the degassing temperature range that the present maximum represents. Therefore,

$$g_k^m = \sum f \, \rho_{gas} (HVL + HVR) \frac{(T_2 - T_1)}{(T_{DG\max} - T_{DG\min})} \Big/ \Delta t_m \qquad (2\text{-}145)$$

where

g_k^m = degassing rate for k-th source at time t_m, kg/s

$\sum$ = sum over all heat structure nodes containing gas sources

f = geometry factor

= surface area of heat structure for rectangular geometries, m^2

= axial length of heat structure for cylindrical geometries, m

= 1.0 for spherical or hemispherical geometries

ρ_{gas} = source density, kg/m^3

HVL = volume weight for left surface (Table 2.1 and Table 2.2)

HVR = volume weight for right surface (Table 2.1 and Table 2.2)

T_2 = min ($T^m_{n\max}$, T_{DGmax}), K

T_1 = min (T^{m-1}_{nmax}, T_{DGmax}), K

T^m_{nmax} = maximum temperature in mesh interval n at time t_m, K

T^{m-1}_{nmax} = maximum temperature in mesh interval n at time t_{m-1}, K

T_{DGmax} = upper temperature in degassing temperature range, K

T_{DGmin} = lower temperature in degassing temperature range, K

Δt_m = system timestep size ($t_m - t_{m-1}$), s

The HS package calculates the mass and internal energy of the gas (at the boundary volume temperature) that is released by each source through the present computational cycle. These data are then used to update the mass and energy transfer communication arrays for the CVH package.

2.11 Ice Condenser Model

The ice condenser model allows the description of certain features found in Westinghouse PWR ice condenser containments. This model is a specially modified application of the heat structure degassing model described in Section 2.10. The user activates the ice condenser logic by including a prescribed keyword in the input for multiple solid, vertical cylindrical structures. A special “gas” source is defined to release liquid water into the pool of the outer associated CVH volume. The “degassing” temperature range should have a lower temperature of 274 K (just above the melting temperature of ice) to avoid problems associated with limits of the thermodynamic and material properties routines. The upper temperature of the “degassing” range is a modeling choice typically assigned a value approximately ten degrees higher. The heat of reaction of the gas source should include sensible heating of the ice from its actual subcooled temperature to the melting point in addition to the latent heat of fusion. A special ice condenser Nusselt number multiplier has been added to the gas source input to account for effects not explicitly modeled that may affect the rate of heat transfer to the ice cylinder. Similarly, an ice condenser radionuclide deposition surface area enhancement factor has been added to account for unmodeled effects that enhance the rate of radionuclide deposition on the ice condenser. Finally, a parameter has been added that can be adjusted by user input to vary the rate of decrease of the ice surface area as the ice melts. The ice surface area varies as

$$(V/V_o)^{EXPICE}$$

where

V = current ice volume

V_o = initial ice volume

EXPICE = user-specified exponent

The total surface area for heat transfer to the ice condenser (ice and baskets) is the initial surface area of the cylindrical ice columns, A_o, multiplied by the factor

$$RNDICE + (1 - RNDICE) \cdot (V/V_o)^{EXPICE}$$

to provide a smooth transition to the minimum surface area of the ice baskets, *RNDICE* x A_o, where *RNDICE* is user defined. The total surface area for RN deposition is equal to the ice surface area plus the surface area of the ice baskets. The "gas" source density is that of liquid water.

MELCOR automatically accounts for the volume change associated with the reduction in ice mass as melting proceeds. The user should define tabular input to specify properties for the metal baskets that hold the granular ice. The appropriate density is the value of the metal mass divided by the total volume occupied by the baskets. The thermal conductivity should exceed the value associated with ice to account for steam penetration into the granular matrix and conduction in the metal. The specified heat capacity is that of the metal.

2.12 Steel Melt Model

To allow for core boundary structure heating and subsequent melting, an option has been included in the degassing model to allow the user to input stainless steel as a degassing source. This option has been included only for use when the COR package is being used and is ignored otherwise. The implementation of the stainless steel degassing source is very similar to that used for ice as part of the ice condenser model (described above) except the heat of reaction of the stainless steel gas source should only include the latent heat of fusion. Because stainless steel is not a hydrodynamic material included in the CVH package, the volume of the melting stainless steel is associated with the COR package materials. As such, these materials are represented by the CVH package as "virtual volume" and, as with the ice condenser model, the volume changes due to melting are explicitly represented.

To prevent potential problems of adding a large amount of heat to a stainless steel degassing structure with an insignificant residual mass, a sensitivity coefficient, C4205,

has been provided as a lower limit such that if the remaining unmelted structure mass falls below this limit, then the structure is assumed to have completely melted. When the structure is calculated to have completely melted, it is deactivated and HS processing for this structure is discontinued.

2.13 Communication with Other Packages

After completing the calculations discussed in Sections 2.1 through 2.12, the HS package communicates various changes to other packages using well-defined communication interfaces. The HS package communicates to the CVH package any mass, energy, and virtual volume changes in each control volume due to the following mechanisms:

(1) heat transfer between each heat structure and the pool and atmosphere of its boundary volumes

(2) condensation of steam onto each heat structure from the atmosphere of its boundary volumes

(3) evaporation or flashing of liquid (water) from each heat structure boundary surface into the atmosphere of its boundary volumes

(4) deposition of liquid (water) from each heat structure into the pool of its boundary volumes

(5) degassing of materials within each heat structure

The virtual volume with which the HS package is concerned is the volume occupied by all water films, ice condenser ice and meltable (degassible) steel associated with the core boundary structure melting model. Initial values of virtual volume are calculated during MELGEN execution and changes in virtual volume are calculated each computational cycle during MELCOR execution.

Prior to communication of control volume mass and energy changes to the CVH package, the HS package determines if these changes lead to a negative mass of some material. If a negative mass is detected, the HS package requests that the present computational cycle be repeated with a timestep reduction and the changes not be communicated to the CVH package.

During MELCOR execution, the HS package calculates and communicates to the RadioNuclide (RN) package the fraction of liquid (water) mass on each heat structure boundary surface deposited during each computational cycle in the pool of its boundary volume. These fractions are used to calculate the relocation of radionuclides from deposition surfaces to the pools of their boundary volumes.

3. Solution Methods

The finite-difference approximation to the heat conduction equation with boundary conditions utilized by the HS package results in a tridiagonal system of N equations (N+1 or N+2 if there is a liquid film on one or both surfaces) for a heat structure with N (N+1 or N+2) temperature nodes. In order to reduce roundoff problems, the temperature of the heat structure relative to the minimum value of that heat structure is used to set up and solve the equations. The solution procedure is usually more complex than the standard solution for a tridiagonal system of equations. The boundary conditions often include energy input due to mass transfer and may include the deposition of energy from other sources such as decay-heat of radionuclides. Furthermore, the temperature nodes near the surface of a heat structure may be too closely spaced to accurately calculate the temperature at the surface, or the computational timestep may be large. This section discusses some of the special solution procedures that are used to obtain the steady-state and transient temperature distribution of a heat structure.

3.1 Iteration Strategy

By default an iterative procedure is employed to determine the temperature profile in each heat structure. This procedure repeats the following calculations until either convergence is attained or a maximum number of iterations is performed:

(1) thermal properties

(2) heat transfer coefficients

(3) mass transfer

(4) boundary condition coefficients

(5) temperature distribution

Convergence is determined by testing the relative error in several dependent variables calculated during the temperature iteration:

(1) the temperature at each node in the structure (including the film interfacial temperature[s]),

(2) the mass of the film(s) (if the film thickness exceeds 10^{-5} m), and

(3) the boiling heat transfer coefficient(s).

The relative error for dependent variable X is defined as

$$ERR_X = (X^m / X^{m-1}) - 1 \qquad (3\text{-}1)$$

where

$X^{m\text{-}1}$ = value of X at iteration m-1

X^m = value of X at iteration m

If the relative error in the temperature profile falls below a threshold value (ERR_{prp} in Section 3.3 below) during an iteration, then material properties are generally not recalculated for that iteration step. Values from the previous iteration step are used until the relative error again becomes higher than ERR_{prp} or until convergence is achieved. However, during degassing, properties must be updated after every iteration to ensure sufficient accuracy of the degassing rate.

Relaxation of the pool boiling heat transfer coefficient may be required in some situations, since it is extremely sensitive to changes in the surface temperature. Because relaxation effectively falsifies the value of the heat transfer coefficient until the relaxed value stabilizes, it is necessary to check the relative difference (error) between the unrelaxed value (which is determined by the latest surface temperature iterate) and the relaxed value. When pool boiling occurs, the pool heat transfer coefficient h_{pool}^m is relaxed between temperature iterations to be:

$$W_B h_{pool}^{m-1} + (1 - W_B)\ h_{pool}^m \qquad (3\text{-}2)$$

where

$h_{pool}^{m\text{-}1}$ = pool heat transfer coefficient at iteration m-1, W/m^2•K

h_{pool}^m = pool heat transfer coefficient at iteration m, W/m^2•K

and W_B is the modified relaxation parameter, which is set to an initial value that depends on whether it is a steady-state or transient iteration (RLX_B in Sections 3.2 and 3.3 below), and which is then decreased by a factor of 0.95 for each iteration during which the relative error in the boiling surface temperature is greater than -0.5 (i.e., the boiling surface temperature is not oscillating excessively so that relaxation may be reduced). Note that corrections have been made to this equation relative to the MELCOR 1.8.5 HS Reference Manual.

The system of equations describing the transient temperatures within a structure can become ill-conditioned if the timestep becomes too large. A precision limit (P_{illmat} in Section 3.3) is imposed in the routine that performs the direct inversion of the tridiagonal coefficient matrix. If the relative difference between terms used in evaluating a nonzero difference in the algorithm is less than this limit, then there are too few significant figures

in the difference to achieve the requested degree of precision. During MELGEN execution this condition is fatal but may be corrected by reducing the value of Δt_o, discussed in Section 3.2 below. During MELCOR execution the cycle is repeated with a smaller timestep to alleviate the problem.

There are three levels of convergence criteria used: the desired convergence criteria, a more stringent override convergence criterion, and the less stringent acceptance convergence criteria. The desired and acceptance criteria may be assigned different values for MELGEN and MELCOR execution, as discussed in Sections 3.2 and 3.3, respectively.

The desired convergence criteria are normally stringent enough to ensure reasonable accuracy in the overall results in the absence of phenomena that demand very high temperature resolution. During the occurrence of phenomena such as degassing or mass transfer (condensation/evaporation), however, very small errors in temperature can cause quite large errors in degassing and mass transfer rates. Therefore, during the occurrence of degassing or mass transfer, the desired temperature convergence criteria are overridden by the override temperature convergence criterion (as long as it is more stringent). The override temperature convergence criterion is contained in sensitivity coefficient array 4055 and discussed in Section 3.3 below.

The iteration procedure for a heat structure continues until either the desired criteria (or the override criterion during degassing or mass transfer) have been met or the iteration count reaches a prescribed maximum. During override, convergence is declared after the maximum number of iterations if the desired criteria have been met, even though the more stringent criterion has not been satisfied. If the acceptance criteria have not been satisfied for all tested variables on a heat structure after the maximum number of permitted iterations is performed, failure is declared for that heat structure. If the acceptance criteria have been met but the desired criteria have not been met, then success is declared but a message is issued to warn the user that the desired criteria were not met.

3.2 Steady-State Convergence Criteria

During MELGEN execution, an initial temperature distribution is calculated for a given heat structure if specified by user input. The following constants are the iteration parameters used for steady-state heat conduction calculations. They are implemented as sensitivity coefficient array C4051.

ITR_{ss} = maximum number of permitted steady-state iterations (default = 400)

ERR_{ss} = desired relative error tolerance for temperatures during steady-state calculations (default = 10^{-5})

Δt_o = initial steady-state timestep (default = 10^5 s)

ERF_{ss} = desired relative error tolerance for film mass for steady-state calculations (default = 10^{-2})

DIE_{ss} = acceptable relative error tolerance for temperature during steady-state calculations (default = 10^{-2})

There is no acceptance criterion for film mass; the iteration procedure continues for ITR_{ss} iterations to satisfy the desired criterion, ERF_{ss}, but after ITR_{ss} iterations the film mass value is declared acceptable no matter what. The following coefficients are the iteration relaxation parameters used for steady-state heat conduction calculations to mitigate temperature oscillations. They are implemented as sensitivity coefficient array C4052.

RLX_B = steady-state boiling heat transfer coefficient relaxation parameter (default = 0.0)

ERR_B = desired steady-state boiling heat transfer coefficient error tolerance (default = 0.05)

The boiling heat transfer coefficient relative error acceptance criterion is 100%, that is it may double or vanish.

If any heat structure fails to meet the acceptance criteria, a restart file is not written and MELCOR execution may not begin.

3.3 Transient Convergence Criteria

During MELCOR execution, an iterative procedure is invoked if specified by user input (highly recommended). The following constants are the iteration parameters used for transient heat conduction calculations. They are implemented as sensitivity coefficient array C4055.

ITR_{trn} = maximum number of transient iterations (default = 30.)

ERR_{trn} = desired relative error tolerance for temperature during transient conduction calculations (default = 0.0005)

ITR_{cut} = minimum number of transient iterations required to prevent increase of the timestep size (default = 31.)

ERR_{prp} = minimum relative error tolerance for material property determination (default = 0.01)

P_{illmat} = tridiagonal matrix solver precision requirement (default = 1.0×10^{-10})

ERR_{ovr} = error tolerance override during degassing/mass transfer (default = 5.0×10^{-6})

ERF_{trn} = maximum relative error tolerance for film mass during transient conduction calculations (default = 0.01)

DIE_{trn} = maximum relative error tolerance for transient temperature during conduction calculations (default = 0.005)

There is no acceptance criterion for film mass; the iteration procedure continues for ITR_{trn} to satisfy the desired criterion, ERF_{trn}, but after ITR_{trn} iterations the film mass value is declared acceptable no matter what. The boiling heat transfer coefficient relative error acceptance criterion is 100%, that is, it may double or vanish.

Although inactive by default, the value of ITR_{cut} can be adjusted downward (below ITR_{trn}) to prevent the MELCOR system timestep from increasing if the HS package is taking too many iterations per timestep. Judicious use of this feature requires comparing total CPU usage for various strategies.

The following constants are the relaxation parameters used for transient calculations. These parameters are implemented as sensitivity coefficient array C4056:

RLX_B = transient boiling heat transfer coefficient relaxation parameters (default = 0.9)

ERR_T = transient boiling heat transfer coefficient error tolerance (default = 0.05)

If the temperature solution fails for any structure during a calculation cycle, the HS package immediately requests that the cycle be repeated with the timestep reduced by a factor of one half. Failure may occur for several reasons, including excessive error in the temperatures, excessive error in the boiling heat transfer coefficient, numerical problems associated with finite precision or the generation of an out-of-range temperature (less than 273 K or greater than 4990 K), either legitimately or from divergence of the iterative algorithm.

4. Timestep Control

Timestep control is exercised by the Heat Structure package in cases (a) and (b) below by the HS package requesting that the current timestep be repeated with a smaller timestep size to correct the problem.

(a) condensation is causing excessive steam depletion in the CVH package as discussed in Section 2.7.3 or,

(b) the temperature solution for a heat structure fails to converge within the prescribed maximum number of iterations as discussed in Section 3.3. (a timestep reduction request to one-half the current timestep is also made if a physically unreasonable value is detected during the iterative solution).

In these cases, the HS package requests that the current timestep be repeated with a smaller timestep size to attempt to correct the problem.

Appendix A: Sensitivity Coefficients

This appendix provides the sensitivity coefficients associated with various correlations and modeling parameters used in the HS package and described in this reference manual.

Equation or Section	Coefficient	Value	Units
(2-115)	C4000(1)	0.2358	N/m
	C4000(2)	1.0	
	C4000(3)	-0.625	K^{-1}
	C4000(4)	1.256	
	C4000(5)	0.	N/m
	C4000(6)	1.0	
	C4000(7)	647.3	K
Section 3.2	C4051(1)	400.	
	C4051(2)	10^{-5}	
	C4051(3)	10^{5}	s
	C4051(4)	0.01	
	C4051(5)	0.01	
Section 3.2	C4052(1)	0.0	
	C4052(2)	0.05	
Section 3.3	C4055(1)	30.	
	C4055(2)	0.0005	
	C4055(3)	31.	
	C4055(4)	0.01	
	C4055(5)	10^{-10}	
	C4055(6)	$5.x10^{-6}$	
	C4055(7)	0.01	
	C4055(8)	0.005	
Section 3.3	C4056(1)	0.9	
	C4056(2)	0.05	
(2-79) - (2-81)	C4060(1)	1.0	
	C4060(2)	10.0	
(2-85) - (2-87)	C4061(1)	10^{9}	
	C4061(2)	10^{10}	
	C4062(1)	10^{9}	
	C4062(2)	10^{10}	
	C4063(1)	10^{9}	
	C4063(2)	10^{10}	
(2-88a)–(2-90b)	C4064(1)	$3.0x10^{5}$	
	C4064(2)	$6.0x10^{5}$	
	C4065(1)	$2.0x10^{3}$	

Equation or Section	Coefficient	Value	Units
	C4065(2)	10^4	
	C4066(1)	$2.0x10^3$	
	C4066(2)	10^4	
(2-4)	C4071(1)	0.02	
	C4071(2)	0.98	
(2-103) - (2-105)	C4080(1)	1.0	
	C4080(2)	10.0	
(2-108) - (2-110)	C4081(1)	10^9	
	C4081(2)	10^{10}	
	C4082(1)	10^9	
	C4082(2)	10^{10}	
	C4083(1)	10^9	
	C4083(2)	10^{10}	
(2-105) – (2-107)	C4084(1)	$3.0x10^5$	
	C4084(2)	$6.0x10^5$	
	C4085(1)	$2.0x10^3$	
	C4085(2)	10^4	
	C4086(1)	$2.0x10^3$	
	C4086(2)	10^4	
(2-82)	C4101(1)	0.046	
	C4101(2)	1/3	
	C4101(3)	0.	
(2-82)	C4102(1)	0.046	
	C4102(2)	1/3	
	C4102(3)	0.	
(2-82)	C4103(1)	0.228	
	C4103(2)	0.226	
	C4103(3)	0.	
(2-82)	C4104(1)	0.046	
	C4104(2)	1/3	
	C4104(3)	0.	
(2-82)	C4105(1)	0.046	
	C4105(2)	1/3	
	C4105(3)	0.	
(2-82)	C4106(1)	0.228	
	C4106(2)	0.226	
	C4106(3)	0.	
(2-82)	C4107(1)	0.59	
	C4107(2)	0.25	
	C4107(3)	0.	

Equation or Section	Coefficient	Value	Units
(2-82)	C4108(1)	0.59	
	C4108(2)	0.25	
	C4108(3)	0.	
(2-82)	C4109(1)	0.43	
	C4109(2)	0.25	
	C4109(3)	12.0	
(2-82)	C4110(1)	0.10	
	C4110(2)	1/3	
	C4110(3)	0.	
(2-82)	C4111(1)	0.10	
	C4111(2)	1/3	
	C4111(3)	0.	
(2-82)	C4112(1)	0.43	
	C4112(2)	0.25	
	C4112(3)	2.0	
(2-83)	C4113(1)	8.235	
	C4113(2)	0.	
	C4113(3)	0.	
	C4113(4)	0.	
(2-83)	C4114(1)	48/11	
	C4114(2)	0.	
	C4114(3)	0.	
	C4114(4)	0.	
(2-83)	C4115(1)	48/11	
	C4115(2)	0	
	C4115(3)	0.	
	C4115(4)	0.	
(2-83)	C4116(1)	0.023	
	C4116(2)	0.8	
	C4116(3)	1/3	
	C4116(4)	0.	
(2-83)	C4117(1)	0.023	
	C4117(2)	0.8	
	C4117(3)	1/3	
	C4117(4)	0.	
(2-83)	C4118(1)	0.023	
	C4118(2)	0.8	
	C4118(3)	1/3	
	C4118(4)	0.	
	C4119(1)	0.664	

Equation or Section	Coefficient	Value	Units
(2-83)	C4119(2)	0.5	
	C4119(3)	1/3	
	C4119(4)	0.	
(2-83)	C4120(1)	0.664	
	C4120(2)	0.5	
	C4120(3)	1/3	
	C4120(4)	0	
(2-83)	C4121(1)	0.60	
	C4121(2)	0.5	
	C4121(3)	1/3	
	C4121(4)	2.0	
(2-83)	C4122(1)	0.037	
	C4122(2)	0.8	
	C4122(3)	1/3	
	C4122(4)	0.	
(2-83)	C4123(1)	0.037	
	C4123(2)	0.8	
	C4123(3)	1/3	
	C4123(4)	0.	
(2-83)	C4124(1)	0.60	
	C4124(2)	0.5	
	C4124(3)	1/3	
	C4124(4)	2.0	
(2-106)	C4151(1)	0.046	
	C4151(2)	1/3	
	C4151(3)	0.	
(2-106)	C4152(1)	0.046	
	C4152(2)	1/3	
	C4152(3)	0.	
(2-106)	C4153(1)	0.228	
	C4153(2)	0.226	
	C4153(3)	0.	
(2-106)	C4154(1)	0.046	
	C4154(2)	1/3	
	C4154(3)	0.	
(2-106)	C4155(1)	0.046	
	C4155(2)	1/3	
	C4155(3)	0.	
(2-106)	C4156(1)	0.228	
	C4156(2)	0.226	

Equation or Section	Coefficient	Value	Units
	C4156(3)	0.	
(2-106)	C4157(1)	0.59	
	C4157(2)	0.25	
	C4157(3)	0.	
(2-106)	C4158(1)	0.59	
	C4158(2)	0.25	
	C4158(3)	0.	
(2-106)	C4159(1)	0.43	
	C4159(2)	0.25	
	C4159(3)	2.0	
(2-106)	C4160(1)	0.10	
	C4160(2)	1/3	
	C4160(3)	0.	
(2-106)	C4161(1)	0.10	
	C4161(2)	1/3	
	C4161(3)	0.	
(2-106)	C4162(1)	0.43	
	C4162(2)	0.25	
	C4162(3)	2.0	
(2-107)	C4163(1)	8.235	
	C4163(2)	0.	
	C4163(3)	0.	
	C4163(4)	0.	
(2-107)	C4164(1)	48/11	
	C4164(2)	0.	
	C4164(3)	0.	
	C4164(4)	0.	
(2-107)	C4165(1)	48/11	
	C4165(2)	0.	
	C4165(3)	0.	
	C4165(4)	0.	
(2-107)	C4166(1)	0.023	
	C4166(2)	0.8	
	C4166(3)	1/3	
	C4166(4)	0.	
(2-107)	C4167(1)	0.023	
	C4167(2)	0.8	
	C4167(3)	1/3	
	C4167(4)	0.	
	C4168(1)	0.023	

Equation or Section	Coefficient	Value	Units
(2-107)	C4168(2)	0.8	
	C4168(3)	1/3	
	C4168(4)	0.	
(2-107)	C4169(1)	0.664	
	C4169(2)	0.5	
	C4169(3)	1/3	
	C4169(4)	0.	
(2-107)	C4170(1)	0.664	
	C4170(2)	0.5	
	C4170(3)	1/3	
	C4170(4)	0.	
(2-107)	C4171(1)	0.60	
	C4171(2)	0.5	
	C4171(3)	1/3	
	C4171(4)	2.0	
(2-107)	C4172(1)	0.037	
	C4172(2)	0.8	
	C4172(3)	1/3	
	C4172(4)	0.	
(2-107)	C4173(1)	0.037	
	C4173(2)	0.8	
	C4173(3)	1/3	
	C4173(4)	0.	
(2-107)	C4174(1)	0.60	
	C4174(2)	0.5	
	C4174(3)	1/3	
	C4174(4)	2.0	
(2-114)	C4180(1)	0.013	
	C4180(2)	0.5	
	C4180(3)	1.0	
	C4180(4)	0.33	
(2-116)	C4181(1)	0.18	
	C4181(2)	0.25	
	C4181(3)	0.5	
(2-117)	C4182(1)	0.09	
	C4182(2)	0.25	
	C4182(3)	0.5	
(2-118)	C4183(1)	0.943	
	C4183(2)	0.25	
	C4183(3)	0.75	

Equation or Section	Coefficient	Value	Units
(2-122)	C4184(1)	0.943	
Section 2.6.6	C4186(1)	1.0	
	C4186(2)	1.0	
	C4186(3)	1.0	
Section 2.7	C4200(1)	0.9995	
(2-134)	C4201(1)	1.0	
	C4201(2)	1.0	
	C4201(3)	1/3	
	C4201(4)	-1/3	
(2-137)	C4202(1)	$5.x10^5$	W/m^2-K
Section 2.7.3	C4203(1)	-1.0	
	C4203(2)	0.9	
Section 2.1.2	C4205(1)	10.	kg
Section 2.6.1.1	C4210(1)	30.	
	C4210(2)	100.	
	C4210(3)	(not used)	
	C4210(4)	0.1686289	
Section 2.6.1.1	C4211(1)	0.943	
	C4211(2)	0.25	
Section 2.6.1.1	C4212(1)	0.3333333	
	C4212(2)	-0.44	
	C4212(3)	$5.82x10^{-6}$	
	C4212(4)	0.8	
	C4212(5)	0.3333333	
	C4212(6)	0.5	
Section 2.6.1.1	C4213(1)	10^6	
	C4213(2)	10^8	
	C4213(3)	10^{10}	
Section 2.6.1.1	C4214(1)	0.6	
	C4214(2)	0.2	
Section 2.6.1.1	C4215(1)	0.72	
	C4215(2)	0.19	
Section 2.6.1.1	C4220(1)	30.	
	C4220(2)	100.	
	C4220(4)	0.9715642	
Section 2.6.1.1	C4221(1)	0.729	
	C4221(2)	0.25	
Section 2.6.1.1	C4222(1)	0.3333333	
	C4222(2)	-0.44	
	C4222(3)	$5.82x10^{-6}$	

Equation or Section	Coefficient	Value	Units
	C4222(4)	0.8	
	C4222(5)	0.3333333	
	C4222(6)	0.5	
Section 2.6.1.1	C4230(1)	30.	
	C4230(2)	100.	
Section 2.6.1.1	C4231(1)	0.815	
	C4231(2)	0.25	
Section 2.6.1.1	C4232(1)	0.3333333	
	C4232(2)	-0.44	
	C4232(3)	5.82×10^{-6}	
Section 2.6.1.1	C4232(4)	0.8	
	C4232(5)	.3333333	
	C4232(6)	0.5	
Section 2.8.1	C4251(1)	10^{-9}	
Section 2.8.2	C4251(2)	0.0005	
Sections 2.6.1.2 & 2.8.2	C4253(1)	0.909	
	C4253(2)	0.3333333	
	C4253(3)	0.115	
	C4253(4)	0.6	
	C4253(5)	1000.	
	C4253(6)	3000.	
	C4253(7)	-0.44	
	C4253(8)	5.82×10^{-6}	
	C4253(9)	0.8	
	C4253(10)	0.3333333	
	C4253(11)	0.5	

References

1. V. H. Ransom et al., RELAP5/MOD1 Code Manual Volume 1: System Models and Numerical Methods, NUREG/CR-1826, EGG-2070, R2, September 1981.

2 C. O. Bennett and J. E. Myers, Momentum, Heat, and Mass Transfer, 3rd ed., Mc-Graw-Hill, New York, 1982, p. 341.

3. D. K. Edwards, On the Use of Total Radiation Properties of Gases, ANL/RAS 75-12, April 1975.

4. D. K. Edwards and R. Matavosian, "Scaling Rules for Total Absorptivity and Emissivity of Gases, "Trans. ASME, JHT, Vol. 106, pp. 684-689, November 1984.

5. K. K. Murata, et al., "Code Manual for CONTAIN 2.0: A Computer Code for Nuclear Reactor Containment Analysis," NUREG/CR-6533, SAND97-1735, 1997.

6. J. G. Collier, Convective Boiling and Condensation, 2nd ed., McGraw-Hill, New York, 1981.

Lower Head Containment (LHC) Package

The Lower Head Containment (LHC) package models the degradation of a “second lower head” due to debris interaction. The effects of heat transfer on the mechanical properties of the support structure as well melting of the structure are modeled. The LHC is coupled to the Control Volume Hydrodynamic (CVH) package to establish thermal boundaries between the debris and the LHC structure.

This Reference Manual provides an overview of modeling in the LHC package. User input requirements for the LHC package are described in the LHC Users’ Guide.

Contents

List of Figures

1. Introduction

The Lower Head Containment (LHC) package models a “second lower head” sitting below/around the reactor pressure vessel lower head that may catch and contain ejected core debris. The user specifies LHC plate geometry (curvature, thickness, and transverse/through-wall plate nodalization), materials, and initial conditions. The user also associates control volumes with the space above/inside the plate and the space below/outside the plate. LHC communicates with COR and RN via TP, so user definitions of transfer processes are required. In the future, the user is able to optionally define an initial debris/melt inventory, the decay heat computation rules for debris, failure parameters, and certain sensitivity coefficients.

The goal of the LHC package is to ascertain the thermal-hydraulic response of the lower head containment plate and any debris ejected from the core. The LHC package must:

1. Receive debris (mass, energy) from COR via TP
2. Receive radionuclide inventory information (decay heat, RN transport)
3. Predict the various heat transfers between the ejected debris and both hydrodynamic materials and the LHC plate so as to ascertain debris temperatures
4. Predict debris phase change as a result of aforementioned heat transfers
5. Predict LHC plate failure by over-temperature, stress/strain considerations, etc.
6. Predict debris overflow/relocation to another LHC structure or a cavity in CAV

To facilitate a discussion of solution methodology, aspects of LHC modeling (plate structure, debris topology) are discussed in this section. Global LHC geometry and plate nodalization are addressed as are the concepts of a “debris cylinder” (for a flat LHC plate) and a “debris bowl” (for a curved LHC plate), both of which are comprised of a slurry region and crusts.

The LHC plate is either a flat or curved structure with some user-specified transverse and through-wall (radial and axial) nodalization and some material composition. The geometry is two-dimensional (i.e. there is azimuthal symmetry). Figure 1.1 and Figure 1.2 show details of the LHC plate geometry. In Figure 1.1, the various user-defined plate dimensions are clarified. The user gives dimensions RADOUT and RADLH as well as the elevation HGTLH. Taken together, these imply a plate curvature, segment length S, and the point of transition from curved/flat geometry to the vertical/cylindrical side-walls. In Figure 1.2, a section of a representative plate is shown and the transverse (coordinate r)/through-wall (coordinate z) nodalization is illustrated. For each transverse region NREGR there are one or more sub-regions DRR of uniform thickness and identical material. From region to region (but not from sub-region to sub-region within a given region) the user may define a different material and thickness. The situation is similar for

through-wall regions NREGZ and sub-regions DZR. There are MELGEN checks to ensure that user-defined geometry meets certain criteria, e.g. that (1) the sum of all transverse region widths equals the segment length S plus the length of the vertical side-wall and (2) one region boundary coincides with the point of transition from curved/flat to vertical geometry.

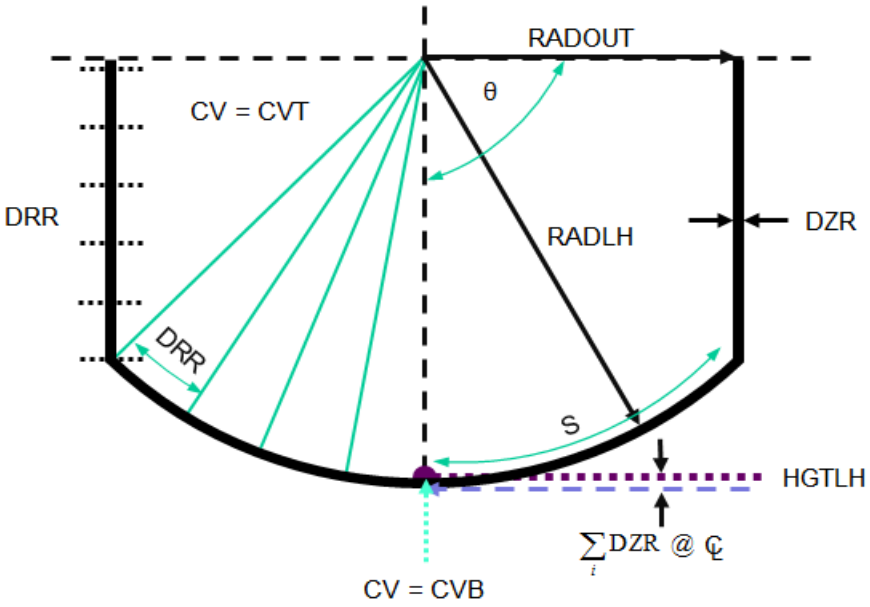


Figure 1.1 LHC Geometry, Depicts a Curved Plate that Transitions to a Vertical Wall

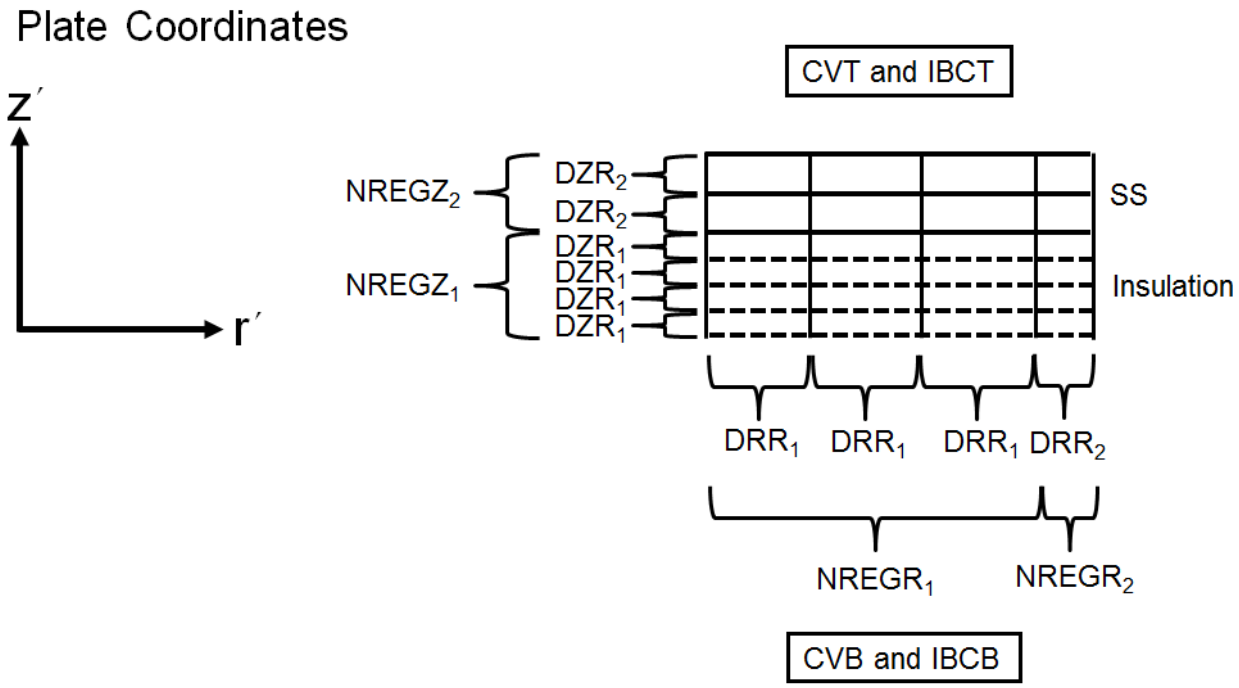


Figure 1.2 LHC Plate Nodalization Example Showing Transverse and Through-wall Regions and Sub-regions

If the LHC plate is flat, debris ejected to LHC from COR is conceptualized as a right circular cylinder consisting of an interior slurry zone and three crust zones (top, bottom, side). Azimuthal symmetry is assumed. Figure 1.3 is a half-section view of a general LHC debris cylinder. The various brackets delineate crust thicknesses, crust growth region thicknesses, and slurry region thickness. Names ending in “o” denote “old-time” dimensions and “ddel…” indicates growth by solidification of liquid mass in the slurry region (in general, crust melting is allowed too).

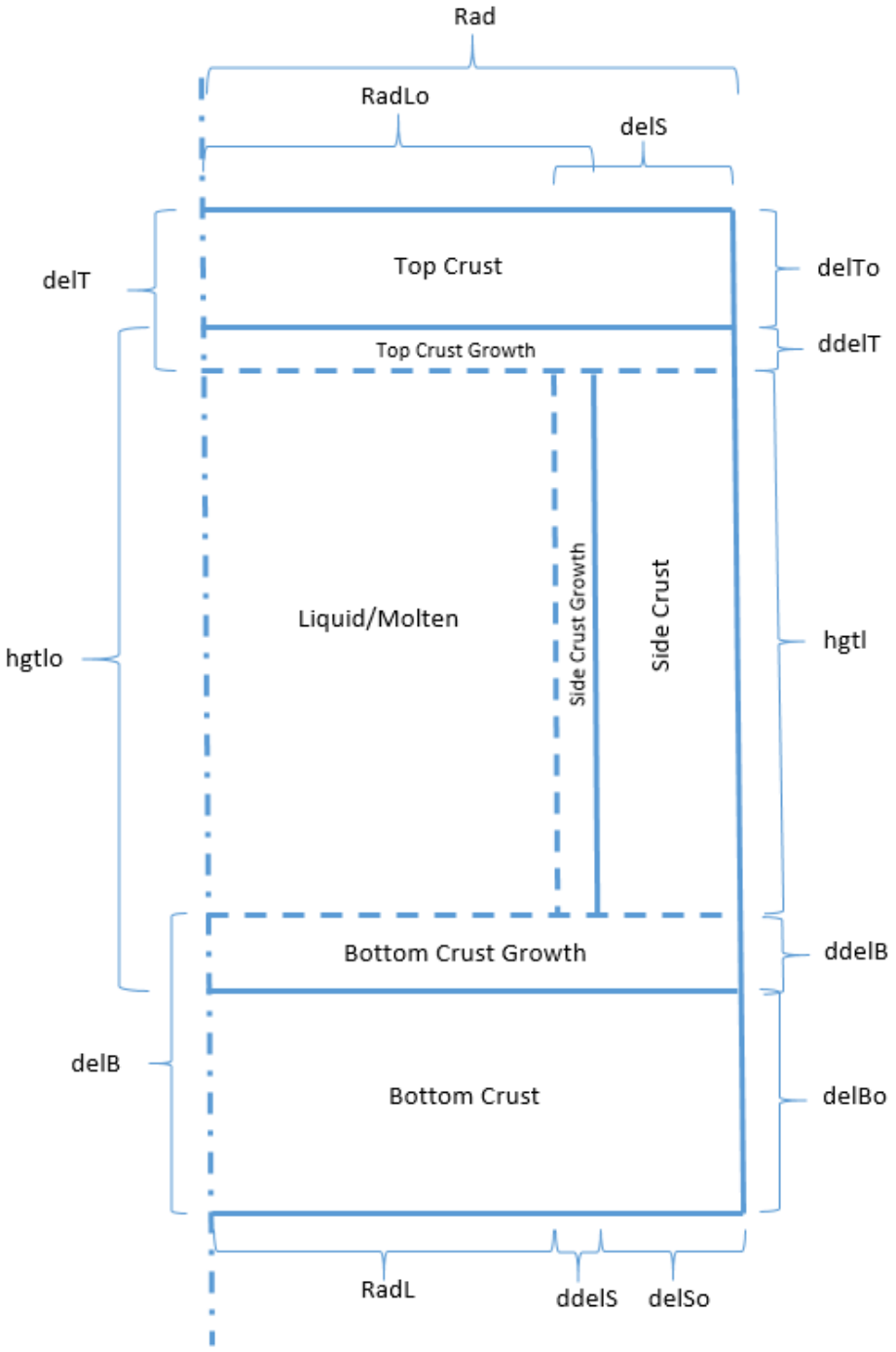


Figure 1.3 Debris Cylinder with Various Regions and Dimensions

If the LHC plate is curved, debris ejected to LHC from COR first assumes the shape of the “bowl” formed by LHC plate curvature. There is not debris “side” or lateral surface in this case. Rather, the debris has only a bottom (in contact with the curved LHC plate)

and a top. Azimuthal symmetry is still assumed and the concept of an interior slurry zone surrounded by crust zones (a top and a bottom) is still used for modeling purposes. Figure 1.4 is a half-section view of an LHC debris bowl. The various brackets delineate crust thicknesses, crust growth region thicknesses, and slurry region thicknesses. Names ending in "o" denote "old-time" dimensions and "ddel…" indicates growth by solidification of liquid mass in the slurry region (in general, crust melting is allowed too).

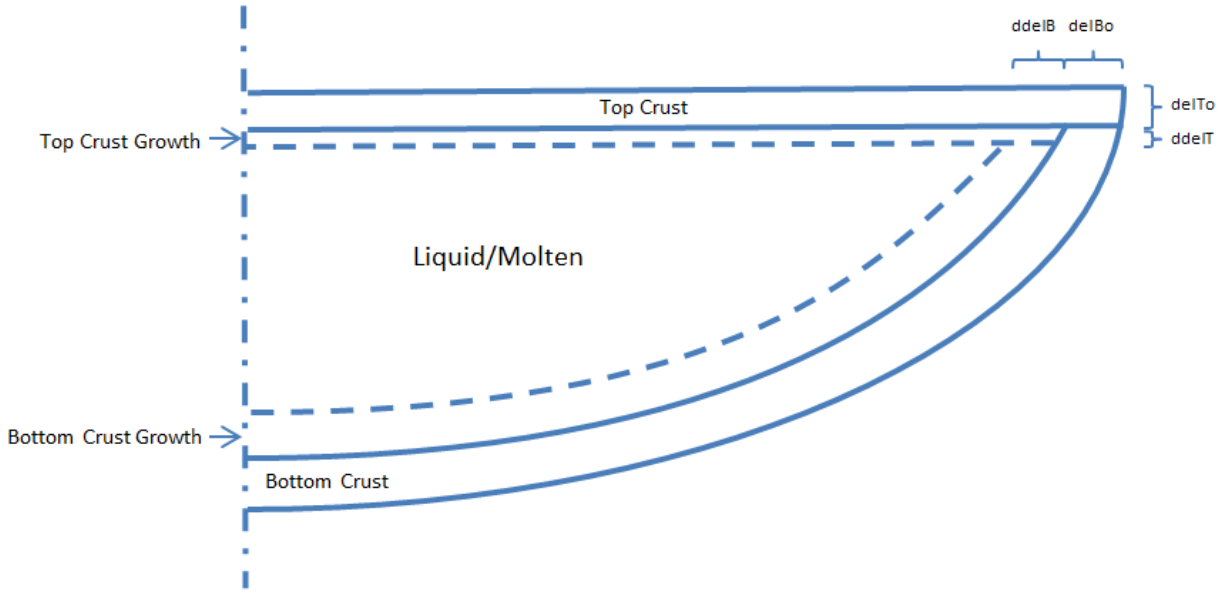


Figure 1.4 Debris Bowl with Various Regions and Dimensions

For purposes of computing masses, enthalpies, and temperatures, debris consists of: (1) the interior slurry mixture that can bear metal and oxide and is in general some mixture of solid and liquid, (2) crusts at the top, bottom, and side that can freeze (from molten debris slurry) or melt (to molten debris slurry). Any mass incoming from COR via TP is sent to the slurry zone inventory immediately, and crust may form from the liquid portion of that transferred mass if the balance equations should so predict. There is one temperature and one enthalpy for the slurry zone. A detailed discussion of debris physics models is deferred to a later section.

2. Models

Aspects of LHC physics modeling are discussed in greater detail below including property calculations, debris heat and mass transfer, LHC plate heat transfer, LHC/CVH energy exchange, and plate failure.

2.1 Debris mixture enthalpy and specific heat

Basic equations include (1) a running tally of mixture enthalpy that is tracked as the sum of the enthalpies of mixture components, (2) a running tally of mixture heat capacity that is tracked as the sum of the product of specific heat capacity and mass of each constituent component, (3) adjustments to both 1 and 2 because of latent heat effects.

$$h_{mix} = \sum_{i=1}^{12} h_i M_i \Big/ \sum_{i=1}^{12} M_i \tag{2-1}$$

$$Cp_{mix} = \sum_{i=1}^{12} Cp_i M_i \Big/ \sum_{i=1}^{12} M_i \tag{2-2}$$

Where

$$Cp_i = \begin{cases} Cp(T_L), & T_L \text{ outside } (T_{MELTS}, T_{MELTE}) \\ Cp(T_L) + \dfrac{h_{fs}}{0.1} = Cp_{avg,i}, & T_L \text{ inside } (T_{MELTS}, T_{MELTE}) \end{cases} \tag{2-3}$$

$$h_i = \begin{cases} h(T_L), & T_L \text{ outside } (T_{MELTS}, T_{MELTE}) \\ h(T_{MELTS}) + Cp_{avg,i}(T_L - T_{MELTS}), & T_L \text{ inside } (T_{MELTS}, T_{MELTE}) \end{cases} \tag{2-4}$$

$i =$ material component ; $h_i =$ enthalpy $\left[\frac{J}{kg}\right]$; $Cp_i =$ specific heat$[\frac{\text{J}}{\text{kg*K}}]$

$T_L =$ debris slurry temperature [K] ; h_{fs} = latent heat of fusion $[\frac{\text{J}}{\text{kg}}]$

T_{MELTS} = component melt temperature [K] ; $T_{MELTE} = T_{MELTS} + 0.1\ [K]$

2.2 Property/Geometry Calculations

Oxide and metal densities are computed for debris added to LHC from COR/TP. For oxide/metal density:

$$\rho_{ox/met} = 1000 * \frac{\sum_i [M_i]}{\sum_i \left[\chi_i V_{m,i} (1 + \beta_i)\left(T_{avg} - 1673\right)\right]} \tag{2-5}$$

For average volumetric expansion coefficient:

$$\beta_{ox/met} = \frac{\sum_i [\chi_i V_{m,i} \beta_i]}{\sum_i [\chi_i V_{m,i} (1 + \beta_i)(T_{avg} - 1673)]} \quad (2\text{-}6)$$

In the equations above: $M, \chi, V_m, \beta, T_{avg}$ are mass, mole fraction (relative to all the metal/oxide moles), molar volume, volumetric expansion coefficient, and oxide/metal mass-averaged temperature, respectively. The factor of 1000 appearing in the density equation is applied for dimensional consistency.

The method of computing dimensions for the conceptual debris cylinder is:

- Use the known debris volume to perform a volume/altitude table look-up in CVolT, the LHC CV above the LHC plate, and determine debris cylinder height
- Use the debris volume and the just-computed height with the formula for right circular cylinder volume to compute a debris cylinder radius
- Apply dimension limiting and set stop-transfer flags as necessary
- Note that a physics-based spreading model similar to that of CAV may soon be included

The method of computing dimensions for the conceptual debris bowl is:

- Use the known debris volume to perform a volume/altitude table look-up in CVolT, the LHC CV above the LHC plate, and determine debris bowl height
- Use known formulae derived from spherical coordinates to compute a debris bowl radius (the radius of the circular top face) and an arc length of debris coverage along the plate
- Apply dimension limiting and set stop-transfer flags as necessary
- Note that, at present, no physics-based spreading model is used

2.3 Average Temperature, Debris Solids Fractions

The existing LHC debris inventory is combined with incoming debris from COR so as to update debris average and liquid/molten region temperatures, enthalpies, and masses. Masses are incremented and temperatures are computed as mass averages. Enthalpies and specific heats are updated after temperature calculations are performed. Note these operations occur before any debris heat/mass transfer calculations (crust dynamics, plate energy transfers, CVH energy transfers).

With $xmass$ the existing LHC debris mass inventory, $xmtrn$ the mass transferred from COR in this step, and T_{COR} the debris temperature passed from COR, the debris slurry temperature is:

$$T_L = \frac{xmasL * T_{liq/molten}^{old} + xmtrn * T_{COR}}{xmasL + xmtrn} \tag{2-7}$$

Note $xmasL$ is the existing total slurry zone mass accounting for current top/bottom/side crust mass. Also, the slurry zone receives all incoming debris mass from COR (generally a mixture of solid and liquid).

The solids fractions described below apply to the slurry zone total inventory $xmasL$ such that $\phi_{s,m}$ is the total solid metal mass fraction in the slurry zone, $\phi_{s,o}$ is the total solid oxide mass fraction in the slurry zone, and $\phi_s = (\phi_{s,m} + \phi_{s,o})$ is the total solid mass fraction in the slurry zone. Analogously, there are liquids fractions (total, metal, oxide). A mass averaging scheme involving just-transferred and pre-existing fractions is used:

$$\phi_{s,m} = \frac{\phi_{s,m_{xfr}} * xmtrn + \phi_{s,m_{old}} xmasL_{old}}{xmasL} \tag{2-8}$$

$$\phi_{s,o} = \frac{\phi_{s,o_{xfr}} * xmtrn + \phi_{s,o_{old}} xmasL_{old}}{xmasL} \tag{2-9}$$

$$\phi_s = \frac{\left(\phi_{s,o_{xfr}} + \phi_{s,m_{xfr}}\right) * xmtrn + \phi_{s_{old}} xmasL_{old}}{xmasL} \tag{2-10}$$

And there are analogous equations for $\phi_{l,m}, \phi_{l,o}, \phi_l = (\phi_{l,m} + \phi_{l,o})$

2.4 Debris Property Calculations

The following properties of the debris metal and debris oxide are separately computed by mole fraction weighting:

- Viscosity [Pa*s] via function Get_VISCMX
- Thermal conductivity via function Get_ThermalCondMix
- Surface tension via function Get_SigMix
- Emissivity via function Get_Emiss
- Density and coefficient of volumetric thermal expansion via function Get_DensityMix

Then, mixture (metal plus oxide) averages for all above quantities are computed by either volume weighting (viscosity, thermal conductivity, and emissivity) or mass weighting (surface tension).

2.5 Debris Heat and Mass Transfer

2.5.1 Crust Dynamics

Crust dynamics (formation and growth of solid crust at the debris slurry periphery) and debris temperature response depend on the thermal energy generation within debris (due to decay heat generation) and the thermal energy removal from debris (due to CVH and LHC plate transfers).

Slurry zone heat transfer coefficients are required to determine heat flows from the slurry to either the crusts at the debris periphery, the plate at the bottom of bare debris (without an intervening bottom crust), or the CVH pool/atmosphere at the top/side of bare debris (without an intervening top/side crust). When the slurry zone has solidified (solids fraction very close to unity), the coefficients come from dividing the debris-average thermal conductivity by a length characterizing the direction of conduction heat transfer (debris height for axial conduction, debris radius for radial conduction). When the slurry contains molten material, heat transfer coefficients are obtained from natural circulation correlations for a flat plate (hot surface facing up/down, cold surface facing up/down, etc). For purposes of estimating heat transfer coefficients in this case, slurry zone periphery temperatures are needed and are obtained from either old-time information or from steady-state considerations of heat flux continuity at the boundary in question assuming conduction heat transfer. For the top and bottom surfaces, respectively, these initializations assume the form of:

$$T_B = \frac{T_{pl}\left(\frac{2k_{pl}}{dyc_{nnz}}\right) + T_L\left(\frac{2k_m}{H_L}\right)}{\frac{2k_m}{H_L} + \frac{2k_{pl}}{dyc_{nnz}}} \tag{2-11}$$

$$T_T = \frac{T_{bnd}h_{bnd} + T_L\left(\frac{2k_m}{H_L}\right)}{\frac{2k_m}{H_L} + h_{bnd}} \tag{2-12}$$

where

T_B = Bottom crust temp[K], T_T = Top crust temp[K], same as side crust T_R

T_{pl} = Plate temperature (avg over melt-covered nodes) [K]

k_{pl}, k_m = Plate (avg over melt-covered nodes) and debris thermal conductivity $\left[\frac{\text{W}}{\text{m*K}}\right]$

dyc_{nnz} = top-most plate node thickness [m] , H_L = Debris slurry region height [m]

T_{bnd} = Boundary (pool or atm) temp [K] , h_{bnd} = Boundary heat transfer coefficient $\left[\frac{\text{W}}{\text{m}^2\text{*K}}\right]$

Crust dynamics calculations proceed similarly for top/side and bottom crusts. The calculations are based on an analytical solution for conduction in a moving solid (i.e. a solid with a moving boundary as occurs when slurry adjacent to crust freezes/melts to grow/shrink the crust thickness). Some pool or atmosphere bounds the other side of the crust. The governing equation is:

$$\frac{d^2T(z)}{dz^2} - \left(\frac{u}{\alpha}\right)\frac{dT(z)}{dz} = -\frac{Q}{k} \tag{2-13}$$

where

$T(z)$ = Temperature in solid [K] , z = Thru-wall or axial coordinate [m]

u = Velocity of inner surface of solid $\left[\text{m}/\text{s}\right]$, $\alpha = \frac{k}{\rho C_p}$ = thermal diffusivity of solid $\left[\text{m}^2/\text{s}\right]$

Q = Energy generation in solid $\left[\text{W}/\text{m}^3\right]$, k = Thermal conductivity of solid $\left[\text{W}/\text{m*K}\right]$

It can be shown that a temperature profile $T(z) = T_p(z) + T_h(z)$ that is the sum of a particular and a homogeneous solution satisfies the governing equation subject to boundary conditions: (1) $T(z=0) = T_{sol}$, the solidus temperature of adjacent freezing/melting material, and (2) $T(z=\delta) = T_{bnd}$, the boundary (pool or atmosphere) temperature. The solution temperature profile is:

$$T(z) = T_{sol} - \left[T_{sol} - T_{bnd} + \frac{Q\delta}{\rho C_p u}\right]\left[\frac{\exp\left(\frac{uz}{\alpha}\right) - 1}{exp\left(\frac{u\delta}{\alpha}\right) - 1}\right] + \frac{Qz}{\rho C_p u} \tag{2-14}$$

An equation for the boundary velocity, u, can be obtained by returning to the governing equation and using the facts that:

$$-k\frac{dT}{dz}_{z=0} = \rho_m h_{fs} u + q''_{slr} \quad and \quad -k\frac{dT}{dz}_{z=\delta} = q''_{bnd} \tag{2-15}$$

to obtain

$$u = \frac{q''_{bnd} - q''_{slr} - Q\delta}{\rho_m \left(h_{fs} - C_{pm}(T_{bnd} - T_{sol})\right)} \tag{2-16}$$

where

ρ_m = Debris density $\left[{}^{kg}/_{m^3}\right]$,

h_{fs} = Latent heat of fusion for debris $\left[{}^{J}/_{kg}\right]$

C_{pm} = Specific heat capacity of debris $\left[{}^{J}/_{kg*K}\right]$

δ = Crust thickness [m]

q''_{slr} = Heat flux from slurry to crust at phase-change boundary $\left[{}^{W}/_{m^2}\right]$

q''_{bnd} = Heat flux from bounding pool/atm at other crust boundary $\left[{}^{W}/_{m^2}\right]$

To obtain a single “crust average” temperature from the solution profile $T(z)$ an averaging can be done:

$$\bar{T} = \frac{\int_0^{\delta} T(z)dz}{\delta} \tag{2-17}$$

This is essentially a volume averaging in Cartesian coordinates. When applied to each term of the temperature profile equation, a crust averaged temperature is recovered as:

$$\bar{T} = T_{sol} - \left[\frac{T_{sol} - T_{bnd} + \frac{Q\delta}{\rho C_p u}}{\exp\left(\frac{u\delta}{\alpha}\right) - 1}\right]\left[\left(\frac{\alpha}{u\delta}\right)\left(exp\left(\frac{u\delta}{\alpha}\right) - 1\right) - 1\right] + \frac{Q\delta}{2\rho C_p u} \tag{2-18}$$

Thus, for any of the top, bottom, or side crusts a system of two equations can be solved in a given time-step to recover a crust average temperature and a crust velocity (and thus a change in thickness, a total thickness, and a mass). Within the iterative Newton’s method solution to the two-equation system, the crust thickness and boundary heat

transfer coefficient can be updated according to improved estimates of crust temperature and crust velocity.

The crust dynamics equations are only solved if the slurry boundary temperature suggests a crust should begin to form or should grow/shrink from its old-time state. Comparison with the debris slurry solidus temperature indicates what should occur.

2.5.2 Debris and CVH/Plate Heat Transfer

Enthalpy and temperature of the debris slurry is updated subsequent to crust dynamics calculations according to the total energy transfer from all slurry boundaries. An energy balance dictates the change in enthalpy and the debris-average specific heat can then be used to compute a change in slurry temperature. Energy transfer tracking variables for debris heat transfer are tallied for purposes of energy error computation.

2.6 Plate Heat Transfer

2.6.1 Governing Equation

A two-dimensional cylindrical conduction problem is solved via a finite-volume method on a sub-divided grid. The heat conduction equation is:

$$\frac{\partial H}{\partial t} = \frac{1}{r}\frac{\partial}{\partial r}\left(rk(T)\frac{\partial T}{\partial r}\right) + \frac{\partial}{\partial z}\left(k(T)\frac{\partial T}{\partial z}\right) \qquad (2\text{-}19)$$

where

H = Node enthalpy $\left[{}^{J}/_{m^3}\right]$

r = radial coordinate [m]

z = Axial coordinate[m]

T = Node Temperature [K]

k = Node thermal conductivity $\left[{}^{W}/_{m*K}\right]$

It can be integrated over cylindrical coordinates r (from r_i to r_{i+1}) , θ (from 0 to 2π), and z (from z_j to z_{j+1}). The result with an additional integration over time t (from t to t+Δt), for any interior plate node with i denoting radial/transverse node i and j denoting axial/through-wall node j, is:

$$V_{i,j}\int_{t}^{t+\Delta t}\frac{\partial H}{\partial t}dt=\int_{t}^{t+\Delta t}A_z^{i,j}\left[k_n\left(\frac{T_{i,j+1}-T_{i,j}}{dy_{j+1}}\right)-k_s\left(\frac{T_{i,j-1}-T_{i,j}}{dy_j}\right)\right]dt$$
$$+\int_{t}^{t+\Delta t}\left[A_r^{i+1,j}\left[k_e\left(\frac{T_{i+1,j}-T_{i,j}}{dx_{i+1}}\right)\right]\right. \quad (2\text{-}20)$$
$$\left.-A_r^{i,j}\left[k_w\left(\frac{T_{i-1,j}-T_{i,j}}{dx_i}\right)\right]\right]dt$$

where

$A_Z^{i,j}$ = node i,j axial area $[\mathrm{m}^2]$

$A_R^{i,j}$ = node i,j radial area $[\mathrm{m}^2]$

T = temperature [K]

$\rho c_p\left(T_{i,j}^o\right)V_{i,j}$ = volumetric heat capacity $\left[\frac{J}{m^3K}\right]$ multiplied by node volume $[\mathrm{m}^3]$

k = thermal conductivity $\left[\frac{W}{m\,K}\right]$

Node axial and radial areas are computed from the user-supplied LHC plate geometry. For a flat plate, surface and volume integrals in cylindrical coordinates are used to derive expressions for node areas and volumes. For a curved plate, surface and volume integrals in spherical coordinates are used. In either case, the vertical, cylindrical section of the plate is comprised of nodes defined by cylindrical geometry.

The enthalpy H depends on a node volume liquid fraction g [$\mathrm{m}^3_{\mathrm{liq}}/\mathrm{m}^3_{\mathrm{node}}$] such that:

$$H(g)=\begin{cases}\int_{T_{ref}}^{T}\rho_s(T')C_{ps}(T')dT', & T<T_{MELTS}\\ (1-g)\int_{T_{ref}}^{T}\rho_s(T')C_{ps}(T')dT'+g\int_{T_{ref}}^{T}\rho_l(T')C_{pl}(T')dT'+\rho_l(T)gh_{fs}, & T\epsilon(T_{MELTS},T_{MELTE})\\ \int_{T_{ref}}^{T}\rho_l(T')C_{pl}(T')dT'+\rho_l(T)h_{fs}, & T>T_{me}\end{cases} \quad (2\text{-}21)$$

The node liquid volume fraction is defined as:

$$g = \begin{cases} 0, & for\ T < T_{MELTS} \\ \dfrac{T - T_{MELTS}}{T_{MELTE} - T_{MELTS}}, & for\ T\epsilon(T_{MELTS}, T_{MELTE}) \\ 1, & for\ T > T_{MELTE} \end{cases} \quad (2\text{-}22)$$

2.6.2 Time Discretization

Applying a fully-implicit time discretization scheme, the time integrals above can be evaluated:

$$\begin{aligned} \frac{V_{i,j}}{\Delta t}\left(H_{i,j}^{n} - H_{i,j}^{o}\right) & \\ = A_z^{i,j}\left[k_n\left(\frac{{T_{i,j+1}}^{n} - {T_{i,j}}^{n}}{dy_{j+1}}\right) - k_s\left(\frac{{T_{i,j-1}}^{n} - {T_{i,j}}^{n}}{dy_j}\right)\right] & \\ + A_r^{i+1,j}\left[k_e\left(\frac{{T_{i+1,j}}^{n} - {T_{i,j}}^{n}}{dx_{i+1}}\right)\right] + A_r^{i,j}\left[k_w\left(\frac{{T_{i-1,j}}^{n} - {T_{i,j}}^{n}}{dx_i}\right)\right] & \end{aligned} \quad (2\text{-}23)$$

Note that only new-time (superscript n) temperature quantities appear on the right-hand side of the nodal equation. The geometric interpretations for node indices and distances dy and dx are made clearer in a subsequent description of the solution grid.

2.6.3 Enthalpy Term Linearization

The new-time enthalpy must be linearized via an expansion of new-time enthalpy in a Taylor series so that it can be written in terms of new-time temperature in the nodal equation:

$$H_{i,j}^{n} = H_{i,j}^{n-1} + \left(\frac{dH}{dT}\right)_{H_{i,j}^{n-1}}\left({T_{i,j}}^{n} - {H_{i,j}}^{-1}{}_{H_{i,j}^{n-1}}\right) \quad (2\text{-}24)$$

Thus the source terms in the nodal equation involve an enthalpy derivative and an "inverse enthalpy", which is actually just the temperature that corresponds to the last-iterate enthalpy $H_{i,j}^{n-1}$. The value of ${H_{i,j}}^{-1}{}_{H_{i,j}^{n-1}}$ can be recovered by a table look-up where temperature is found from enthalpy data. The value of $\left(\frac{dH}{dT}\right)_{H_{i,j}^{n-1}}$ comes from differentiating the piece-wise expression for $H(g)$ from above.

2.6.4 Spatial Discretization and Finite Volumes

The computational grid takes the general form shown in Figure 2.1. It is two-dimensional cylindrical and azimuthal symmetry is assumed. A "node" is the physical center of a finite volume and the volume dimensions determine the exact location of the volume interfaces with its nearest neighbors to the north (increasing j), south (decreasing j), east (increasing

i), and west (decreasing i). There are no a priori restrictions on region-to-region (e.g. NREGZ to NREGZ+1) node spacing in either of the two directions (but recall nodes are equally spaced within a region, e.g. NREGZ). If the LHC plate is curved, the appropriate transverse/through-wall areas, node volumes, and node-to-node distances are computed from formulae derived with double and triple integrals in spherical coordinates. Note the arrangement of the finite volumes on the grid is such that boundary nodes coincide with boundary volume interfaces, thus obviating any need for “one-and-a-half” volumes or ghost volumes at the boundaries. This arrangement also allows known boundary condition information to be used more directly in the discretized nodal temperature equations.

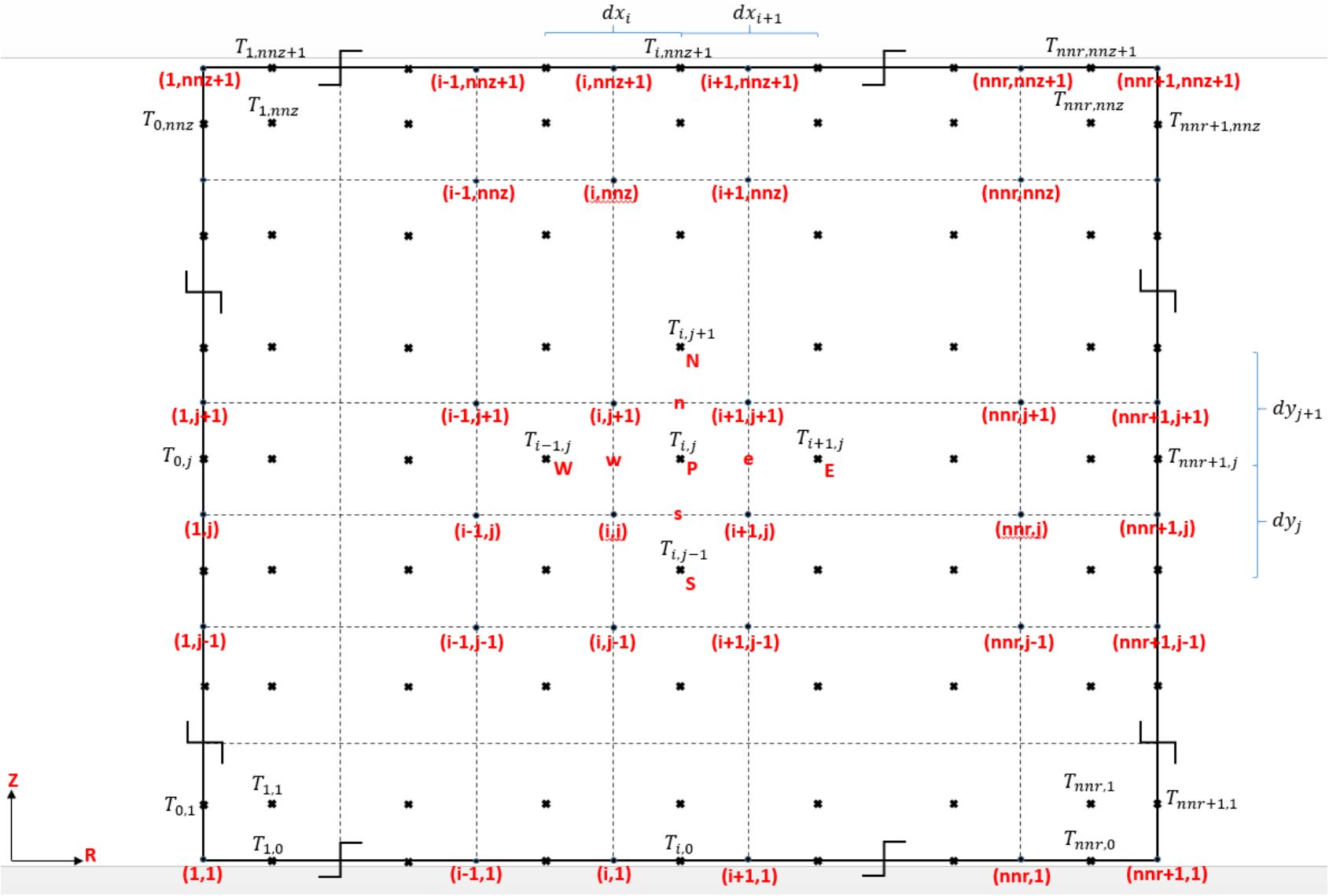


Figure 2.1 Generic Computational Grid for LHC Plate Conduction Solution

2.6.5 Nodal Temperature Equation

Working with the time-discretized equation above, defining several coefficients, and implementing the new-time enthalpy linearization, the interior nodal equation becomes:

$$\left(A_p^{i,j} - S_p\right)T_{i,j}^n = A_N^{i,j}T_{i,j+1}^n + A_S^{i,j}T_{i,j-1}^n + A_E^{i,j}T_{i+1,j}^n + A_W^{i,j}T_{i-1,j}^n + S_c \tag{2-25}$$

Where the source term has been split into two parts:

$$S_p = -\frac{V_{i,j}}{\Delta t}\left(\frac{dH}{dT}\right)_{H_{i,j}^{n-1}} \tag{2-26}$$

$$S_c = \frac{V_{i,j}}{\Delta t}\left(H_{i,j}^o - H_{i,j}^{n-1}\right) + \frac{V_{i,j}}{\Delta t}\left(\frac{dH}{dT}\right)_{H_{i,j}^{n-1}} {H_{i,j}}^{-1}{}_{H_{i,j}^{n-1}} \tag{2-27}$$

Algebraically re-arranging for a successive over-relaxation solution with ω a successive over-relaxation factor, an equation for temperature of interior node i,j is:

$$T_{i,j}^n = T_{i,j}^*(1-\omega) + \frac{\omega}{A_p^{i,j} - S_p}\left(\sum_{nb} A_{nb}T_{nb}^n + b^{i,j} + S_c\right) \tag{2-28}$$

In the previous two equations:

$$A_p^{i,j} = A_N^{i,j} + A_S^{i,j} + A_E^{ij} + A_W^{i,j} \tag{2-29}$$

$$A_N^{i,j} = \frac{A_Z^{i,j}k_n}{dy_{j+1}}\ ;\ A_S^{i,j} = \frac{A_Z^{i,j}k_s}{dy_j}\ ;\ A_E^{i,j} = \frac{A_R^{i,j}k_e}{dx_{i+1}}\ ;\ A_W^{i,j} = \frac{A_R^{i,j}k_w}{dx_i}\text{, neighbor conductance coefficients} \tag{2-30}$$

$$b^{i,j} = \frac{V_{i,j}}{\Delta t}H_{i,j}^o \tag{2-31}$$

where

$k_\alpha =$ thermal conductivity at $\alpha\ (= n, s, e, w)$, i.e. at finite volume interfaces $\left[\frac{W}{mK}\right]$

$\omega =$ successive over-relaxation (SOR)factor ;

$nb =$ neighbor ;

n = new time;

o = old time

The conductance coefficients contain node-to-node conduction heat transfer linkage information and encapsulate the grid geometry. They require finite volume face areas, the distance between neighboring nodes, and volume-to-volume interface conductivity. Because there is no requirement in general for equal node spacing between nodes of adjoining regions, an appropriate distance-weighting scheme is used to calculate interface conductivity. Using data stored at the finite volume centers (where temperature is known) and using node-to-node distance, a distance weighting can be applied. If two neighboring nodes are equally spaced, the interface is midway between the nodes and the distance-weighted interface conductivity is a harmonic mean.

Subsequent to solution of the nodal equations for new-time temperature, the enthalpy is updated all over the grid as:

$$H_{i,j}^{n} = H_{i,j}^{n-1} + \left(\frac{dH}{dT}\right)_{H_{i,j}^{n-1}} \left({T_{i,j}}^{n} - {H_{i,j}}^{-1}{}_{H_{i,j}^{n-1}} \right) \tag{2-32}$$

2.6.6 Boundary Conditions

Conductance coefficients for boundary control volumes are manipulated according to boundary condition (specified heat flux, specified heat transfer coefficient, compound heat transfer coefficient, or mixed). Thus, there are different nodal equations for boundary volumes.

2.6.6.1 Heat Flux Condition

As an illustration of a heat flux boundary condition, consider the nodal equation re-arranged slightly and with the northern term on the right-hand side replaced to include the heat flux information:

$$\begin{aligned} \frac{V_{i,j}}{\Delta t}\left(H_{i,j}^{n} - H_{i,j}^{o}\right) & \\ = q_N'' A_z^{i,j+1} + A_S^{i,j}\left(T_{i,j-1}^{n} - T_{i,j}^{n}\right) + A_E^{i,j}\left(T_{i+1,j}^{n} - T_{i,j}^{n}\right) & \\ + A_W^{i,j}\left(T_{i-1,j}^{n} - T_{i,j}^{n}\right) & \end{aligned} \tag{2-33}$$

By equating the known conduction heat flux at the boundary to the known expression for conduction heat flux in terms of temperature, interface conductivity, and nodal spacing:

$$q_N'' = \frac{k_n}{dy}\left(T_N^n - T_{i,j}^n\right) = \frac{A_N^{i,j}}{A_z^{i,j+1}}\left(T_N^n - T_{i,j}^n\right) \tag{2-34}$$

Then, upon solving for the boundary node temperature:

$$T_N^n = \frac{q_N'' dy}{k_n} + T_{i,j}^n \tag{2-35}$$

Therefore, to get the specialized nodal equation for $T_{i,j}^n$ (belonging to a finite volume on the Northern boundary) from the general nodal equation:

- $A_p^{i,j}$ must be modified to exclude $A_N^{i,j}$,
- $b^{i,j}$ must be modified to include $q_N'' A_z^{i,j+1}$, and
- $A_N^{i,j}$ must be set to zero

A similar treatment would apply for the Southern boundary if necessary. Note that an adiabatic condition is just a special case of the heat flux condition with the boundary heat flux equal to zero.

2.6.6.2 Heat Transfer Coefficient Condition

The heat transfer coefficient condition implementation is more algebraically complex. By enforcing a heat flux continuity and equating the boundary heat flux to the conduction heat flux between node edge and node center, the form of the nodal equation is evident:

$$\begin{aligned} q_N'' &= \frac{k_n}{dy}\left(T_N^n - T_{i,j}^n\right) \\ &= h_N\left(T_{bndN} - T_N^n\right), eliminating\ T_N^n\ and\ substituting: \end{aligned} \tag{2-36}$$

$$\begin{aligned} \frac{\rho c_p(T) V_{i,j}}{\Delta t}\left(T_{i,j}^n - T_{i,j}^o\right) &= A_S^{i,j}\left(T_{i,j-1}^n - T_{i,j}^n\right) + A_E^{i,j}\left(T_{i+1,j}^n - T_{i,j}^n\right) \\ &+ A_W^{i,j}\left(T_{i-1,j}^n - T_{i,j}^n\right) + \frac{h_N A_z^{i,j+1} T_{bndN} + T_{i,j}^n A_N^{i,j}}{\frac{h_N A_z^{i,j+1}}{A_N^{i,j}} + 1} \end{aligned} \tag{2-37}$$

The boundary temperature can then be solved for directly:

$$T_N^n = \frac{\frac{h_N A_z^{i,j+1} T_{bndN}}{A_N^{i,j}} + T_{i,j}^n}{\frac{h_N A_z^{i,j+1}}{A_N^{i,j}} + 1} \tag{2-38}$$

As for the process of modifying conductance coefficients:

- Subtract from $A_P^{i,j}$ the quantity $\frac{A_N^{i,j}}{\frac{h_N dy_{j+1}}{k_n}+1}$
- Add to $b^{i,j}$ the quantity $\frac{A_N^{i,j} T_{bndN}}{\left(\frac{h_N dy_{j+1}}{k_n}\right)^{-1}+1}$
- Set $A_N^{i,j}$ to zero

A similar treatment would apply for the Southern boundary.

2.6.6.3 Mixed Heat Flux and Heat Transfer Coefficient Condition (Debris Coverage)

A mixed heat flux and heat transfer coefficient boundary condition is needed for the LHC plate top surface because often the edge of the debris lies somewhere in between plate node boundaries. When this occurs, a covered area fraction is computed for the partially-covered node. The debris heat flux is applied to the covered area while a heat transfer coefficient is applied to the uncovered area as appropriate for either pool or atmosphere coverage. The derivation of this boundary condition proceeeds as before but is more algebraically complex:

$$\begin{aligned} &q_N'' f A_z^{i,j+1} + (1-f) h_N (T_{bndN} - T_N^n) A_z^{i,j+1} \\ &\qquad = \frac{k_n}{dy}\left(T_N^n - T_{i,j}^n\right), eliminating\ T_N^n\ and\ substituting: \end{aligned} \tag{2-39}$$

$$\begin{aligned} &\frac{\rho c_p(T) V_{i,j}}{\Delta t}\left(T_{i,j}^n - T_{i,j}^o\right) \\ &\qquad = A_S^{i,j}\left(T_{i,j-1}^n - T_{i,j}^n\right) + A_E^{i,j}\left(T_{i+1,j}^n - T_{i,j}^n\right) + A_W^{i,j}\left(T_{i-1,j}^n - T_{i,j}^n\right) \\ &\qquad - T_{i,j}^n A_N^{i,j} + \frac{q_N'' f A_z^{i,j+1} + (1-f) h_N A_z^{i,j+1} T_{bndN} + T_{i,j}^n A_N^{i,j}}{\frac{(1-f) h_N A_z^{i,j+1}}{A_N^{i,j}} + 1} \end{aligned} \tag{2-40}$$

The boundary temperature can then be solved for directly:

$$T_N^n = \frac{q_N'' f A_z^{i,j+1} + (1-f) h_N A_z^{i,j+1} T_{bndN} + T_{i,j}^n A_N^{i,j}}{(1-f) h_N A_z^{i,j+1} + A_N^{i,j}} \quad (2\text{-}41)$$

As for the process of modifying conductance coefficients:

- Subtract from $A_P^{i,j}$ the quantity $\frac{A_N^{i,j}}{\frac{(1-f)h_N A_z^{i,j+1}}{A_N^{i,j}}+1}$
- Add to $b^{i,j}$ the quantity $\frac{q_N'' f A_z^{i,j+1} + (1-f) h_N A_z^{i,j+1} T_{bndN}}{\frac{(1-f)h_N A_z^{i,j+1}}{A_N^{i,j}}+1}$
- Set $A_N^{i,j}$ to zero

Note for the mixed heat flux and heat transfer coefficient boundary condition an area fraction f is used to describe the extent of debris coverage for a given node. The value of this fraction is unity if the node is fully-covered with debris and is computed from spherical (curved LHC plate) or cylindrical (flat LHC plate) coordinates formulae if the node is only partially covered by debris.

2.6.6.4 Compound Heat Transfer Coefficient Condition and Critical Pool Fractions

One last boundary condition formulation is required for both the LHC plate top and bottom surfaces. It is similar to a specified heat transfer condition and applies when a plate node is partially covered by pool and partially covered by atmosphere (no debris coverage allowed). Its implementation depends on user-defined LHC critical pool fractions that are similar in function to the critical pool fractions that apply to heat structures as discussed in Section 2.4 of the Heat Structure Package reference manual.

Essentially, four user-defined critical pool fractions – two for the North (inside) surface and two for the South (outside) surface – dictate the phase (pool, atmosphere, or both) with which a given surface node exchanges energy. The pool area coverage fraction for a node is computed from plate geometry and known pool height (indexed to zero at LHC elevation HGTLH). If the area coverage fraction for a node exceeds the critical pool fraction for the pool (specified for that node's side of the LHC plate), then heat transfer to the pool can occur. Otherwise, heat transfer exclusively to the atmosphere can occur with the pool-covered area effectively insulated. Conversely if the area coverage fraction for a node exceeds the critical pool fraction for the atmosphere (specified for that node's side of the LHC plate), then heat transfer to the atmosphere cannot occur and heat transfer exclusively with the pool occurs.

Thus it is possible for a partially-covered node to exchange heat with:

1) Only the pool across its covered area,
2) Only the atmosphere across its area uncovered by pool,
3) Both the pool and the atmosphere across their appropriate areas, and
4) Neither pool nor atmosphere (an effectively insulated node surface)

The compound heat transfer coefficient boundary condition is meant to account for the possibility of simultaneous pool/atmosphere heat transfer when different heat transfer coefficients characterize energy exchange with each phase. When one phase is excluded due to critical pool fraction settings, this condition reduces to that of a single specified heat transfer coefficient although with a fractional multiplier accounting for reduced area. When both phases are excluded, the condition reduces to that of a specified zero heat flux. When heat transfer to both phases occurs, heat flux continuity requires:

$$q_N'' = \frac{k_n}{dy}\left(T_N^n - T_{i,j}^n\right) = A_{wat}h_{N,pool}\left(T_{N,pool} - T_N^n\right) + \left(1 - A_{wat}\right)h_{N,atm}\left(T_{N,atm} - T_N^n\right) \quad (2\text{-}42)$$

where

$A_{wat} = Area\ fraction\ of\ pool\ coverage\ on\ Northern\ surface\ node\ in\ question [m^2]$

$h_{N,pool} = Heat\ transfer\ coefficient\ for\ pool\ heat\ transfer\ \left[\frac{W}{m^2K}\right]$

$h_{N,atm} = Heat\ transfer\ coefficient\ for\ atmosphere\ heat\ transfer\ \left[\frac{W}{m^2K}\right]$

$T_{N,pool} = Pool\ temperature\ in\ bounding\ control\ volume\ [K]$

$T_{N,atm} = Atmosphere\ temperature\ in\ bounding\ control\ volume\ [K]$

The node boundary temperature can then be solved for directly:

$$T_N^n = \frac{A_{wat}h_{N,pool}T_{N,pool}\left(\frac{dy}{k_n}\right) + \left(1 - A_{wat}\right)h_{N,atm}T_{N,atm}\left(\frac{dy}{k_n}\right) + T_{i,j}^n}{A_{wat}h_{N,pool}\left(\frac{dy}{k_n}\right) + \left(1 - A_{wat}\right)h_{N,atm}\left(\frac{dy}{k_n}\right) + 1} \quad (2\text{-}43)$$

As for the process of modifying conductance coefficients:

- Subtract from $A_P^{i,j}$ the quantity $\frac{A_N^{i,j}}{A_{wat}h_{N,pool}\left(\frac{dy}{k_n}\right) + \left(1 - A_{wat}\right)h_{N,atm}\left(\frac{dy}{k_n}\right) + 1}$

- Add to $b^{i,j}$ the quantity $\frac{A_N^{i,j} A_{wat} h_{N,pool} T_{N,pool} \left(\frac{dy}{k_n}\right) + A_N^{i,j} (1-A_{wat}) h_{N,atm} T_{N,atm} \left(\frac{dy}{k_n}\right)}{A_{wat} h_{N,pool} \left(\frac{dy}{k_n}\right) + (1-A_{wat}) h_{N,atm} \left(\frac{dy}{k_n}\right) + 1}$
- Set $A_N^{i,j}$ to zero

Boundary conditions at east and west borders of the grid are always adiabatic. Heat transfer coefficient boundary conditions or compound heat transfer coefficient boundary conditions may exist at both north and south borders. Heat flux boundary conditions (for debris heat transfer) may be set at the north border if it is determined that debris sits atop a given northern boundary node. Note that heat transfer coefficients are computed for each surface node depending on the current scenario:

- Boiling heat transfer correlations (HS pool or inverted surface boiling routines) if node is water-covered and node temperature exceeds saturation temperature of the water,
- Convection correlations (HS routine) if node is water-covered but not superheated with respect to the water,
- Radiation to the atmosphere if a node is bare

Also note that the heat flux from debris to plate is computed during the debris mass/energy balance equation solutions in LHCMELT.

2.6.6.5 Boiling Heat Transfer Coefficient Relaxation

Since the boiling heat transfer coefficients at the surface of LHC plate boundary nodes are highly sensitive functions of surface temperature, an iteration-to-iteration relaxation strategy is employed as necessary.

2.6.7 Solution and Convergence

The solution to all fully-implicit nodal equations can be obtained with a sweeping strategy (e.g. South to North and West to East). When all nodal equations are solved after a full grid sweep, residuals (measures of energy imbalance in each finite volume) can be computed and convergence checked.

To judge convergence, a residual (the amount by which the nodal energy equation is dissatisfied) is computed for every volume. Those residuals are summed up over the whole domain to get a global residual, and that quantity is compared to the initial (i.e. first-iterate) global residual to compute a global relative residual. When either the global relative residual is low enough or the absolute residual (units of W) is low enough (globally and within each finite volume), convergence is declared. Note that if the calculation doesn't change considerably from time-step to time-step (temperatures and boundary conditions are not altered appreciably e.g. due to a small time-step) it is possible for global

relative residuals to appear high (on the order of one) because the old-time solution and hence the first-iterate solution very nearly satisfies the new-time energy balance. This in part explains the need for multiple convergence criteria. Residual values have implications for the plate energy error. High residuals reflect too large of an energy imbalance and would therefore lead to an unacceptable error in the plate temperature solution. A further convergence criterion is set on plate energy error within a time-step.

Residual equations for each node are gotten from the nodal equation for that node by subtracting the right-hand side terms from the left hand side terms. This indicates how unsatisfied the nodal balance equation is and, ideally, should equal some very small number indiciative of good energy conservation.

2.7 LHC/CVH Energy Transfers

The debris and the LHC plate exchange energy with CVH materials. The debris cylinder top/side exchanges energy with pool or atmosphere in CVolT (the LHC CV above the plate) as does the top surface of the LHC plate. The bottom surface of the LHC plate exchanges energy with the pool or atmosphere in CVolB (the LHC CV below the plate). The LHC plate is “awake” regardless of whether the LHC package is considered “awake” or “asleep”. Thus the LHC plate essentially behaves as a heat structure before debris relocates from COR to LHC. This requires the LHC plate conduction solution to be performed for all problem times before and after COR ejection.

In every pass through LHC execution, while debris and plate heat transfers are computed, counters track the total energy transfer to CVolT (and to which phase) and CVolB (and to which phase). These counters are interrogated and applied as energy sources to the appropriate phases in the appropriate control volumes. This strategy applies whether the LHC is sleeping or not, though the debris cannot contribute to the energy source if the LHC is asleep. Energy transfers to CVolT and CVolB are:

$$xqtop = \Delta t * \left(Q_{top}A_{deb,top} + Q_{side}A_{deb,side}\right) + \Delta t * \left(\sum_{i=1}^{nnr}\left(A_Z^{i,top}h_{N,i}\left(T_i - T_{bndN,i}\right)\right)\right)_{IA=nnz} \quad (2\text{-}44)$$

$$xqbot = \Delta t * \left(\sum_{i=1}^{nnr}\left(A_Z^{i,bottom}h_{S,i}\left(T_i - T_{bndS,i}\right)\right)\right)_{IA=1} \quad (2\text{-}45)$$

where

$A_Z^{i,j}$ = node i,j axial area $\left[\text{m}^2\right]$;

$A_{deb,top}$ = Debris cylinder top area $\left[\text{m}^2\right]$

$A_{deb,side}$ = Debris cylinder side area $\left[\text{m}^2\right]$;

$Qtop$ = Debris top heat flux $\left[\frac{\text{W}}{\text{m}^2}\right]$

Q_{side} = Debris side heat flux $\left[\frac{\text{W}}{\text{m}^2}\right]$;

hN,i = North bnd heat transfer coeff. $\left[\frac{\text{W}}{\text{m}^2\text{K}}\right]$

$h_{S,i}$ = South bnd heat transfer coeff. $\left[\frac{\text{W}}{\text{m}^2\text{K}}\right]$;

Ti = ith node Temperature [K]

nnr = Number of radial/transverse nodes ;

nnz = Number of axial/through-wall nodes

Note that the sums appearing in the xqtop and xqbot counters occur over all radial nodes 1…nnr at the top surface (IA=nnz) and over all radial nodes 1…nnr at the bottom surface (IA=1). If a top surface node is partially covered by debris, the node area is multiplied by one minus its covered area fraction. There is also accounting for cases of partial pool coverage.

2.8 LHC Plate Failure

Currently there are three failure modes allowed for the LHC plate:

1) Over-pressure
2) Over-temperature
3) Larson Miller creep rupture (1-D model)

The LHC failure logic also allows for debris ejection to CAV via TP upon LHC failure. Similar to COR ejection, LHC ejection must account for mass, energy, and radionuclides.

2.9 LHC Debris Ejection to CAV

If at least one failure has occurred and LHC has a nonzero breach radius the melt ejection subroutine LHCMEJ is called. It is similar to but much simpler than the CORMEJ routine. LHCMEJ populates the mass and energy arrays that are communicated to TP as an IN transfer process. The intent is that CAV searches for and retrieves as an OUT transfer process the mass, energy, and radionuclides from LHC. Note that if the requisite transfer processes are not configured for CAV to receive LHC ejection, the calculation terminates.

LHC ejects only the materials given it by COR ejection in the first place: UO2, SS, SSOX, Zr, and ZrO2. Debris mass and energy arrays are segregated by solid/liquid oxide/metal.

Hence there are four parcels comprising the ejection. LHC uses its aforementioned debris solids fraction concept to allocate material mass to the relocation arrays as solid or liquid. The energy of each material is computed from known debris temperatures and does account for latent heat effects. As in COR, the ejection velocity is taken as 1 m/s. Radionuclide relocation is accounted for with separate subroutine calls following the LHCMEJ call but before debris mass and energy transfer arrays are actually sent to TP as an IN transfer process. The radionuclide inventory relocation involves TP as well (an IN transfer process according to user definition).

Material Properties (MP) Package

The MELCOR Material Properties package models many of the properties needed by the various physics packages. This is done by using analytical laws, correlations, and linear tables. New materials and their properties may be defined through user input, and properties for default materials may be redefined by user input.

This document identifies the default material property values and functions used in the MELCOR MP package. References for the data are provided. Detailed descriptions of input requirements are provided in the MP Package Users' Guide.

The thermodynamic properties of water vapor and liquid water are contained in the H_2O package and cannot be modified through user input. Properties of noncondensible gases are calculated by the NonCondensible Gas (NCG) package. A description of the default values and available user input options is provided in the MELCOR NCG Package Users' Guide.

CORCON and VANESA properties are included in the Cavity (CAV) and RadioNuclide (RN) packages, respectively. See the reference manuals and users' guides of those packages.

Contents

List of Tables

1. Default Material Properties

The MELCOR Material Properties (MP) package models many common properties needed by the various phenomenological packages through the use of analytical laws, correlations, and tabulated values. These properties are thermodynamic state and transport properties needed for structural materials. Transport and thermodynamic state properties for water and noncondensible gases are provided by the H2O and NCG packages (see the NCG/H2O Reference Manual).

In a few cases, stand-alone codes that have been wholly integrated into MELCOR still use properties defined within those codes; a notable example is CORCON, which has been integrated into the Cavity (CAV) package. Also, properties unique to a package, such as those for trace species used in the RadioNuclide (RN) package, are generally modeled within that package. The Core (COR), Fuel Dispersal Interactions (FDI), and Heat Structures (HS) packages use principally the structural materials properties, while the Control Volume Hydrodynamics (CVH), Engineered Safety Features (ESF), Containment Sprays (SPR), and RN packages use principally the fluid transport properties.

The following materials, listed with their mnemonic identifiers, are defined in the Material Properties package:

1. Zircaloy (ZR)
2. Zirconium Oxide (ZRO2)
3. Zirconium Oxide (ZRO2-INT)
4. Uranium Dioxide (UO2)
5. Uranium Dioxide (UO2-INT)
6. Stainless Steel (SS)
7. Stainless Steel Oxide (SSOX)
8. Boron Carbide (B4C)
9. Boron Carbide (B4C-INT)
10. Silver-Indium-Cadmium (AGINC)
11. Uranium Metal (UMETL)
12. Graphite (GRAPH)
13. Concrete (CON)
14. Aluminum (ALUM)
15. Aluminum Oxide (AL2O3)
16. Cadmium (CADM)
17. Stainless Steel 304 (SS304)
18. Carbon Steel (CS)
19. Iron-Cromium Aluminum (FCA)
20. Iron-Cromium Aluminum Oxide (FCAO)

Material 6, Stainless Steel (SS), is a type 347 stainless steel and is typically used in the Core package, whereas material 17 (SS304) is a type 304 stainless steel. Materials ZRO2-INT, UO2-INT, and B4C-INT are identical to materials ZRO2, UO2, and B4C, respectively, except for modified melting properties that simulate the reduction in liquefaction temperature that results from materials interactions.

The following properties are defined in the package:

	Type	Units
1. Enthalpy as a function of temperature (ENH**)**	Tabular	J/kg
2. Temperature as a function of enthalpy (TMP**)**	Calc.	K
3. Specific Heat Capacity as a function of temperature (CPS**)**	Tabular	J/kg-K
4. Thermal Conductivity as a function of temperature (THC**)**		
a. From tables, TF, or CF	Tabular	W/m-K
5. Density		
a. Constant (RHOM)	Constant	kg/m^3
b. Function of temperature (RHO)	Tabular	kg/m^3
6. Melting Temperature (TMLT**)**	Constant	K
7. Latent Heat of Fusion (LHF**)**	Constant	J/kg

Default values are provided for some, but not all, combinations of materials and physical properties. Table 2.1 summarizes the default values available. A ‘T’ indicates that the default function can be changed through user-defined tabular functions and an MP_PRTF input record. A ‘C’ indicates that the default function can be changed through user-defined constant values input on an MP_PRC record. An ‘X’ indicates that the default function cannot be changed through user input. A blank space indicates that no default is provided, but may be supplied by the user, although in some cases that property for that material may not be used by MELCOR.

Also shown is the mnemonic(s) used to add new values or alter the default values through user input for those properties that can be changed.

Sections 2 through 8 identify the default values for those combinations defined in MELCOR. User definition of the materials properties is also discussed in each section.

2. Specific Enthalpy as a Function of Temperature

The specific enthalpy may be computed from either a user-specified tabular function or a MELCOR default table.

The user-specified tabular function to define a new material or to override the default table for an existing material is invoked by using a standard tabular function (see the TF Package Users' Guide) to input the enthalpy (J/kg) as a function of temperature (K). Negative enthalpies are permitted. Currently, there are no checks made on the consistency of user-input values for enthalpy, specific heat capacity, melting temperature, and latent heat of fusion; this could be rectified in future code versions.

The following materials have default tables for enthalpy:

Zircaloy	Concrete
Zirconium Oxide	Aluminum
Uranium Dioxide	Aluminum Oxide
Stainless Steel	Cadmium
Stainless Steel Oxide	Stainless Steel 304
Boron Carbide	Carbon Steel
Silver-Indium-Cadmium	Iron-Chromium-Aluminum
Uranium Metal	Iron-Chromium-Aluminum Oxide
Graphite	

Table 2.1 Default material properties, property mnemonics, and user input capabilities.

Property* (Mnemonic)	ENH	TMP	CPS	THC	RHOM	RHO	TMLT	LHF
ZR	T	T	T	T	C	T	C	C
ZRO2	T	T	T	T	C	T	C	C
ZRO2-INT	X		X	T	C	T	C	C
UO2	T	T	T	T	C	T	C	C
UO2-INT	X			T	C	T	C	C
SS	T	T	T	T	C	T	C	C
SSOX	T	T	T	T	C	T	C	C
B4C	T	T	T	T	C	T	C	C
B4C-INT	X		X	T	C	T	C	C
AGINC	T	T	T	T	C	T	C	C
UMETL	T	T	T	T	C	T	C	C
GRAPH	T	T	T	T	C	T	C	
CON			T	T		T		
ALUM	T	T	T	T	C	T	C	C
AL2O3	T	T	T	T	C	T	C	C
CADM	T	T	T	T	C	T	C	C
SS304	T	T	T	T	C	T	C	C
CS	T	T	T	T	C	T	C	C
FCA	T	T	T	T	C	T	C	C
FCAO	T	T	T	T	C	T	C	C

T - The default function can be changed using tabular functions and an MP_PRTF input record.

C - The default function can be changed using constant values input on an MP_PRC input record.

X - The default function cannot be changed through user input.

Note: A blank space indicates that no default is provided, but may be supplied by the user, although in some cases the property may not be used.

* See Section 1 for a full description of these properties.

The default specific enthalpy values are computed by linear interpolation of the tabulated values listed below. The tabular values were computed by integrating the tables of specific heat capacities from Section 4 . The latent heat of fusion from Section 8 was added at the melting point given in Section 7 over a range of 0.01 K.

2.1 Zircaloy

The default tabular values of specific enthalpy as a function of temperature for Zircaloy are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Zircaloy

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
400.0	21915.0
640.0	105110.0
1090.0	263960.0
1093.0	265275.5
1113.0	276195.5
1133.0	288245.5
1153.0	301585.5
1173.0	316935.5
1193.0	332795.5
1213.0	346685.5
1233.0	357565.5
1248.0	363753.0
2098.0	666353.0
2098.01	891353.0
3598.0	1425353.0

2.2 Zirconium Oxide

The default tabular values of specific enthalpy as a function of temperature for zirconium oxide are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Zirconium Oxide

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
2990.0	1464167.0
2990.01	2171167.0
3500.0	2448760.0

2.3 Uranium Dioxide

The default tabular values of specific enthalpy as a function of temperature for uranium dioxide are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Uranium Dioxide

Temperature (K)	Specific Enthalpy (J/kg)
300.0	33143.0
400.0	58419.0
500.0	85883.0
600.0	114638.0
700.0	144257.0
800.0	174517.0
900.0	205288.0
1000.0	236492.0
1100.0	268080.0
1200.0	300023.0
1300.0	332309.0
1400.0	364947.0
1500.0	397973.0
1600.0	431455.0
1700.0	465502.0
1800.0	500266.0
1900.0	535945.0
2000.0	572782.0
2100.0	611064.0
2200.0	651111.0
2300.0	693275.0
2400.0	737927.0
2500.0	785450.0
2600.0	836232.0
2700.0	890656.0
2800.0	949096.0
2900.0	1011906.0
3000.0	1079422.0
3113.0	1161764.0
3113.01	1435764.0
3513.0	1636964.0

2.4 Stainless Steel

The default tabular values of specific enthalpy as a function of temperature for stainless steel are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Stainless Steel

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
400.0	48926.0
500.0	99624.0
600.0	152092.0
700.0	206332.0
800.0	262343.0
900.0	320125.0
1000.0	379679.0
1100.0	441003.0
1200.0	504099.0
1300.0	568966.0
1400.0	635604.0
1500.0	704014.0
1600.0	774194.0
1700.0	846146.0
1700.01	1114146.0
1800.0	1186986.0
3800.0	2643786.0

2.5 Stainless Steel Oxide

The default tabular values of specific enthalpy as a function of temperature for stainless steel oxide are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Stainless Steel Oxide

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
1870.0	785000.0
1870.01	1383000.0
3500.0	2198000.0

2.6 Boron Carbide

The default tabular values of specific enthalpy as a function of temperature for boron carbide are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Boron Carbide

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
2620.0	1160000.0
2620.01	1660000.0
3500.0	2100000.0

2.7 Silver-Indium-Cadmium

The default tabular values of specific enthalpy as a function of temperature for silver-indium-cadmium are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Silver-Indium-Cadmium

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
400.0	21759.0
500.0	44031.0
600.0	66801.0
700.0	90091.0
800.0	113890.0
900.0	138200.0
1000.0	163010.0
1075.0	211000.0
1075.01	309000.0
1100.0	315350.0
5000.0	1306600.0

2.8 Uranium Metal

The default tabular values of specific enthalpy as a function of temperature for uranium metal are listed below. Linear extrapolation is allowed from the lower end of the tabulated range. No extrapolation is allowed from the upper end of the tabulated range.

Uranium Metal

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
400.0	12050.0
600.0	39150.0
800.0	71350.0
1000.0	106950.0
1200.0	141050.0
1406.0	172259.0
1406.01	222499.0
5000.0	732847.0

2.9 Graphite

The default tabular values of specific enthalpy as a function of temperature for graphite are listed below. Linear extrapolation is allowed from the lower end of the tabulated range. No extrapolation is allowed from the upper end of the tabulated range.

Graphite

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
773.0	547910.0
1273.0	1414010.0
1773.0	2381110.0
2273.0	3405060.0
2773.0	4464560.0
3866.0	6879871.0
5000.0	9456545.0

2.10 Aluminum

The default tabular values of specific enthalpy as a function of temperature for aluminum are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Aluminum

Temperature (K)	Specific Enthalpy (J/kg)
273.15	0.00
313.15	36056.00
353.15	72822.00
393.15	110304.00
433.15	148506.00
473.15	187432.00
513.15	227088.00
553.15	267464.00
593.15	308580.00
633.15	350458.00
673.15	393086.00
713.15	436470.00
753.15	480616.00
793.15	525528.00
833.15	571210.00
873.15	617668.00
913.15	664908.00
933.00	688643.00
933.01	1086443.00
1000.00	1165269.00
1500.00	1753519.00
2000.00	2341769.00

2.11 Aluminum Oxide

The default tabular values of specific enthalpy as a function of temperature for aluminum oxide are listed below. Linear extrapolation is allowed from both ends of the tabular range.

Aluminum Oxide

Temperature (K)	Specific Enthalpy (J/kg)
273.15	0.0
298.0	19243.0
350.0	62146.0
400.0	107619.0
500.0	206437.0
600.0	312785.0

800.0	540165.0
1000.0	780637.0
1500.0	1410855.0
2327.0	2518696.0
2327.01	3588710.0
5000.0	7386414.0

2.12 Cadmium

The default tabular values of specific enthalpy as a function of temperature for cadmium are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Cadmium

Temperature (K)	Specific Enthalpy (J/kg)
298.15	0.00
400.00	24093.00
500.00	48813.00
594.00	73079.00
594.01	128347.00
600.00	129933.00
700.00	156373.00
800.00	182813.00
900.00	209253.00
1000.00	235693.00
1040.00	246269.00

2.13 Stainless Steel 304

The default tabular values of specific enthalpy as a function of temperature for stainless steel 304 are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Stainless Steel 304

Temperature (K)	Specific Enthalpy (J/kg)
300.00	0.00
400.00	52005.00
500.00	105370.00
600.00	160085.00

Stainless Steel 304

Temperature (K)	Specific Enthalpy (J/kg)
700.00	216155.00
800.00	273585.00
900.00	332375.00
1000.00	392520.00
1100.00	454020.00
1200.00	516880.00
1300.00	581095.00
1400.00	646665.00
1500.00	713630.00
1600.00	781950.00
1700.00	851590.00
1700.01	1120790.00
1800.00	1200800.00
1900.00	1280810.00
2000.00	1360820.00
2500.00	1760870.00
3000.00	2160920.00

2.14 Carbon Steel

The default tabular values of specific enthalpy as a function of temperature for carbon steel are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Carbon Steel

Temperature (K)	Specific Enthalpy (J/kg)
273.15	0.0
373.15	45667.0
473.15	95490.8
573.15	149471.4
673.15	207608.3
773.15	271000.8
873.15	341966.8
923.15	381218.1
973.15	424656.3
1023.15	475944.8
1033.15	488295.9

Carbon Steel

Temperature (K)	Specific Enthalpy (J/kg)
1073.15	531838.7
1123.15	571090.0
1223.15	642265.5
1349.82	729771.6
1373.15	745920.6
1473.15	815868.4
1573.15	886996.6
1673.15	959305.2
1773.15	1032794.1
1810.90	1060843.1
1810.91	1332803.1
5000.00	3709472.4

2.15 Iron-Chromium-Aluminum

The default tabular values of specific enthalpy as a function of temperature for iron-chromium-aluminum are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Iron-Chromium-Aluminum

Temperature (K)	Specific Enthalpy (J/kg)
300.00	0.0
400.00	48580.6
500.00	102685.0
600.00	161742.6
700.00	226972.0
800.00	301381.0
852.00	345028.0
873.00	362757.6
900.00	384088.0
973.00	438865.4
1173.00	582349.5
1273.00	653237.0

Iron-Chromium-Aluminum

Temperature (K)	Specific Enthalpy (J/kg)
1373.00	725741.6
1473.00	801517.1
1573.00	882524.3
1673.00	971064.7
1773.00	1069797.7
1773.01	1339797.7
5000.00	4713282.7

2.16 Iron-Chromium-Aluminum Oxide

The default tabular values of specific enthalpy as a function of temperature for iron-chormium-aluminum oxide are listed below. Linear extrapolation is allowed from both ends of the tabulated range.

Iron-Chromium-Aluminum Oxide

Temperature (K)	Specific Enthalpy (J/kg)
300.0	0.0
1901.0	1440900.0
1901.01	2128363.0
3500.0	3567454.0

3. Temperature as a Function of Special Enthalpy

The temperature as a function of specific enthalpy may be computed from either a user-specified tabular function or a MELCOR default table.

The user-specified tabular function to define a new material or to override the default table for an existing material is invoked by using a standard tabular function (see the TF Package Users' Guide) to input the temperature (K) as a function of enthalpy (J/kg). Currently, there are no checks made on the consistency of user-input values for enthalpy, specific heat capacity, melting temperature, and latent heat of fusion; this could be rectified in future code versions.

The following materials have default tables for temperature as a function of enthalpy:

Zircaloy Boron Carbide Aluminum Oxide

Zirconium Oxide	Silver-Indium-Cadmium	Cadmium
Uranium Dioxide	Uranium Metal	Stainless Steel 304
Stainless Steel	Graphite	Carbon Steel
Stainless Steel Oxide	Aluminum	

The default specific enthalpy values are calculated by linear interpolation of tabulated values computed by inverting the tables of specific enthalpy as a function of temperature from Section 2. Extrapolation rules are the same as those listed in Section 2.

4. Specific Heat Capacity as a Function of Temperature

The specific heat capacity at constant pressure may be computed from either a user-specified tabular function or a MELCOR default table defined in subroutine MPDFVL.

The user-specified tabular function to define a new material or to override the default table for an existing material is invoked by using a standard tabular function (see the TF Package Users' Guide) to input the specific heat capacity (J/kg-K) as a function of temperature (K). There are no checks made on the consistency of user-input values for enthalpy, specific heat capacity, melting temperature, and latent heat of fusion.

The following materials have default tables for specific heat capacity:

Zircaloy	Concrete
Zirconium Oxide	Aluminum
Uranium Dioxide	Aluminum Oxide
Stainless Steel	Cadmium
Stainless Steel Oxide	Stainless Steel 304
Boron Carbide	Carbon Steel
Silver-Indium-Cadmium	Iron-Chromium-Aluminum
Uranium Metal	Iron-Chromium-Aluminum Oxide
Graphite	

The default specific heat capacity values are computed by linear interpolation of the tabulated values listed below. Data sources are given with each table.

4.1 Zircaloy

The default tabular values of specific heat capacity as a function of temperature for Zircaloy are listed below. No extrapolation is allowed.

Zircaloy

Temp(K)	Specific Heat Capacity (J/kg-K)	Data Source
273.1	275.0	Ref. [1], extrapolated
400.0	302.0	Ref. [1]
640.0	331.0	Ref. [1]
1090.0	375.0	Ref. [1]
1093.0	502.0	Ref. [1]
1113.0	590.0	Ref. [1]
1133.0	615.0	Ref. [1]
1153.0	719.0	Ref. [1]
1173.0	816.0	Ref. [1]
1193.0	770.0	Ref. [1]
1213.0	619.0	Ref. [1]
1233.0	469.0	Ref. [1]
1248.0	356.0	Ref. [1]
2098.0	356.0	Ref. [1]
5000.0	356.0	Ref. [1], extrapolated

4.2 Zirconium Oxide

The default tabular values of specific heat capacity as a function of temperature for zirconium oxide are listed below. No extrapolation is allowed.

Zirconium Oxide

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	544.3	Ref. [1]
5000.0	544.3	Ref. [1]

4.3 Uranium Dioxide

The default tabular values of specific heat capacity as a function of temperature for uranium dioxide are listed below. No extrapolation is allowed.

Uranium Dioxide

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	230.22	Ref. [1], extrapolated
400.0	265.84	Ref. [1]
500.0	282.07	Ref. [1]
600.0	292.36	Ref. [1]
700.0	299.67	Ref. [1]
800.0	305.31	Ref. [1]
900.0	309.98	Ref. [1]
1000.0	314.03	Ref. [1]
1100.0	317.69	Ref. [1]
1200.0	321.15	Ref. [1]
1300.0	324.59	Ref. [1]
1400.0	328.24	Ref. [1]
1500.0	332.40	Ref. [1]
1600.0	337.43	Ref. [1]
1700.0	343.76	Ref. [1]
1800.0	351.84	Ref. [1]
1900.0	362.14	Ref. [1]
2000.0	375.09	Ref. [1]
2100.0	391.08	Ref. [1]
2200.0	410.45	Ref. [1]
2300.0	433.45	Ref. [1]
2400.0	460.23	Ref. [1]
2500.0	490.88	Ref. [1]
2600.0	525.40	Ref. [1]
2700.0	563.71	Ref. [1]
2800.0	605.67	Ref. [1]
2900.0	651.09	Ref. [1]
3000.0	699.73	Ref. [1]
3113.0	758.23	Ref. [1]
3113.01	503.0	Ref. [1]
5000.0	503.0	Ref. [1], extrapolated

4.4 Stainless Steel

The default tabular values of specific heat capacity as a function of temperature for 347 stainless steel are listed below. No extrapolation is allowed.

Stainless Steel

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	475.6	Ref. [1], extrapolated
400.0	498.1	Ref. [1]
500.0	515.8	Ref. [1]
600.0	533.5	Ref. [1]
700.0	551.3	Ref. [1]
800.0	569.0	Ref. [1]
900.0	586.7	Ref. [1]
1000.0	604.4	Ref. [1]
1100.0	622.1	Ref. [1]
1200.0	639.8	Ref. [1]
1300.0	657.5	Ref. [1]
1400.0	675.2	Ref. [1]
1500.0	693.0	Ref. [1]
1600.0	710.7	Ref. [1]
1700.0	728.4	Ref. [1]
1700.01	728.4	Ref. [1]
1800.0	728.4	Ref. [1]
5000.0	728.4	Ref. [1], extrapolated

4.5 Stainless Steel Oxide

The default tabular values of specific heat capacity as a function of temperature for stainless steel oxide are listed below. No extrapolation is allowed.

Stainless Steel Oxide

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	500.0	Estimated
5000.0	500.0	Estimated

4.6 Boron Carbide

The default tabular values of specific heat capacity as a function of temperature for boron carbide are listed below. No extrapolation is allowed.

Boron Carbide

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	500.0	Estimated

5000.0	500.0	Estimated

4.7 Silver-Indium-Cadmium

The default tabular values of specific heat capacity as a function of temperature for silver-indium-cadmium are listed below. Linear extrapolation below 300 K is permitted.

Silver-Indium-Cadmium

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
300.0	215.04	Ref. [2]
400.0	220.14	Ref. [2]
500.0	225.23	Ref. [2]
600.0	230.33	Ref. [2]
700.0	235.42	Ref. [2]
800.0	240.52	Ref. [2]
900.0	245.61	Ref. [2]
1000.0	250.71	Ref. [2]
1075.0	254.15	Ref. [2]
5000.0	254.15	Ref. [2]

4.8 Uranium Metal

The default tabular values of specific heat capacity as a function of temperature for uranium metal are listed below. No extrapolation is allowed.

Uranium Metal

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	113.6	Ref. [3], p. 758, extrapolated
300.0	116.0	Ref. [3], p. 758
400.0	125.0	Ref. [3], p. 758
600.0	146.0	Ref. [3], p. 758
800.0	176.0	Ref. [3], p. 758
1000.0	180.0	Ref. [3], p. 758
1200.0	161.0	Ref. [3], p. 758
1406.0	142.0	Ref. [3], p. 758, extrapolated
5000.0	142.0	Constant from melting point

4.9 Graphite

The default tabular values of specific heat capacity as a function of temperature for graphite are listed below. No extrapolation is allowed.

Graphite

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	665.16	Ref. [4], p. 180, generic graphite, extrapolated
298.0	711.7	Ref. [4], p. 180, generic graphite
773.0	1601.3	Ref. [4], p. 180, generic graphite
1273.0	1863.0	Ref. [4], p. 180, generic graphite
1773.0	2005.3	Ref. [4], p. 180, generic graphite
2273.0	2090.6	Ref. [4], p. 180, generic graphite
2773.0	2147.5	Ref. [4], p. 180, generic graphite
3866.0	2272.0	Ref. [4], p. 180, generic graphite, extrapolated
5000.0	2272.0	Constant from melting point of 3866 K

4.10 Concrete

The default tabular values of specific heat capacity as a function of temperature for concrete are listed below. No extrapolation is allowed.

Concrete

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	837.3	Ref. [5], p. 635, stone concrete at 294 K
5000.0	837.3	Ref. [5], p. 635, stone concrete at 294 K

4.11 Aluminum

The default tabular values of specific heat capacity as a function of temperature for aluminum are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Aluminum

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	892.60	Ref. [6]

Aluminum

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
313.15	910.20	Ref. [6]
353.15	928.10	Ref. [6]
393.15	946.00	Ref. [6]
433.15	964.10	Ref. [6]
473.15	982.20	Ref. [6]
513.15	1000.60	Ref. [6]
553.15	1018.20	Ref. [6]
593.15	1037.60	Ref. [6]
633.15	1056.30	Ref. [6]
673.15	1075.10	Ref. [6]
713.15	1094.10	Ref. [6]
753.15	1113.20	Ref. [6]
793.15	1132.40	Ref. [6]
833.15	1151.70	Ref. [6]
873.15	1171.20	Ref. [6]
913.15	1190.80	Ref. [6]
933.00	1200.60	Ref. [6]
933.01	1176.50	Ref. [6]
1000.00	1176.50	Ref. [6]
1500.00	1176.50	Ref. [6]
2000.00	1176.50	Ref. [6]

4.12 Aluminum Oxide

The default tabular values of specific heat capacity as a function of temperature for aluminum oxide are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Aluminum Oxide

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	774.38	Ref. [7]
298.0	774.38	Ref. [7]
350.0	875.73	Ref. [7]
400.0	943.20	Ref. [7]
500.0	1033.16	Ref. [7]

600.0	1093.80	Ref. [7]
800.0	1180.00	Ref. [7]
1000.0	1224.72	Ref. [7]
1500.0	1296.15	Ref. [7]
2327.0	1383.03	Ref. [7]
2327.01	1420.77	Ref. [7]
5000.0	1420.77	Ref. [7]

4.13 Cadmium

The default tabular values of specific heat capacity as a function of temperature for cadmium are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Cadmium

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
298.15	231.30	Ref. [6]
400.00	241.80	Ref. [6]
500.00	252.60	Ref. [6]
594.00	263.70	Ref. [6]
594.01	264.40	Ref. [6]
600.00	264.40	Ref. [6]
1040.00	264.40	Ref. [6]

4.14 Stainless Steel 304

The default tabular values of specific heat capacity as a function of temperature for stainless steel 304 are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Stainless Steel 304

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
300.00	513.20	Ref. [6]
400.00	526.90	Ref. [6]
500.00	540.40	Ref. [6]
600.00	553.90	Ref. [6]
700.00	567.50	Ref. [6]
800.00	581.10	Ref. [6]

900.00	594.70	Ref. [6]
1000.00	608.20	Ref. [6]
1100.00	621.80	Ref. [6]
1200.00	635.40	Ref. [6]
1300.00	648.90	Ref. [6]
1400.00	662.50	Ref. [6]
1500.00	676.80	Ref. [6]
1600.00	689.60	Ref. [6]
1700.00	703.20	Ref. [6]
1700.01	800.10	Ref. [6]
1800.00	800.10	Ref. [6]
3000.00	800.10	Ref. [6]

4.15 Carbon Steel

The default tabular values of specific heat capacity as a function of temperature for carbon steel are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Carbon Steel

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	435.89	Ref. [8]
373.15	477.45	Ref. [8]
473.15	519.02	Ref. [8]
573.15	560.59	Ref. [8]
673.15	602.15	Ref. [8]
773.15	665.70	Ref. [8]
873.15	753.62	Ref. [8]
923.15	816.43	Ref. [8]
973.15	921.10	Ref. [8]
1023.15	1130.44	Ref. [8]
1033.15	1339.78	Ref. [8]
1073.15	837.36	Ref. [8]
1123.15	732.69	Ref. [8]
1223.15	690.82	Ref. [8]
1349.82	690.82	Ref. [8]
1373.15	693.58	Ref. [8]
1473.15	705.38	Ref. [8]

Carbon Steel

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
1573.15	717.18	Ref. [8]
1673.15	728.99	Ref. [8]
1773.15	740.79	Ref. [8]
1810.90	745.25	Ref. [8]
1810.91	745.25	Ref. [8]
5000.00	745.25	Ref. [8]

4.16 Iron-Chromium-Aluminum

The default tabular values of specific heat capacity as a function of temperature for iron-chromium-aluminum are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Iron-Chromium-Aluminum

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
300.0	454.52	Ref. [9]
400.0	517.08	Ref. [9]
500.0	565.00	Ref. [9]
600.0	616.15	Ref. [9]
700.0	688.43	Ref. [9]
800.0	799.74	Ref. [9]
852.0	878.98	Ref. [9]
873.0	809.54	Ref. [9]
900.0	770.48	Ref. [9]
973.0	730.27	Ref. [9]
1173.0	704.57	Ref. [9]
1273.0	713.17	Ref. [9]
1373.0	736.91	Ref. [9]
1473.0	778.59	Ref. [9]
1573.0	841.54	Ref. [9]
1673.0	929.26	Ref. [9]
1773.0	1045.39	Ref. [9]
1773.01	1045.39	Ref. [9]

Iron-Chromium-Aluminum

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
5000.0	1045.39	Ref. [9]

4.17 Iron-Chromium-Aluminum Oxide

The default tabular values of specific heat capacity as a function of temperature for iron-chromium-aluminum oxide are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Iron-Chromium-Aluminum Oxide

Temp (K)	Specific Heat Capacity (J/kg-K)	Data Source
273.15	900.00	Estimated
5000.0	900.00	Estimated

5. Thermal Conductivity as a Function of Temperature

The thermal conductivity may be computed from two different methods. One method, used for structural materials in the COR and HS packages, uses tabular data that may be either a user-specified tabular function or a MELCOR default table. The other method, used for noncondensible gases and optionally for steam and air, utilizes the Eucken correlation for single, low-pressure gases and the Wassijewa equation for a combination of gases.

5.1 Tabular

The user-specified tabular function to define a new material or to override the default table for an existing material is invoked by using a standard tabular function to input the thermal conductivity (W/m-K) as a function of temperature (K).

The following materials have default tables for thermal conductivity:

Zircaloy	Concrete
Zirconium Oxide	Aluminum
Uranium Dioxide	Aluminum Oxide
Stainless Steel	Cadmium

Stainless Steel Oxide	Stainless Steel 304
Boron Carbide	Carbon Steel
Silver-Indium-Cadmium	Iron-Chromium-Aluminum
Uranium Metal	Iron-Chromium-Aluminum Oxide
Graphite	

The default thermal conductivity values are computed by linear interpolation of the tabulated values listed below. Data sources are given with each table.

5.1.1 Zircaloy

The default tabular values of thermal conductivity as a function of temperature for Zircaloy are listed below. No extrapolation is allowed.

Zircaloy

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	12.1	Ref. [1], p. 218, formula
293.2	12.6	Ref. [1], p. 221, Zircaloy-2
473.2	14.5	Ref. [1], p. 221, Zircaloy-2
673.2	17.0	Ref. [1], p. 221, Zircaloy-2
873.2	19.9	Ref. [1], p. 221, Zircaloy-2
1073.2	23.1	Ref. [1], p. 221, Zircaloy-2
1269.2	26.2	Ref. [1], p. 219, Zircaloy-4
1508.2	31.7	Ref. [1], p. 219, Zircaloy-4
1624.2	36.3	Ref. [1], p. 219, Zircaloy-4
1771.2	41.8	Ref. [1], p. 219, Zircaloy-4
2098.2	58.4	Ref. [1], p. 218, formula
5000.0	58.4	Constant beyond melting point of 2098 K

5.1.2 Zirconium Oxide

The default tabular values of thermal conductivity as a function of temperature for zirconium oxide are listed below. No extrapolation is allowed.

Zirconium Oxide

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	1.94	Ref. [1], p. 224, formula
500.0	1.98	Ref. [1], p. 224, formula
750.0	2.06	Ref. [1], p. 224, formula
1000.0	2.17	Ref. [1], p. 224, formula
1250.0	2.28	Ref. [1], p. 224, formula
1500.0	2.39	Ref. [1], p. 224, formula
2000.0	2.49	Ref. [1], p. 224, formula
5000.0	2.49	Constant beyond 2000 K

5.1.3 Uranium Dioxide

The default tabular values of thermal conductivity as a function of temperature for uranium dioxide are listed below. No extrapolation is allowed.

Uranium Dioxide

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	9.24	Ref. [10], p. 104,
366.3	7.79	Ref. [10], p. 104
539.0	6.53	Ref. [1], p. 30
757.0	4.92	Ref. [1], p. 30
995.0	3.87	Ref. [1], p. 30
1182.0	3.20	Ref. [1], p. 30
1490.0	2.53	Ref. [1], p. 30
1779.0	2.19	Ref. [1], p. 30
1975.0	2.17	Ref. [1], p. 30
2181.0	2.25	Ref. [1], p. 30
2373.0	2.56	Ref. [1], p. 30
2577.0	2.80	Ref. [1], p. 35
2773.0	3.15	Ref. [1], p. 35
3026.0	3.75	Ref. [1], p. 35
3113.0	3.96	Ref. [1], p. 35, extrapolated
5000.0	3.96	Constant beyond melting point of 3113 K

5.1.4 Stainless Steel (SS)

The default tabular values of thermal conductivity as a function of temperature for stainless steel (SS), type 347, are listed below. No extrapolation is allowed.

Stainless Steel (SS)

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	13.8	Ref. [3], p. 757, extrapolated
400.0	15.8	Ref. [3], p. 757
600.0	18.9	Ref. [3], p. 757
800.0	21.9	Ref. [3], p. 757
1000.0	24.7	Ref. [3], p. 757
1700.0	34.5	Ref. [3], p. 757, extrapolated
5000.0	34.5	Constant beyond melting point of 1700 K

5.1.5 Stainless Steel Oxide

The default tabular values of thermal conductivity as a function of temperature for stainless steel oxide are listed below.

Stainless Steel Oxide

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	20.0	Estimated
5000.0	20.0	Estimated

5.1.6 Boron Carbide

The default tabular values of thermal conductivity as a function of temperature for boron carbide are listed below. No extrapolation is allowed.

Boron Carbide

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	2.0	Estimated
5000.0	2.0	Estimated

5.1.7 Silver-Indium-Cadmium

The default tabular values of thermal conductivity as a function of temperature for silver-indium-cadmium are listed below. No extrapolation is allowed.

Silver-Indium-Cadmium

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
300.0	57.088	Ref. [2]
400.0	64.992	Ref. [2]
500.0	72.010	Ref. [2]
600.0	78.140	Ref. [2]
700.0	83.384	Ref. [2]
800.0	87.740	Ref. [2]
900.0	91.208	Ref. [2]
1000.0	93.790	Ref. [2]
1050.0	94.748	Ref. [2]
1075.0	48.000	Ref. [2]
5000.0	48.000	Ref. [2]

5.1.8 Uranium Metal

The default tabular values of thermal conductivity as a function of temperature for uranium metal are listed below. No extrapolation is allowed.

Uranium Metal

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	24.31	Ref. [10], p. 104, extrapolated
298.0	25.12	Ref. [10], p. 104
366.3	27.34	Ref. [10], p. 104
421.9	28.38	Ref. [10], p. 104
477.4	29.34	Ref. [10], p. 104
533.0	30.28	Ref. [10], p. 104
588.6	31.32	Ref. [10], p. 104
644.1	32.22	Ref. [10], p. 104
699.7	33.22	Ref. [10], p. 104
755.2	34.09	Ref. [10], p. 104
810.8	35.04	Ref. [10], p. 104
866.3	35.90	Ref. [10], p. 104

Uranium Metal

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
921.9	36.68	Ref. [10], p. 104
977.4	37.37	Ref. [10], p. 104
1033.0	38.07	Ref. [10], p. 104
1406.0	42.77	Ref. [10], p. 104, extrapolated
5000.0	42.77	Constant beyond melting point of 1406 K

5.1.9 Graphite

The default tabular values of thermal conductivity as a function of temperature for graphite are listed below. No extrapolation is allowed.

Graphite

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	35.55	Ref. [11], irradiated graphite
5000.0	35.55	Ref. [11], irradiated graphite

5.1.10 Concrete

The default tabular values of thermal conductivity as a function of temperature for concrete are listed below. No extrapolation is allowed.

Concrete

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	0.9344	Ref. [5], p. 635, stone concrete @ 294 K
5000.0	0.9344	Ref. [5], p. 635, stone concrete @ 294 K

5.1.11 Aluminum

The default tabular values of thermal conductivity as a function of temperature for aluminum are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Aluminum

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	236.00	Ref. [6]
300.00	237.00	Ref. [6]
350.00	240.00	Ref. [6]
400.00	240.00	Ref. [6]
500.00	237.00	Ref. [6]
600.00	232.00	Ref. [6]
700.00	226.00	Ref. [6]
800.00	220.00	Ref. [6]
900.00	213.00	Ref. [6]
933.00	211.00	Ref. [6]
933.01	90.70	Ref. [6]
1000.00	93.00	Ref. [6]
1100.00	96.40	Ref. [6]
1200.00	99.40	Ref. [6]
1300.00	102.00	Ref. [6]

5.1.12 Aluminum Oxide

The default tabular values of thermal conductivity as a function of temperature for aluminum oxide are listed below. Constant extrapolation is allowed from both ends of the tabular range.

Aluminum Oxide

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	18.73	Ref. [7]
300.0	17.27	Ref. [7]
350.0	15.12	Ref. [7]
400.0	13.47	Ref. [7]
500.0	11.11	Ref. [7]
600.0	9.49	Ref. [7]
700.0	8.31	Ref. [7]
800.0	7.41	Ref. [7]
900.0	6.69	Ref. [7]
1000.0	6.11	Ref. [7]
1200.0	5.22	Ref. [7]
1400.0	4.57	Ref. [7]

Aluminum Oxide

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
1600.0	4.07	Ref. [7]
1800.0	3.68	Ref. [7]
2000.0	3.36	Ref. [7]
2400.0	2.87	Ref. [7]
2800.0	2.51	Ref. [7]
3400.0	2.12	Ref. [7]
4200.0	1.77	Ref. [7]
5000.0	1.42	Ref. [7]

5.1.13 Cadmium

The default tabular values of thermal conductivity as a function of temperature for cadmium are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Cadmium

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	97.50	Ref. [6]
283.15	97.30	Ref. [6]
293.15	97.00	Ref. [6]
303.15	96.80	Ref. [6]
313.15	96.60	Ref. [6]
323.15	96.40	Ref. [6]
333.15	96.20	Ref. [6]
343.15	96.00	Ref. [6]
353.15	95.70	Ref. [6]
363.15	95.50	Ref. [6]
373.15	95.30	Ref. [6]
383.15	95.10	Ref. [6]
393.15	94.90	Ref. [6]
403.15	94.70	Ref. [6]
413.15	94.40	Ref. [6]
423.15	94.20	Ref. [6]
433.15	94.00	Ref. [6]
443.15	93.70	Ref. [6]
453.15	93.50	Ref. [6]
463.15	93.20	Ref. [6]
473.15	92.90	Ref. [6]

Cadmium

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
483.15	92.60	Ref. [6]
493.15	92.30	Ref. [6]
503.15	91.90	Ref. [6]
513.15	91.60	Ref. [6]
523.15	91.20	Ref. [6]
533.15	90.80	Ref. [6]
543.15	90.40	Ref. [6]
553.15	89.90	Ref. [6]
563.15	89.40	Ref. [6]
573.15	88.90	Ref. [6]
583.15	88.40	Ref. [6]
594.00	87.90	Ref. [6]
594.01	41.60	Ref. [6]
600.00	42.00	Ref. [6]
700.00	49.00	Ref. [6]
800.00	55.90	Ref. [6]
1040.00	72.50	Ref. [6]

5.1.14 Stainless Steel 304

The default tabular values of thermal conductivity as a function of temperature for stainless steel 304 are listed below. Constant extrapolation is allowed from the lower end of the tabulated range. Linear extrapolation is allowed from the upper end of the tabulated range.

Stainless Steel 304

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
300.00	13.00	Ref. [6]
400.00	14.60	Ref. [6]
500.00	16.20	Ref. [6]
600.00	17.80	Ref. [6]
700.00	19.40	Ref. [6]
800.00	21.10	Ref. [6]
900.00	22.70	Ref. [6]
1000.00	24.30	Ref. [6]
1100.00	25.90	Ref. [6]

Stainless Steel 304

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
1200.00	27.50	Ref. [6]
1300.00	29.10	Ref. [6]
1400.00	30.80	Ref. [6]
1500.00	32.40	Ref. [6]
1600.00	34.00	Ref. [6]
1700.00	35.60	Ref. [6]
1700.01	17.80	Ref. [6]
1800.00	18.10	Ref. [6]
1900.00	18.50	Ref. [6]
2000.00	18.80	Ref. [6]
2100.00	19.10	Ref. [6]
2200.00	19.40	Ref. [6]
2300.00	19.80	Ref. [6]
2400.00	20.10	Ref. [6]
2500.00	20.40	Ref. [6]
2600.00	20.70	Ref. [6]
2700.00	21.10	Ref. [6]
2800.00	21.40	Ref. [6]
2900.00	21.70	Ref. [6]
3000.00	22.00	Ref. [6]

5.1.15 Carbon Steel

The default tabular values of thermal conductivity as a function of temperature for carbon steel are listed below. No extrapolation is allowed.

Carbon Steel

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
273.15	45.437	Ref. [8]
373.15	44.229	Ref. [8]
473.15	42.681	Ref. [8]
573.15	40.794	Ref. [8]
673.15	38.568	Ref. [8]
773.15	36.002	Ref. [8]
873.15	33.098	Ref. [8]
973.15	29.854	Ref. [8]
1076.80	26.135	Ref. [8]

Carbon Steel

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
1173.15	27.100	Ref. [8]
1273.15	28.100	Ref. [8]
1373.15	29.100	Ref. [8]
1473.15	30.100	Ref. [8]
1573.15	31.100	Ref. [8]
1673.15	32.100	Ref. [8]
1773.15	33.100	Ref. [8]
1810.90	33.477	Ref. [8]
5000.00	33.477	Ref. [8]

5.1.16 Iron-Chromium-Aluminum

The default tabular values of thermal conductivity as a function of temperature for iron-chromium-aluminum are listed below. No extrapolation is allowed.

Iron-Chromium-Aluminum

Temp(K)	Thermal Conductivity (W/m-K)	Data Source
300.0	11.2	Ref. [9]
400.0	12.7	Ref. [9]
500.0	14.2	Ref. [9]
600.0	15.7	Ref. [9]
700.0	17.2	Ref. [9]
800.0	18.6	Ref. [9]
852.0	19.4	Ref. [9]
873.0	19.7	Ref. [9]
900.0	20.0	Ref. [9]
973.0	21.1	Ref. [9]
1173.0	23.9	Ref. [9]
1273.0	25.3	Ref. [9]
1373.0	26.7	Ref. [9]
1473.0	28.0	Ref. [9]
1573.0	29.4	Ref. [9]
1673.0	30.7	Ref. [9]
1773.0	32.0	Ref. [9]
1773.01	32.0	Ref. [9]

Iron-Chromium-Aluminum

Temp(K)	Thermal Conductivity (W/m-K)	Data Source
5000.0	32.0	Ref. [9]

5.1.17 Iron-Chromium-Aluminum Oxide

The default tabular values of thermal conductivity as a function of temperature for iron-chromium-aluminum oxide are listed below. No extrapolation is allowed.

Iron-Chromium-Aluminum Oxide

Temp(K)	Thermal Conductivity (W/m-K)	Data Source
273.15	4.0	Estimated
5000.0	4.0	Estimated

6. Density

The density of most materials may be computed as a constant value, a user-specified tabular function or a MELCOR default table.

6.1 Constant Density

The constant density may be input by the user or read from a MELCOR default table. There are no checks made on the consistency of user-input values for enthalpy, specific heat capacity, melting temperature, and latent heat of fusion.

The following materials have default values for the constant density:

Material	Density (kg/m^3)	Data Source
Zircaloy	6500.0	Ref. [12]
Zirconium Oxide	5600.0	Ref. [12]
Uranium Dioxide	10960.0	Ref. [12]
Stainless Steel	7930.0	Ref. [12]
Stainless Steel Oxide	5180.0	Ref. [12]
Boron Carbide	2520.0	Ref. [12]
Silver-Indium-Cadmium	9689.4	Ref. [2], at 1000 K
Uranium Metal	18210.0	Ref. [10], p. 78
Graphite	1730.0	Ref. [4], p. 436
Carbon Steel	7752.9	Ref. [8]
Iron-Chromium-Aluminum	7100.0	Ref. [13]

Material	Density (kg/m^3)	Data Source
Iron-Chromium-Aluminum Oxide	5180.0	Estimated

6.2 Tabular as a Function of Temperature

The user-specified tabular function to define a new material or to override the default table for an existing material is invoked by using a standard tabular function (see the TF Package Users' Guide) to input the density (kg/m^3) as a function of temperature (K). The densities used by the COR and FDI packages (see the users' guides for these packages) for user-defined tabular functions are determined by evaluating the respective tabular functions at 1000 K. If the input tabular function does not allow an evaluation to be made at 1000 K, an input error occurs. Currently, only constant functions should be user-input, since temperature dependent values are not addressed by the HS package.

The following materials have default tables for density that may be altered through user input tabular functions:

- Zircaloy
- Zirconium Oxide
- Uranium Dioxide
- Stainless Steel
- Stainless Steel Oxide
- Boron Carbide
- Silver-Indium-Cadmium
- Uranium Metal
- Graphite
- Concrete
- Carbon Steel
- Iron-Chromium-Aluminum
- Iron-Chromium-Aluminum Oxide

The default density values for the above materials are computed by linear interpolation of the tabulated values listed in Sections 6.2.1 through 6.2.15, below. Data sources are given with each table.

6.2.1 Zircaloy

The default tabular values of density as a function of temperature for Zircaloy are listed below. No extrapolation is allowed.

Zircaloy

Temp (K)	Density (kg/m3)	Data Source
273.15	6500.0	Ref. [12]
5000.0	6500.0	Ref. [12]

6.2.2 Zirconium Oxide

The default tabular values of density as a function of temperature for zirconium oxide are listed below. No extrapolation is allowed.

Zirconium Oxide

Temp (K)	Density (kg/m3)	Data Source
273.15	5600.0	Ref. [12]
5000.0	5600.0	Ref. [12]

6.2.3 Uranium Dioxide

The default tabular values of density as a function of temperature for uranium dioxide are listed below. No extrapolation is allowed.

Uranium Dioxide

Temp (K)	Density (kg/m3)	Data Source
273.15	10960.0	Ref. [12]
5000.0	10960.0	Ref. [12]

6.2.4 Stainless Steel

The default tabular values of density as a function of temperature for stainless steel are listed below. No extrapolation is allowed.

Stainless Steel

Temp (K)	Density (kg/m3)	Data Source

273.15	7930.0	Ref. [12]
5000.0	7930.0	Ref. [12]

6.2.5 Stainless Steel Oxide

The default tabular values of density as a function of temperature for stainless steel oxide are listed below. No extrapolation is allowed.

Stainless Steel Oxide

Temp (K)	Density (kg/m3)	Data Source
273.15	5180.0	Ref. [12]
5000.0	5180.0	Ref. [12]

6.2.6 Boron Carbide

The default tabular values of density as a function of temperature for boron carbide are listed below. No extrapolation is allowed.

Boron Carbide

Temp (K)	Density (kg/m3)	Data Source
273.15	2520.0	Ref. [12]
5000.0	2520.0	Ref. [12]

6.2.7 Silver-Indium-Cadmium

The default tabular values of density as a function of temperature for silver-indium-cadmium are listed below. No extrapolation is allowed.

Silver-Indium-Cadmium

Temp (K)	Density (kg/m3)	Data Source
273.15	9689.4	Ref. [2], at 1000 K
5000.0	9689.4	Ref. [2], at 1000 K

6.2.8 Uranium Metal

The default tabular values of density as a function of temperature for uranium metal are listed below. No extrapolation is allowed.

Uranium Metal

Temp (K)	Density (kg/m3)	Data Source
273.15	19080.0	Ref. [10] p. 78, extrapolated
298.0	19050.0	Ref. [10] p. 78
366.3	18970.0	Ref. [10] p. 78
477.4	18870.0	Ref. [10] p. 78
588.6	18760.0	Ref. [10] p. 78
699.7	18640.0	Ref. [10] p. 78
810.8	18500.0	Ref. [10] p. 78
921.9	18330.0	Ref. [10] p. 78
1406.0	17580.0	Ref. [10] p. 78, extrapolated
5000.0	17580.0	Constant beyond melting point of 1406 K

6.2.9 Graphite

The default tabular values of density as a function of temperature for graphite are listed below. No extrapolation is allowed.

Graphite

Temp (K)	Density (kg/m3)	Data Source
273.15	1730.0	Ref. [4] p. 436, nuclear graphite, grade A
5000.0	1730.0	Ref. [4] p. 436, nuclear graphite, grade A

6.2.10 Concrete

The default tabular values of density as a function of temperature for concrete are listed below. No extrapolation is allowed.

Concrete

Temp (K)	Density (kg/m3)	Data Source
273.15	2306.7	Ref. [5] p. 635, stone concrete @ 294 K
5000.0	2306.7	Ref. [5] p. 635, stone concrete @ 294 K

6.2.11 Aluminum

The default tabular values of density as a function of temperature for aluminum are listed below. Constant extrapolation is allowed from the lower end of the tabulated range. Linear extrapolation is allowed from the upper end of the tabulated range.

Aluminum

Temp (K)	Density (kg/m3)	Data Source
273.15	2705.00	Ref. [6]
300.00	2701.00	Ref. [6]
400.00	2681.00	Ref. [6]
500.00	2661.00	Ref. [6]
600.00	2639.00	Ref. [6]
800.00	2591.00	Ref. [6]
933.00	2559.00	Ref. [6]
933.01	2385.00	Ref. [6]
1000.00	2365.00	Ref. [6]
1200.00	2305.00	Ref. [6]
1400.00	2255.00	Ref. [6]

6.2.12 Aluminum Oxide

The default tabular values of density as a function of temperature for aluminum oxide are listed below. Linear extrapolation is allowed from the upper end of the tabulated range.

Aluminum Oxide

Temp (K)	Density (kg/m3)	Data Source
273.15	4000.0	Ref. [7]
5000.0	4000.0	Ref. [7]

6.2.13 Cadmium

The default tabular values of density as a function of temperature for cadmium are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Cadmium

Temp (K)	Density (kg/m3)	Data Source
273.15	8670.0	Ref. [6]
283.15	8660.0	Ref. [6]
293.15	8650.0	Ref. [6]
303.15	8640.0	Ref. [6]
313.15	8630.0	Ref. [6]

Cadmium

Temp (K)	Density (kg/m3)	Data Source
323.15	8620.0	Ref. [6]
333.15	8610.0	Ref. [6]
343.15	8600.0	Ref. [6]
353.15	8590.0	Ref. [6]
363.15	8580.0	Ref. [6]
373.15	8570.0	Ref. [6]
383.15	8561.0	Ref. [6]
393.15	8551.0	Ref. [6]
403.15	8541.0	Ref. [6]
413.15	8531.0	Ref. [6]
423.15	8521.0	Ref. [6]
433.15	8511.0	Ref. [6]
443.15	8501.0	Ref. [6]
453.15	8491.0	Ref. [6]
463.15	8481.0	Ref. [6]
473.15	8470.0	Ref. [6]
483.15	8460.0	Ref. [6]
493.15	8450.0	Ref. [6]
503.15	8439.0	Ref. [6]
513.15	8428.0	Ref. [6]
523.15	8417.0	Ref. [6]
533.15	8406.0	Ref. [6]
543.15	8395.0	Ref. [6]
553.15	8384.0	Ref. [6]
563.15	8372.0	Ref. [6]
573.15	8360.0	Ref. [6]
583.15	8348.0	Ref. [6]
594.00	8336.0	Ref. [6]
594.01	8016.0	Ref. [6]
600.00	8010.0	Ref. [6]
800.00	7805.0	Ref. [6]
1000.00	7590.0	Ref. [6]
1040.00	7547.0	Ref. [6]

6.2.14 Stainless Steel 304

The default tabular values of density as a function of temperature for stainless steel 304 are listed below. Constant extrapolation is allowed from both ends of the tabulated range.

Stainless Steel 304

Temp (K)	Density (kg/m3)	Data Source
273.15	8025.00	Ref. [6]
323.15	8003.00	Ref. [6]
373.15	7981.00	Ref. [6]
423.15	7958.00	Ref. [6]
473.15	7936.00	Ref. [6]
523.15	7914.00	Ref. [6]
573.15	7891.00	Ref. [6]
623.15	7869.00	Ref. [6]
673.15	7847.00	Ref. [6]
723.15	7824.00	Ref. [6]
773.15	7802.00	Ref. [6]
823.15	7780.00	Ref. [6]
873.15	7757.00	Ref. [6]
923.15	7735.00	Ref. [6]
973.15	7713.00	Ref. [6]
1023.15	7690.00	Ref. [6]
1073.15	7668.00	Ref. [6]
1123.15	7646.00	Ref. [6]
1173.15	7623.00	Ref. [6]
1223.15	7601.00	Ref. [6]
1273.15	7579.00	Ref. [6]
1373.15	7534.00	Ref. [6]
1473.15	7489.00	Ref. [6]
1573.15	7445.00	Ref. [6]
1673.15	7400.00	Ref. [6]
1700.00	7388.00	Ref. [6]
1700.01	6926.00	Ref. [6]
1800.00	6862.00	Ref. [6]
1900.00	6785.00	Ref. [6]
2000.00	6725.00	Ref. [6]
2100.00	6652.00	Ref. [6]
2200.00	6576.00	Ref. [6]

Stainless Steel 304

Temp (K)	Density (kg/m3)	Data Source
2300.00	6498.00	Ref. [6]
2400.00	6416.00	Ref. [6]
2500.00	6331.00	Ref. [6]
2600.00	6243.00	Ref. [6]
2700.00	6152.00	Ref. [6]
2800.00	6058.00	Ref. [6]
2900.00	5961.00	Ref. [6]
3000.00	5861.00	Ref. [6]

6.2.15 Carbon Steel

The default tabular values of density as a function of temperature for carbon steel are listed below. No extrapolation is allowed.

Carbon Steel

Temp (K)	Density (kg/m3)	Data Source
273.15	7752.9	Ref. [8]
5000.00	7752.9	Ref. [8]

6.2.16 Iron-Chromium-Aluminum

The default tabular values of density as a function of temperature for iron-chromium-aluminum are listed below. No extrapolation is allowed.

Iron-Chromium-Aluminum

Temp (K)	Density (kg/m3)	Data Source
273.15	7100.0	Ref. [13]
5000.00	7100.0	Ref. [13]

6.2.17 Iron-Chromium-Aluminum Oxide

The default tabular values of density as a function of temperature for iron-chromium-aluminum oxide are listed below. No extrapolation is allowed.

Iron-Chromium-Aluminum Oxide

Temp (K)	Density (kg/m3)	Data Source
273.15	5180.0	Estimated
5000.00	5180.0	Estimated

7. Constant Melting Temperature

The melting temperature may be input by the user or read from a MELCOR default table. There are no checks made on the consistency of user-input values for enthalpy, specific heat capacity, melting temperature, and latent heat of fusion.

The following materials have default tables for the melting temperature:

Material	Melt Temperature (K)	Data Source
Zircaloy	2098.0	Ref. [1]
Zirconium Oxide	2990.0	Ref [12]
Uranium Dioxide	3113.0	Ref. [1]
Stainless Steel	1700.0	Estimated
Stainless Steel Oxide	1870.0	Ref. [12], Fe3O4
Boron Carbide	2620.0	Ref. [12]
Silver-Indium-Cadmium	1075.0	Ref. [2]
Uranium Metal	1406.0	Ref. [12]
Graphite	3866.0	Ref. [4]
Aluminum	933.0	Ref. [6]
Aluminum Oxide	2327.0	Ref. [7]
Cadmium	594.0	Ref. [6]
Stainless Steel 304	1700.0	Ref. [6]
Carbon Steel	1810.9	Ref. [8]
Iron-Chromium-Aluminum	1773.0	Ref. [13]
Iron-Chromium-Aluminum Oxide	1901.0	Estimated

8. Constant Latent Heat of Fusion

The latent heat of fusion may be input by the user or read from a MELCOR default table. There are no checks made on the consistency of user-input values for enthalpy, specific heat capacity, melting temperature, and latent heat of fusion.

The following materials have default tables for the latent heat of fusion:

Material	Heat of Fusion (J/kg)	Data Source
Zircaloy	2.25E5	Ref. [1]
Zirconium Oxide	7.07E5	Ref [12]
Uranium Dioxide	2.74E5	Ref [1]
Stainless Steel	2.68E5	Estimated
Stainless Steel Oxide	5.98E5	Ref. [12], Fe3O4
Boron Carbide	5.00E5	Estimated
Silver-Indium-Cadmium	9.80E4	Ref. [2]
Uranium Metal	5.025E4	Ref. [4]
Aluminum Oxide	1.07E6	Ref. [14]
Aluminum	3.978E5	Ref. [6]
Cadmium	5.500E4	Ref. [6]
Stainless Steel 304	2.692E5	Ref. [6]
Carbon Steel	2.71960E5	Ref. [8]
Iron-Chromium-Aluminum	2.7E5	Ref. [13]
Iron-Chromium-Aluminum Oxide	6.87463E5	Estimated

References

1. D. L. Hagrman, G. A. Reymann, and R. E. Mason, MATPRO VERSION 11 (Revision 1) A Handbook of Materials Properties for Use in the Analysis of Light Water Reactor Fuel Rod Behavior, NUREG/CR-0497 and TREE-1280 Rev. 1, EG&G Idaho, Inc., Idaho Falls, ID (February 1980).

2. D. L. Hagrman, Materials Properties for Severe Core Damage Analysis, EGG-CDD-5801, EG&G, Idaho Falls, ID (May 1982).

3. F. P. Incropera and D. P. Dewitt, Fundamentals of Heat Transfer, John Wiley and Sons, New York (1981).

4. CRC Handbook of Tables for Applied Engineering Science, 2nd ed., Ed. by R. E. Bolz and G. L. Tuve, CRC Press, Boca Raton FL (1973).

5. F. Kreith, Principles of Heat Transfer, 3rd ed., Intext Educational Publishers, New York (1973).

6. R. C. Nause and M. T. Leonard, Thermophysical Property Assessment Report for Savannah River Site Production Reactor Materials, Science Applications International Corporation, SAIC Report No. 89/6507, Rev. 1 (May 1990).

7. Thermodynamic and Transport Properties for the CONTAIN Code, Sandia National Laboratories, NUREG/CR-5173 (December 1988).

8. Data for carbon steel duplicate those defined and used in the Bottom Head (BH) package. Values were obtained from various subroutines included in that package. References cited in the coding are: (1) Y. S. Touloukian, Thermophysical Properties of High Temperature Solid Materials, and (2) the ASTM Metals Handbook (pages 1-63, 64) for low-alloy steels (particularly, SA508 steel).

9. K.G. Field, Handbook on the Material Properties of FeCrAl Alloys for Nuclear Power Production Applications (FY 18 Rev. 1.1), ORNL/SPR-2018/905, August 2018.

10. M. M. Wakil, Nuclear Heat Transport, International Textbook Co., New York (1971).

11. Heard, et al., N-Reactor Safety Enhancement Report HEDL-TC 2977, Westinghouse Hanford Company, September 1987.

12. CRC Handbook of Chemistry and Physics, 63rd ed., CRC Press, Boca Raton, FL (1982).

13. Kathal APM Tube Datasheet, Kanthal – Update as of 7/6/2018

14. CRC Handbook of Chemistry and Physics, 71st ed., CRC Press, Boca Raton, FL (1990).

Sodium Chemistry (NAC) Package

The MELCOR Sodium Chemistry (NAC) package is intended for applications where sodium is the principal fluid for heat and energy transfer. The NAC package provides chemistry models for sodium spray and sodium pool fires as well as atmospheric chemistry. These sodium chemistry models are based on the CONTAIN-LMR code. The equations of state for sodium must replace water as the hydrodynamic fluid in the Control Volume Hydrodynamic (CVH) package to use the NAC package.

This Reference Manual provides an overview of the models included in the NAC package. The NAC Package Users' Guide contains a description of the NAC package MELGEN and MELCOR input.

CONTENTS

List of Figures

List of Tables

1. Introduction

The MELCOR Sodium Chemistry (NAC) package provides sodium chemistry models to support the analysis of certain postulated liquid sodium accidents, such as loss of coolant accidents. Sodium readily reacts exothermically with common atmospheric constituents, oxygen and water. The resulting energy generation and formation of aerosolized sodium oxides are pertinent phenomena to model in support of Level 2 and Level 3 probabilistic analyses as well as properly characterizing the timing and magnitude of any environmental source term for postulated accidents of advanced sodium-cooled reactor designs.

The sodium chemistry models incorporated into MELCOR are derived from the CONTAIN-LMR code that has been maintained and improved by the Japan Atomic Energy Agency (JAEA). The NAC package computes the combustion rate of sodium in spray and pool geometries as well as reactions between water vapor and sodium vapor and aerosol residing in the atmosphere. Computed mass and energy exchange rates from the sodium chemistry models are provided as sources to the appropriate code packages.

Application of the NAC package requires users to define specific database elements to support the NAC package interfaces with other code packages. Specifically, spray and pool geometries are dependent on the principal hydrodynamic material present in the Control Volume Hydrodynamic (CVH) package. The physical properties and equation of state (EOS) of the principal hydrodynamic material are available to all packages upon request from the Control Volume Thermodynamic (CVT) package. Therefore, analyses of sodium applications require a user to identify sodium as the principal hydrodynamic material. This replaces water in the CVH and CVT packages and is performed by selecting one of the two independently developed sodium equations of state (EOS) datafiles that are distributed along with the MELCOR executables, see the NAC Users' Guide for a detailed description on applying a sodium EOS.

The mass of sodium, once identified as the principal hydrodynamic material, is an explicit component of the CVH package. Therefore, all chemical reactions that decrement the mass of sodium, whether liquid or vapor, result in a negative source of sodium to the corresponding control volume. The reaction products, which are not a principal hydrodynamic material, are modeled as part of the RadioNuclide (RN) package. Therefore, the mass generation rate sources product masses to user identified RN classes. In general, for all the chemistry models this requires specification of the following classes: 'NAOH', 'NA2O', 'NA2O2', and 'NA' in the DecayHeat (DCH) and RN package (see the DCH_CL record). By default, the application of a sodium EOS file replaces the default RN water class, 'H2O', with a representative sodium class, 'NA'. This permits the interfaces between the RN and CVH packages to be generalized for transferring sodium among RN, CVH, and NAC packagesa. Please note that while sodium is a class constituent within the alkali metals class, represented by Cesium, the default alkali metal

is comprised of radioisotopes produced within the fuel components. Therefore, the hydrodynamic class representation of sodium should remain separate from the radioisotopes of sodium included in the alkali metal class.

Specifically, for the atmospheric chemistry model, the chemical reactions involving water vapor add the additional requirement that water vapor be defined. The only facility available currently is to identify a 'H2OV' non-condensible gas within the Non-Condensible Gas (NCG) package. See the NCG_ID record for identifying default gas species. Once a two-component condensation model is incorporated, this requisite is no longer necessary.

2. Phenomenology

This section describes the sodium chemical reaction and generation of aerosol sources from sodium spray and pool fires.

2.1 Sodium Reaction Chemistry

In both spray and pool fire scenarios, sodium vapors can chemically react with both oxygen and water present in the atmosphere, forming various oxides. Sodium monoxide (Na_2O) and sodium peroxide (Na_2O_2), are the most abundant reaction products. Sodium monoxide is formed by:

$$2Na + \frac{1}{2}O_2 \rightarrow Na_2O \qquad (2\text{-}1)$$

This reaction releases h_{Na_2O} (J/kg) of combustion heat with excess oxygen. Sodium monoxide oxidizes to form sodium peroxide:

$$Na_2O + \frac{1}{2}O_2 \rightarrow Na_2O_2 \qquad (2\text{-}2)$$

This reaction releases additional energy, giving a total energy of formation of $h_{Na_2O_2}$ (J/kg).

2.2 Sodium Transport and Interactions

Figure 2.1 shows a graphical representation of sodium chemistry phenomena that can occur in a generic fast reactor containment. Sodium vapor of liquid droplets can leak from a reactor system into the surrounding containment or confinement volume. If oxygen is present as air or water vapor, the sodium vapor and liquid droplets can undergo combustion. This exothermic reaction is referred to as sodium spray fire. If sodium droplets are not completely burned, they can fall onto a horizontal surface to form a sodium pool. The sodium pool may also undergo combustion, forming a pool fire. Sodium condensate along containment walls may also combust.

Sodium combustion forms byproducts in aerosol form. The primary combustion products are sodium monoxide and sodium peroxide when reacting with oxygen as well as sodium hydroxide when reacting with water. Other by-products may form if sodium interacts with concrete (i.e., via containment liner failure). The combustion product aerosols may contain activated sodium and entrained fission products. If these byproduct aerosols are not removed from the atmosphere, they can release to the environment and contribute to the accident source term.

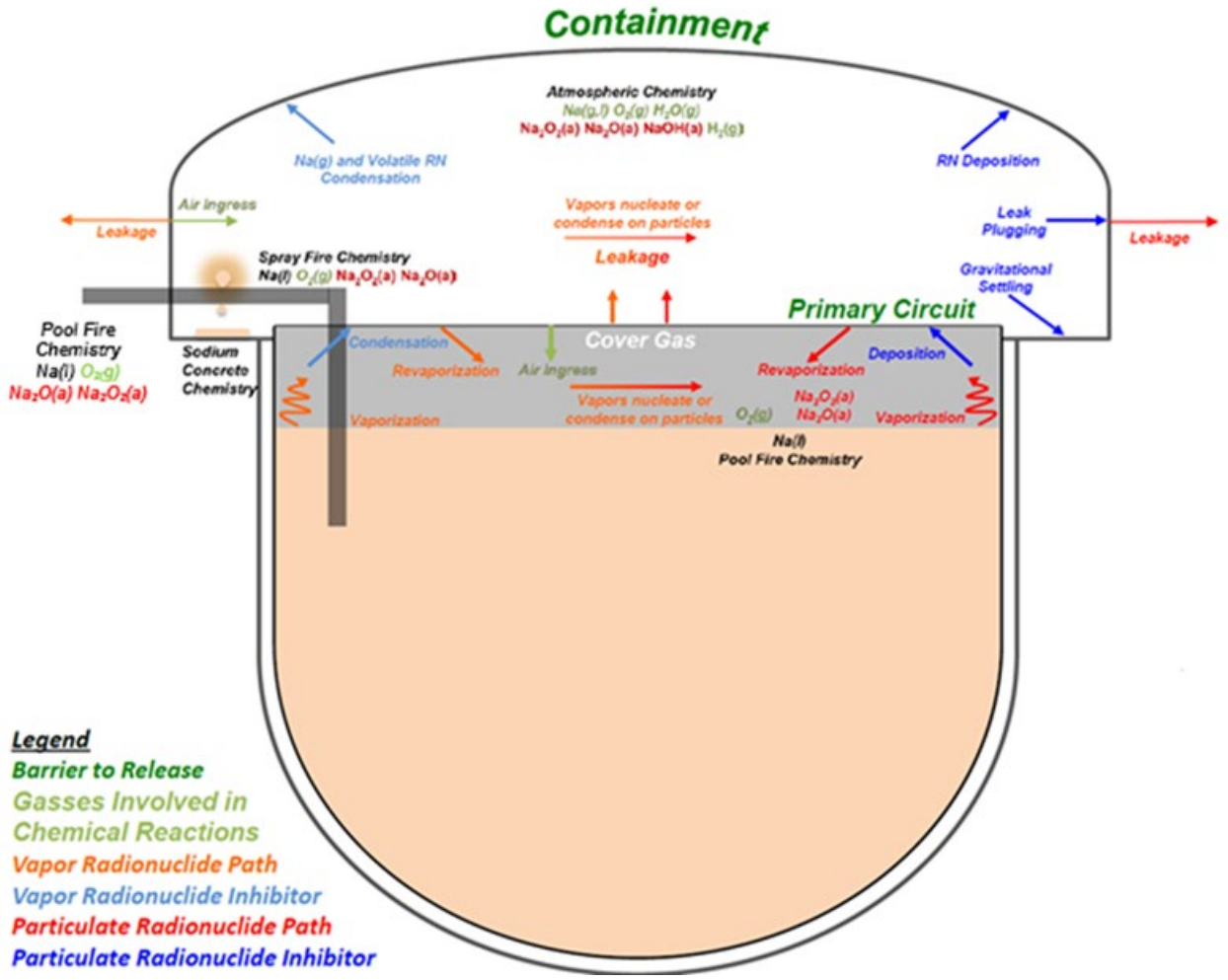


Figure 2.1 Graphical Representation of Sodium Phenomena in a Generic Containment. Adapted from ANL-ART-3 [1].

3. Models

3.1 Sodium Spray Fire

The sodium spray fire scenario considered in this section is generated by sodium droplets originating from a component failure (e.g., a pipe break). Based on the break conditions, the initial velocity and direction of the release sodium can influence the drop fall duration. This duration may limit the total combustion before any remain spray droplets are transitioned to a pool geometry. For relatively high velocity, parabolic paths could also increase the fall duration if orientated upward. However, if the pipe is not significantly pressurized, or if the break is oriented downwards, the droplets may simply flow down in a stream. Regardless of the break direction, the droplets energetically react with oxygen and water in the surrounding atmosphere.

3.1.1 Sodium Spray Model

The sodium spray model determines the mass of sodium reacted, energy generated and transferred to the surrounding control volume, and the oxide masses produced. The model is based on the phenomenological model used in NACOM, a code developed and tested at Brookhaven National Laboratory [2]. This model was later applied in CONTAIN-LMR and modified by JAEA. The JAEA modified version has been implemented into MELCOR.

The sodium combustion rate of the spray is computed by first discretizing the sodium spray source into discrete bin sizes. For each droplet bin size, a representative droplet is considered. An explicit marching scheme, applying a Runge-Kutta 4 for each step, evaluates the change in mass of the droplet while simultaneously evaluating the diameter and velocity. The velocity is evaluated using either the terminal velocity or an acceleration model, which the user specifies along with a corresponding spray fire timestep. The oxidation rate is determined from the physical properties of the sodium droplet and atmosphere. The mass reacted within a step is used to update the droplet diameter and velocity applied on the following step. The marching scheme is concluded for the representative droplet once the total fall height is reached or the droplet mass reaches zero.

At the end of each step within the marching scheme, an energy balance is applied to evaluate the change in the droplet temperature. If the temperature is found to exceed saturation temperature, additional sodium mass can be vaporized and placed into the CVH atmosphere if and only if oxygen is not present.

3.1.1.1 Droplet Size Distribution

The Nukiyama-Tanasawa correlation as presented in [2] is used to determine the droplet size distribution:

$$\frac{dF_V}{dD} = \left(\frac{3.915}{\bar{D}}\right)^6 \frac{D^5}{120} \exp\left(-\frac{3.915D}{\bar{D}}\right) \quad (3\text{-}1)$$

Where D is the droplet diameter, F_v is the volume fraction of the spray containing droplets small than D, and $\bar{D}$ is the volume mean droplet diameter. The Nukiyama-Tanasawa correlation assumes a downward flow at terminal velocity. Droplets from the sodium spray source are partitioned into 11 discrete droplet size classes.

3.1.1.2 Velocity Determination

The selection of either the acceleration model or terminal velocity model is made on the input field SPRDT of the NAC_SPRAY record. The user may either permit the acceleration model to determine the timestep or provide it. The terminal velocity model determines the timestep applied.

Acceleration Model

The equations for droplet motion are based on the droplet acceleration model.

The acceleration model uses a timestep, Δt, as specified by the user or by the following:

$$\Delta t = \min(a\Delta t, 0.02) \quad (3\text{-}2)$$

where a is a multiplier dependent on the droplet diameter:

$$a = \begin{cases} 0.5, & D \le 5e\text{-}4 \\ 1.0, & 5e\text{-}4 < D \le 3e\text{-}3 \\ 2.0, & D \ge 3e\text{-}3 \end{cases} \quad (3\text{-}3)$$

The droplet velocity is calculated using:

$$\frac{dv}{dt} = g - \frac{3}{4}\frac{\rho_g}{\rho_l}\frac{|v|v}{D\,C_d} \quad (3\text{-}4)$$

where v is the droplet velocity, g is gravitational acceleration, ρ_g is the convective gas density, ρ_l is the droplet density, and C_d is a drag coefficient. C_d is given by a piecewise function:

$$C_d = \begin{cases} \frac{24}{Re}, & Re < 1 \\ 2.6 + \frac{23.71}{Re}, & 1 \le Re < 6 \\ \frac{18.5}{Re^{0.6}}, & 6 \le Re < 500 \\ \frac{4}{9}, & Re \ge 500 \end{cases} \quad (3\text{-}5)$$

where Re is the Reynolds number, given by:

$$Re = \frac{vD\rho_g}{\mu_g} \quad (3\text{-}6)$$

where μ_g is the convective gas dynamic viscosity. Note that Equation (3-5) is the drag coefficient for a rigid sphere and does not necessarily represent the drag coefficient of a falling sodium droplet. Falling sodium droplets undergo deformation from inertial forces from both mass effusion and combustion gas buildup in the low-pressure region of the droplet wake. However, little experimental data exist studying drag coefficients under these conditions; whereas, the motion of a rigid sphere has been studied extensively. For this reason, the drag coefficient for a rigid sphere was applied.

A Runge-Kutta 4 method computes the mass of sodium consumed within a given timestep. The remaining sodium mass, fall height, diameter, and velocity are updated each spray fire model timestep until the remaining sodium mass or fall height reaches zero. If the bottom of the control volume is reached, the remaining sodium mass is transitioned to the pool.

Terminal Velocity Model

For the terminal velocity model, the spray fire model timestep is compute using the terminal velocity, v_t, and the fall height, H, as shown:

$$\Delta t = \frac{1}{8}\left(\frac{H}{v_t}\right) \quad (3\text{-}7)$$

The terminal velocity is taken from a correlation that approximates the steady state solution of Equation (3-4).

The solution scheme is the same as the acceleration model, where the terminal velocity is re-evaluated for each sodium spray fire model timestep and used to approximate the velocity within a timestep.

3.1.1.3 Stationary Droplet Mass Burn Rate

Vapor-phase droplet combustion theory typically assumes an individual stationary, symmetrical burning droplet surrounded by a spherical, symmetrical burning zone where all fuel is reaching the burning zone is under steady state conditions. Figure 3.1 shows a schematic of a burning droplet.

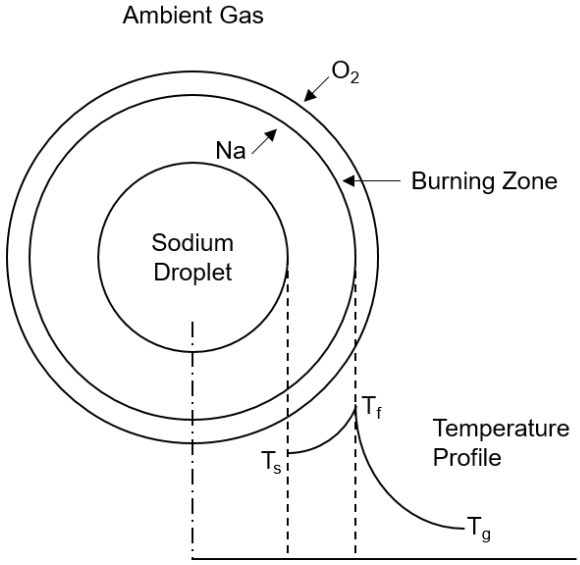


Figure 3.1 Schematic of a Stationary Burning Sodium Droplet

For a single stationary, symmetrical burning droplet surrounded by a spherical, symmetrical burning zone where all fuel is reaching the burning zone is under quasi-steady state conditions, the mass burning rate of the droplet is a function of the evaporation of the liquid droplet, which is a function of heat transfer to the droplet. The mass burning rate of a droplet is related to the change in droplet size by:

$$\dot{m} = -\frac{d}{dt}\left(\frac{\pi}{6}D^3\rho\right) \qquad (3\text{-}8)$$

where $\dot{m}$ is the mass burn rate, D is the droplet diameter, and ρ is the droplet density. Equation (3-8) may be rearranged as:

$$\frac{d\left(D^2\right)}{dt} = -\frac{4\dot{m}}{\pi\rho D} \qquad (3\text{-}9)$$

Equation (3-9) resembles the "D^2 law" for droplet lifetime for combustion of hydrocarbon fuel droplets. The "D^2 law" states that the vaporization time of a droplet decreases quadratically with droplet size. Experimental work has shown that combustion of sodium drops essential follows the "D^2 law". Mathematically, the "D^2 law" is:

$$\frac{d(D^2)}{dt} = -K \tag{3-10}$$

where *K* is a proportionality constant. Integrating Equation (3-10) gives a linear relationship for the squared droplet diameter:

$$D_n^2 = D_o^2 - K\Delta t \tag{3-11}$$

where D_n is the new droplet diameter, D_o is the old droplet diameter, and Δt is the time interval.

Substituting Equation (3-10) into Equation (3-9) and solving for $\dot{m}$ gives a final expression for the mass burn rate of a stationary droplet:

$$\dot{m} = \frac{\pi\rho K}{4} D \tag{3-12}$$

One established solution for *K* given by Spalding [3] is:

$$K = \frac{8k}{C_P\rho} \ln(1+B) \tag{3-13}$$

where k is the gas mixture thermal conductivity, C_p is the gas mixture heat capacity, and B is a transfer number. *B* is defined as:

$$B = \frac{1}{h_{fg}}\left[C_p(T_g - T_{sat}) + \frac{q_c Y_o}{i}\right] \tag{3-14}$$

where h_{fg} is the heat of evaporation for sodium, T_g is the oxidizing gas temperature, T_{sat} is the sodium saturation temperature at 1 atm), q_c is the heat of combustion, Y_o is the ambient oxygen mass fraction, and *i* is the stochiometric ratio of oxygen to sodium. Unlike the CONTAIN-LMR model, if B is less than or equal to zero no pre-ignition oxidation or energy transfer is computed.

Both Q_c and *i* in Equation (3-14) depend on the composition of the combustion reaction products. As discussed in Section 2.1, the primary reactants from sodium combustion in air are Na_2O and Na_2O_2. The fraction of sodium mass that reacts to form Na_2O_2 can be expressed as:

$$S = \frac{1.3478f}{1.6957 - 0.3479f} \tag{3-15}$$

where *f* is the fraction of Na_2O_2 present in the reaction products. Using *S*, the heat of combustion is:

$$q_c = h_{Na_2O}(1 - S) + h_{Na_2O_2}(S) \quad (3\text{-}16)$$

Where h_{Na_2O} and $h_{Na_2O_2}$ are the specific heat of formations to produce sodium monoxide and peroxide. Their respective values are hardcoded as 1.381e7 and 1.588e7 J/kg. The Stochiometric ratio, *i*, is:

$$i = \frac{S}{1.4375} + \frac{1 - S}{2.875} \quad (3\text{-}17)$$

3.1.1.4 Modified Burn Rate for Free-Falling Droplet

The mass burn rate derived in Section 3.1.1.3 only considers a stationary droplet. A free-falling droplet experiences forced convection that increases the burn rate by causing deviations in the spherical burning zone. Analytical solutions for the droplet burning rate under forced convective conditions are difficult to solve. Therefore, empirical and semi-empirical correlations are often used to calculate falling droplet burning rates.

Falling droplets have a preignition period where oxidation reactions cause a coarse film of oxides forms on the droplet surface. Heat generated from oxidation is transferred more readily to the droplet than the atmosphere, and the droplet temperature rapidly increases as a result. The preignition phase ends when the droplet temperature reaches the ignition temperature.

Following ignition, the droplet burns in the vapor phase. The burn rate for a free-falling droplet in the vapor phase can be calculated using the equations for a stationary droplet and a multiplication factor to account for convective effects:

$$\dot{m}_f = \dot{m}\left(1 + C_f Re^{\frac{1}{2}} Pr^{\frac{1}{3}}\right) \quad (3\text{-}18)$$

where $\dot{m}_f$ is the mass burn rate for a free-falling droplet, $\dot{m}$ is the mass burn rate for a stationary droplet given in Equation (3-18), C_f is an empirical constant, Re is the Reynolds number, and Pr is the Prandtl number. For this model, C_f is 0.3. The Prandtl number is:

$$Pr = \frac{c_p \mu_g}{k} \quad (3\text{-}19)$$

where c_p is the convective gas specific heat and k is the convective gas thermal conductivity.

At the end of each integration step, the droplet temperature is recalculated from an energy balance between the combustion energy and the energy lost.

$$\dot{Q}_{com}=\dot{m}\, q_c \tag{3-20}$$

$$\dot{Q}_{con}=\dot{m}\, (q_c-h_{fg}) \tag{3-21}$$

$$T_s^{\,n}=T_s^{\,o}+\frac{(\dot{Q}_{com}-\dot{Q}_{con})}{m\ Cp}\Delta t \tag{3-22}$$

The spray droplet temperature is limited by the saturation temperature of sodium.

3.2 Pool Fire

The sodium pool fire model is adapted from CONTAIN-LMR that is based on the SOFIRE II code. The SOFIRE II model was developed from the results of pool fire experiments. The tests concluded that the sodium burning rate was proportional to the oxygen concentration and was controlled by diffusion of oxygen to the pool surface through a convective boundary layer. As a result, the oxygen mass consumption rate for a pool fire is:

$$\dot{m}_{O2} = A_S H_G \rho_g Y_{O2} \tag{3-23}$$

where $\dot{m}_{O2}$ is the oxygen mass consumption rate, A_S is the surface area of the sodium pool, H_G is a gas transport coefficient, ρ_g is the gas density, and Y_{O2} is the ambient mass fraction of oxygen in the atmosphere. The gas transport coefficient is defined as:

$$H_G = 0.14 D_{diff}\left(g\, S_c \frac{\beta}{\nu^2}\, |T_{surf} - T_g|\right)^{\frac{1}{3}} \tag{3-24}$$

where D_{diff} is the gas diffusion coefficient, g is the gravitational constant, S_C is the Schmidt number, β is the coefficient of gas expansion, ν is the kinematic viscosity, T_{surf} is the sodium pool surface temperature, and T_g is the gas temperature. The diffusion coefficient is:

$$D_{diff} = \frac{6.4312\times10^{-5}}{P_g}\left[\frac{(T_{surf} + T_g)}{2}\right]^{1.823} \tag{3-25}$$

where P_g is the gas pressure. The Schmidt number is:

$$S_c = \frac{\nu}{D_{diff}} \tag{3-26}$$

The coefficient of gas expansion is:

$$\beta = \frac{1}{\frac{1}{2}(T_{surf}+T_g)} \tag{3-27}$$

The mass burn rate for sodium is related to the consumption rate of oxygen as:

$$\dot{m}_{Na} = \dot{m}_{O2} S \tag{3-28}$$

where $\dot{m}_{Na}$ is the sodium pool burn rate and S is the stochiometric combustion ratio of sodium to oxygen:

$$S = 2.88 f_{O2} + 1.44(1 - f_{O2}) \tag{3-29}$$

where f_{O2} is the fraction of the total consumed oxygen that forms Na_2O. An upper limit is imposed on $\dot{m}_{Na}$ such that the amount of sodium burned may not exceed one half of the pool mass within a single timestep.

Unlike the spray fire model, the heat of formations for Na_2O and Na_2O_2, h_{Na_2O} and $h_{Na_2O_2}$, are hardcoded as 9.08e6 and 1.048e7 J/kg. Base on the user specified value for f_{O2}, the specific heat of combustion at a given reference temperature, T_{ref}, of 298.15 is computed as:

$$\begin{aligned} q_{com} = & \left(2 f_{O2} MW_{2Na} h_{Na_2O} + (1 - f_{O2}) MW_{2Na} h_{Na_2O_2}\right) \frac{\dot{m}_{O2}}{MW_{O2}} \\ & - \dot{m}_{O2} \Delta h_{O2,Tref} - \dot{m}_{Na} \Delta h_{Na,Tref} \\ & + \left(2 f_{O2} MW_{Na2O} \Delta h_{Na_2O,Tref} \right. \\ & \left. + (1 - f_{O2}) MW_{Na2O2} \Delta h_{Na_2O_2,Tref}\right) \frac{\dot{m}_{O2}}{MW_{O2}} \end{aligned} \tag{3-30}$$

where, $\Delta h_{n,Tref}$ is the specific enthalpy evaluated at T_{ref} and MW_n is the molecular weight of compound n. The resulting energy balance is given as:

$$\begin{aligned} q_{bal} = & \dot{m}_{O2} \Delta h_{O2,Tatm} + \dot{m}_{Na} \Delta h_{Na,Tatm} + q_{com} \\ & - f_{O2} \dot{m}_{O2} \left(f_{Na_2O,2p} \Delta h_{Na_2O,Tpool} \right. \\ & \left. + \left(1 - f_{Na_2O,2p}\right) \Delta h_{Na_2O,Tatm}\right) \\ & - (1 - f_{O2}) \dot{m}_{O2} \left(f_{Na_2O_2,2p} \Delta h_{Na_2O_2,Tpool} \right. \\ & \left. + \left(1 - f_{Na_2O_2,2p}\right) \Delta h_{Na_2O_2,Tatm}\right) \end{aligned} \tag{3-31}$$

Where, $\Delta h_{n,Tm}$ is the specific enthalpy of compound n evaluated at the temperature of the deposition location, either the atmosphere or the pool, and f_{x2p} is the deposition fraction of x, either Na_2O and Na_2O_2, to the pool.

The associated enthalpies are placed into the corresponding field, atmosphere or pool. The byproduct mass fractions are similarly deposited to the host field for the aerosolized Na_2O and Na_2O_2. Transportation and deposition are treated by the CVH and RN packages, respectively.

3.3 Atmospheric Chemistry Model

Unlike the previous two sodium fire models, the atmospheric chemistry model covers the other interactions between sodium and atmospheric constituents not specifically addressed by the spray or pool fire models. Sodium may reside in various states, or locations, associated with different MELCOR packages: sodium vapor within the CVH package, condensate film on heat structures within the Heat Structure (HS) package, and as aerosol within the RN package. The atmospheric chemistry computes the reaction rates for sodium and sodium oxides across the stated locations. These reactions are generally exothermic, which can add thermal load to the containment system. In addition, any hydrogen generated by the sodium chemical reactions may have additional consequences (e.g., a hydrogen combustion and explosion).

3.3.1 Gas Chemistry

The gas chemistry in this model includes the reactions between sodium and gas and aerosol constituents in the atmosphere, which include water, oxygen, and sodium by-product aerosols. Resulting by-products includes NaOH, Na_2O, Na_2O_2 and hydrogen. The reactants include Na, H_2O, Na_2O, and Na_2O_2. The reactions, descriptions, and reaction energies are provided in Table 3.1. The corresponding definition of the reaction heat is given in Table 3.2. As noted in Table 3.1, reaction 1 requires H_2O as an aerosol. Currently, MELCOR only treats water as a non-condensible material and is treated only as an ideal gas at present. Thus, reaction 1 has been de-activated until the two-condensable model has been implemented. Similarly, reaction 2 requires the water vapor to exceed oxygen fraction in the atmosphere, which means a high humidity. Thus, treatment of ideal gas may not be adequate. Therefore, this reaction has also been de-activated.

Table 3.1 Atmospheric Chemistry Reactions Data and Applications

#	Reaction	Reaction Heat	Locations
1	$Na + H_2O\ (l) \rightarrow NaOH + \frac{1}{2}H_2$	$\Delta H_{NaOH} - \Delta H_{H2O}$	Reactions within aerosol particles or within aerosol deposit or condensable film (inactive)
2	$2\ Na + H_2O\ (g) \rightarrow Na_2O + H_2$	$\Delta H_{Na2O} - \Delta H_{H2O}$	Reactions with gases, aerosols with gases, and deposits or film with gases (inactive)
3	$2\ Na + (1 - 0.5 \cdot f_{ox})O_2 \rightarrow f_{ox} \cdot Na_2O + (1 - 0.5 \cdot f_{ox}) \cdot Na_2O_2$	$f_{ox} \cdot \Delta H_{Na2O} + (1 - f_{ox}) \cdot \Delta H_{Na2O2}$	Reactions of gases with gases, aerosols, and deposits/film
4	$Na_2O_2 + 2\ Na \rightarrow 2\ Na_2O$	$\Delta H_{Na2O} - 0.5\ \Delta H_{Na2O2}$	Contact reactions within an aerosol particle or within aerosol deposits or condensable film. aerosol reaction with excess sodium in the atmosphere, and aerosol reactions on the film
5	$Na_2O + H_2O\ (g) \rightarrow 2\ NaOH$	$\Delta H_{NaOH} - 0.5\ (\Delta H_{Na2O} + \Delta H_{H2O})$.	Reactions within a repository, such as within an aerosol, aerosol deposit or condensable film, reactions of aerosol with gas, and reacting deposits or film with gases
6	$Na_2O_2 + H_2O\ (g) \rightarrow 2\ NaOH + \frac{1}{2}O_2$	$\Delta H_{NaOH} - 0.5(\Delta H_{Na2O2} + \Delta H_{H2O})$	

Table 3.2 Definitions of the Reaction Energies

#	Change of Enthalpy ($\Delta\mathbf{H}$) in J/mole, $\mathbf{H}$ in J/kg, MW in kg/mole
1	$\Delta H_{H2O} = 2.86 \times 10^5 - 0.5 \cdot H_{O2}(T_{ref}) \cdot MW_{O2} - H_{H2O} \cdot MW_{H2O}$
2	$\Delta H_{Na2O} = 4.16 \times 10^5 - 2 \cdot H_{Na}(T_{ref}) \cdot MW_{Na} - 0.5\ H_{O2} \cdot MW_{O2} + H_{Na2O} \cdot MW_{Na2O}$
3	$\Delta H_{Na2O2} = 5.05 \times 10^5 - 2 \cdot H_{Na}(T_{ref}) \cdot MW_{Na} - H_{O2} \cdot MW_{O2} + H_{Na2O2} \cdot MW_{Na2O2}$
4	$\Delta H_{NaOH} = 4.27 \times 10^5 - H_{Na}(T_{ref}) \cdot MW_{Na} - 0.5\ H_{O2} \cdot MW_{O2} - 0.5\ H_{H2} \cdot MW_{H2} + H_{NaOH} \cdot MW_{NaOH}$

References

1 Grabaskas, D., Brunett, A., Bucknor, M., et.al., Regulatory Technology Development Plan Sodium Fast Reactor, ANL-ART-3, Argonne National Laboratory (February 2015).

2 Tsai, S.S, The NACOM Code for Analysis of Postulated Sodium Spray Fires in LMFBRs, NUREG/CR-1405, Brookhaven National Laboratory, Upton, New York (March 1980).

3 Spalding, D.B., Some Fundamentals of Combustion, Butterworths, London, England (1955).

Noncondensible Gas (NCG) and Water (H2O) Packages

Noncondensible gases in the Control Volume Hydrodynamics (CVH) package are modeled as ideal gases. The constant volume heat capacity is approximated as an analytic function of temperature. The equation of state for water is based on the analytic expression for the Helmholtz function used to generate the familiar Keenan and Keyes Steam Tables. [1] This document describes the constitutive relations used for the water and noncondensible gases equations of state, and it lists the default values of the associated constants for the gases provided in the NCG library.

User input requirements for the NCG package are described in the NCG Users' Guide. There is no input allowed for the H2O package.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. NCG Equation of State

Noncondensible gases in the Control Volume Hydrodynamics (CVH) package are modeled as ideal gases. The specific internal energy and enthalpy of an ideal gas is a function only of its temperature, T, the natural state (reference) temperature, T_n, its energy of formation, e_{form}, its enthalpy of formation, h_{form}, the universal gas constant, R, and its molecular weight, w.

$$e(T) = \int_{T_n}^{T} c_v(T')\ dT' + e_{form} \tag{1-1}$$

$$h(T) = \int_{T_n}^{T} \left(c_v(T') + \frac{R}{w}\right)\ dT' + h_{form} \tag{1-2}$$

The pressure, P, is a function of the mass density, ρ, the temperature, T, the universal gas constant, R, and the molecular weight, w.

$$P = \frac{\rho\ R\ T}{w} \tag{1-3}$$

The noncondensible gases in MELCOR are characterized by the temperature dependent constant volume specific heat, $c_v(T)$, the natural state (reference) temperature, T_n, the energy of formation, e_{form}, the entropy at the reference temperature, s_0 (this quantity is not currently used in the calculation but is included for completeness), and the molecular weight of the material, w.

The specific heat for each noncondensible gas calculated from an analytic fit in the general form

$$c_v(T) = c_{v0} + c_{v1}T + c_{v2}T^2 + c_{v3}T^3 + \frac{c_{vsqrt}}{\sqrt{T}} + \frac{c_{vm1}}{T} + \frac{c_{vm2}}{T^2} \tag{1-4}$$

for the temperature range $T_{low} \le T \le T_{up}$, where T_{low} and T_{up} may be different for each gas. The value at T_{low} is used for $T < T_{low}$ and the value at T_{up} is used for $T > T_{up}$.

Using this constitutive relation for the specific heat, the internal energy is given by

$$\begin{aligned} e(T) = e_0 &+ c_{v0}T + \frac{1}{2}c_{v1}T^2 + \frac{1}{3}c_{v2}T^3 + \frac{1}{4}c_{v3}T^4 \\ &+ 2c_{vsqrt}\sqrt{T} + c_{vm1}\ln(T) - \frac{c_{vm2}}{T} \end{aligned} \tag{1-5}$$

for $T_{low} \le T \le T_{up}$, and is extrapolated outside that range using the constant limiting specific heat at T_{low} or T_{up} are used. Here

$$e_0 = e_{form} - c_{v0}T_n - \frac{1}{2}c_{v1}T_n^2 - \frac{1}{3}c_{v2}T_n^3 - \frac{1}{4}c_{v3}T_n^4 - 2c_{vsqrt}\sqrt{T_n} - c_{vm1}\ln(T_n) + \frac{c_{vm2}}{T_n} \quad (1\text{-}6)$$

Each of the coefficients can be specified via user input, as described in the NCG Users' Guide. Appropriate default coefficients for gases of interest from JANAF [2] and other sources are included in the noncondensible gas equation of state library, as described in Section 2. The default natural temperature used is 298.15 K; this may be changed with sensitivity coefficient 2090.

The reader may note that the definition of e_0 is actually inconsistent unless T_n lies in the range $T_{low} \le T_n \le T_{up}$. For a number of gases (N_2, O_2, CH_4, CO and CO_2), T_{low} is 300 K while T_n is 298.15. In these cases, the discrepancy is less than 10 J/kg and is totally insignificant compared to heats of reaction (several MJ/kg). Although the discrepancy for D_2 (T_{low} = 600 K) is significantly greater, this gas is not used in light water reactor simulations.

1.1 Integration Constants in the Energy Function

A modified thermochemical reference point is used in the NCG package. That is, all heats of formation of compounds are included in the enthalpy functions, as in JANAF tables. The advantage is that all heats of reaction are implicitly contained in the enthalpy functions. For example, in a reaction

$$A + B \rightarrow A \quad (1\text{-}7)$$

taking place at constant temperature and pressure, total enthalpy is conserved. The heat released is the difference between the enthalpy of the reactants and that of the products. This is simply the chemists' definition of the heat of reaction,

$$Q_R(T) = h_A(P,T) + h_B(P,T) - h_{AB}(P,T) \quad (1\text{-}8)$$

Therefore, chemical reactions (such as gas combustion simulated by the Burn package) can be treated simply as changes in the masses of various materials; the associated heat effects are accounted for automatically through the equations of state.

Since only differences in enthalpy are significant, one integration constant may be chosen for each element represented in the collection of gases in the database. Conventional practice is to choose these integration constants such that the enthalpy of each element is zero in its standard state (25°C, 1 atm, with the material in its most stable state). However, water properties in MELCOR are defined (in the H2O package) consistent with Keenan and Keyes Steam Tables [1], as discussed in Section 2. Because water is formed from hydrogen and oxygen, the integration constants for

hydrogen, oxygen, and water may *not* be chosen independently. The conventional integration constant is used for hydrogen in the NCG package, but the integration constant for oxygen has therefore been chosen such that the reference point for water vapor is consistent with that used by Keenan and Keyes. This results in a shift in the integration constant for *every* oxygen-containing gas in the NCG package compared to its conventional JANAF value. For all other gases, the integration constants are consistent with conventional practice.

In actuality, the reference point used is significant only if a gas is chemically active. For current MELCOR models, the only such gases are H_2, D_2, O_2, CO, CO_2, and CH_4. (CH_4 is active *only* if the B_4C reaction in the COR package is enabled, in which case the heat of reaction data used there are not fully compatible with NCG data.) Thus, the user need not worry much about the reference points for other (in particular, user-defined) gases. If chemically active gases are modified, the reference point energy must not be arbitrarily redefined.

2. H2O Equation of State

The equation of state for water is based on the analytic expression for the Helmholtz function, $\psi(\rho,T)$, that was used to generate the familiar Keenan and Keyes Steam Tables [1]. The expression, involving a double power series with log and exponential terms, may be found in the Appendix to the 1969 tables. It contains approximately 50 constant coefficients. These cannot be changed in MELCOR.

The Keenan and Keyes formulation is augmented by JANAF data [2] for temperatures greater than 1589 K (2400 °F). The resulting equation of state is valid for temperatures greater than 273.15 K and for pressures less than 100 MPa.

2.1 Single-Phase Properties

The H2O package determines all single-phase thermodynamic properties of water as functions of density and temperature from the equation for ψ. For example, pressure and internal energy may be expressed in terms of the first derivatives of ψ as

$$P = \rho^2(\partial\psi/\partial\rho)_T \tag{2-1}$$

$$e = \psi + T\ s = \psi - T(\partial\psi/\partial T)_\rho \tag{2-2}$$

where s is entropy. These are evaluated from the equation for ψ and those for its analytic term-by-term derivatives. The quantities $(\partial P/\partial T)_\rho$, $(\partial P/\partial\rho)_T$, and $c_v = (\partial e/\partial T)_\rho$, which involve the three independent second derivatives of ψ, are evaluated similarly.

2.2 Mixed-Phase Properties

The coexistence curve (the saturation line) is defined by points where *P*, *T*, and the Gibbs function $g = \psi + P/\rho$ are equal for two different values of ρ. This curve was determined by a calculation external to MELCOR. All properties of each phase were tabulated at 1 K intervals and are included as data in the H2O package. The properties of two-phase states are evaluated from these tables, using the lever rule.

3. Properties Defined in the Package

The following properties are defined in the package:

	Type	Units
1. **Thermal Conductivity as a function of temperature (THC)**		
a. From tables	Tabular	W/m-K
b. From Eucken correlation and Wassijewa equation	Calculated	W/m-K
c. From equation fit	Calculated	W/m-K
2. **Dynamic Viscosity as a function of temperature**		
a. From tables (VIS)	Tabular	Pa-s
b. From Chapman-Enskog equations (SIG) and Lennard-Jones potential parameters (EPS)	Calculated	Pa-s
c. From equation fit	Calculated	Pa-s
3. **Binary Diffusion Coefficient**		
a. Function of temperature and pressure	Calculated	m^2/s
b. From Chapman-Enskog equations (SIG) and Lennard-Jones potential parameters (EPS)	Calculated	m^2/s
4. **Density**		
a. Function of temperature and pressure	Calculated	kg/m^3

Default values are provided for some, but not all, combinations of materials and physical properties. Section 5 summarizes the default values available. A 'T' indicates that the default function can be changed through user-defined tabular functions and an **NCG_PRP** input record. A 'C' indicates that the default function can be changed through user-defined constant values input on a **NCG_PRP** record. An 'X' indicates that the default function cannot be changed through user input. A blank space indicates that no default is provided, but may be supplied by the user, although in some cases that property for that material may not be used by MELCOR.

Table 3.1 Default material properties, property mnemonics, and user input capabilities.

Property*:	1a	1b	2a	2b	3a	3b	4a
Mnemonic:	THC	SIG EPS	VIS	SIG EPS	n/a	SIG EPS	n/a
WATER	T		T				
STEAM	T	C	T	C		C	X
AIR	T	C	T	C		C	X
H2		C	T	C		C	
HE	T	C	T	C		C	
AR	T	C	T	C		C	
D2		C	T	C		C	
O2	T	C	T	C		C	
CO2		C		C		C	
CO		C		C		C	
N2	T	C	T	C		C	
NO		C		C		C	
N2O		C		C		C	
NH3		C		C		C	
C2H2		C		C		C	
CH4		C		C		C	
C2H4		C		C		C	
STEAM + AIR					X		
STEAM + H2					X		

T - The default function can be changed using tabular functions and an NCG_PRP input record.

C - The default function can be changed using constant values input on a NCG_PRP record.

X - The default function cannot be changed through user input.

Note: A blank space indicates that no default is provided, but may be supplied by the user, although in some cases the property may not be used.

4. Thermal Conductivity as a Function of Temperature

The thermal conductivity may be computed from three different methods. One method, used for structural materials in the COR and HS packages, uses tabular data that may be either a user-specified tabular function or a MELCOR default table. The second method, used for some noncondensible gases and optionally for steam and air, utilizes the Eucken correlation for single, low-pressure gases and the Wassijewa equation for a combination of gases. The third method uses a power law fit for the noncondensible gases He, Air, O_2, N_2, and Ar.

4.1 Tabular

The user-specified tabular function to define a new material or to override the default table for an existing material is invoked by using a standard tabular function to input the thermal conductivity (W/m-K) as a function of temperature (K).

The following materials have default tables for thermal conductivity:

- Water
- Steam
- Air

The default thermal conductivity values are computed by linear interpolation of the tabulated values listed below. Data sources are given with each table.

4.1.1 Water

The default tabular values of thermal conductivity as a function of temperature for liquid water are listed below. No extrapolation is allowed.

Water

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
255.37	0.551	Ref. [3]
273.15	0.569	Ref. [3]
283.15	0.586	Ref. [3]
293.15	0.602	Ref. [3]
303.15	0.617	Ref. [3]
313.15	0.630	Ref. [3]
323.15	0.643	Ref. [3]
333.15	0.653	Ref. [3]
343.15	0.662	Ref. [3]
353.15	0.669	Ref. [3]
363.15	0.675	Ref. [3]
373.15	0.680	Ref. [3]
383.15	0.683	Ref. [3]
393.15	0.685	Ref. [3]
403.15	0.687	Ref. [3]
413.15	0.687	Ref. [3]
423.15	0.686	Ref. [3]
433.15	0.684	Ref. [3]
443.15	0.681	Ref. [3]
453.15	0.676	Ref. [3]
463.15	0.671	Ref. [3]
473.15	0.664	Ref. [3]

Water

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
483.15	0.657	Ref. [3]
493.15	0.648	Ref. [3]
503.15	0.639	Ref. [3]
513.15	0.629	Ref. [3]
523.15	0.617	Ref. [3]
533.15	0.604	Ref. [3]
543.15	0.589	Ref. [3]
553.15	0.573	Ref. [3]
563.15	0.557	Ref. [3]
573.15	0.540	Ref. [3]
583.15	0.522	Ref. [3]
593.15	0.503	Ref. [3]
603.15	0.482	Ref. [3]
613.15	0.460	Ref. [3]
623.15	0.435	Ref. [3]
633.15	0.401	Ref. [3]
647.245	0.318	Ref. [3], extrapolated

4.1.2 Steam

The default tabular values of thermal conductivity as a function of temperature for steam are listed below. Constant extrapolation is allowed from the upper end of the tabulated range. No extrapolation is allowed from the lower end of the tabulated range.

Steam

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
255.37	0.0144	Ref. [3]
273.15	0.0176	Ref. [3]
293.15	0.0188	Ref. [3]
313.15	0.0201	Ref. [3]
333.15	0.0216	Ref. [3]
353.15	0.0231	Ref. [3]
373.15	0.0245	Ref. [3]
393.15	0.0260	Ref. [3]
413.15	0.0277	Ref. [3]
433.15	0.0295	Ref. [3]
453.15	0.0313	Ref. [3]
473.15	0.0331	Ref. [3]
493.15	0.0351	Ref. [3]
513.15	0.0371	Ref. [3]
533.15	0.0391	Ref. [3]
553.15	0.0412	Ref. [3]

Steam

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
573.15	0.0433	Ref. [3]
593.15	0.0455	Ref. [3]
613.15	0.0478	Ref. [3]
633.15	0.0501	Ref. [3]
653.15	0.0525	Ref. [3]
673.15	0.0548	Ref. [3]
693.15	0.0573	Ref. [3]
713.15	0.0597	Ref. [3]
733.15	0.0622	Ref. [3]
753.15	0.0648	Ref. [3]
773.15	0.0673	Ref. [3]
793.15	0.0699	Ref. [3]
813.15	0.0725	Ref. [3]
833.15	0.0752	Ref. [3]
853.15	0.0778	Ref. [3]
873.15	0.0805	Ref. [3]
893.15	0.0832	Ref. [3]
913.15	0.0859	Ref. [3]
933.15	0.0887	Ref. [3]
953.15	0.0914	Ref. [3]
973.15	0.0942	Ref. [3]
993.15	0.0970	Ref. [3]
1013.15	0.0998	Ref. [3]
1033.15	0.1026	Ref. [3]
1053.15	0.1054	Ref. [3]
1073.15	0.1081	Ref. [3]
1200.00	0.130	Ref. [3]
1400.00	0.187	Ref. [3]
1600.00	0.219	Ref. [3]
1800.00	0.263	Ref. [3]
2000.00	0.333	Ref. [3]
2200.00	0.459	Ref. [3]
2400.00	0.690	Ref. [3]
2600.00	1.110	Ref. [3]
2800.00	1.820	Ref. [3]
3000.00	2.940	Ref. [3]
3200.00	4.495	Ref. [3]
3400.00	6.625	Ref. [3]
3600.00	7.610	Ref. [3]
3800.00	7.765	Ref. [3]
4000.00	7.280	Ref. [3]

4.1.3 Air

Tabular values of thermal conductivity as a function of temperature for air are listed below. This table was used in early versions of 2.1 as well as 1.8.6 and can be imposed by enabling 1.8.6 defaults through the EXEC_DEFAULT_GLOBAL. Linear extrapolation is allowed from the upper end of the tabulated range. No extrapolation is allowed from the lower end of the tabulated range.

Air

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
255.370	0.0227081	Ref. [4]
310.926	0.0270005	Ref. [4]
366.482	0.0311544	Ref. [4]
422.038	0.0360006	Ref. [4]
477.594	0.0399815	Ref. [4]
533.150	0.0425777	Ref. [4]
588.706	0.0458662	Ref. [4]
644.262	0.0491547	Ref. [4]
699.818	0.0524432	Ref. [4]
755.374	0.0553856	Ref. [4]
810.930	0.0583280	Ref. [4]
866.486	0.0610972	Ref. [4]
922.042	0.0638665	Ref. [4]
977.598	0.0664627	Ref. [4]
1033.154	0.0690589	Ref. [4]
1088.710	0.0718282	Ref. [4]
1144.266	0.0740782	Ref. [4]
1199.822	0.0763283	Ref. [4]
1255.378	0.0785783	Ref. [4]
1310.934	0.0808284	Ref. [4]
1366.490	0.0830784	Ref. [4]
1422.046	0.0853284	Ref. [4]
1477.602	0.0874054	Ref. [4]
1533.158	0.0896554	Ref. [4]
1588.714	0.0920786	Ref. [4]
1644.270	0.0941555	Ref. [4]
1699.826	0.0960594	Ref. [4]
1755.382	0.0979633	Ref. [4]
1810.938	0.0998672	Ref. [4]
1866.494	0.101425	Ref. [4]
1922.050	0.103156	Ref. [4]
1977.606	0.105060	Ref. [4]
2033.162	0.107137	Ref. [4]
2088.718	0.109040	Ref. [4]

Air

Temp (K)	Thermal Conductivity (W/m-K)	Data Source
2144.274	0.110425	Ref. [4]
2199.830	0.111810	Ref. [4]
2255.386	0.113367	Ref. [4]
2310.942	0.115098	Ref. [4]
2366.498	0.116829	Ref. [4]
2422.054	0.113367	Ref. [4]
2477.610	0.120118	Ref. [4]
2533.166	0.121675	Ref. [4]

4.2 Eucken Correlation for a Single, Pure Gas

The thermal conductivity, λ_i, of a single low-pressure gas may be computed using the Eucken correlation [5]:

$$\lambda_i = \left(C_{vi} + \frac{9R}{4M_i}\right)\mu_i \qquad (W/m-K) \tag{4-1}$$

where,

C_{vi} = heat capacity at constant volume (J/kg-K), calculated by the NCG package

R = universal gas constant, 8.31441 J/mol-K

μ_i = viscosity (kg/m-s)

M_i = molecular weight (kg/mol)

4.3 Wassijewa Equation for a Combination of Low-Pressure Gases

The thermal conductivity, λ_{mix}, of a combination of gases may be computed using the Wassijewa equation with the Mason and Saxena modification for the A_{ij} term [5]:

$$\lambda_{mix} = \sum_{i=1}^{n} \frac{\lambda_i \; y_i}{\sum_{j=1}^{n} y_j \; A_{ij}} \tag{4-2}$$

where,

y_i = mole fraction of gas i

λ_i = thermal conductivity of pure gas i (see Section 4.2)

$$A_{ij} = \frac{\left[1 + (\mu_i/\mu_j)^{1/2}(M_j/M_i)^{1/4}\right]^2}{\left[8(1 + M_i/M_j)\right]^{1/2}}$$
$$= \frac{1}{\sqrt{8}}\left(\frac{M_j}{M_i}\right)\left(\frac{M_i}{M_j + M_i}\right)^{1/2}\left[\left(\frac{M_i}{M_j}\right)^{1/4} + \left(\frac{\mu_i}{\mu_j}\right)^{1/2}\right]^2$$

μ_i = viscosity of pure gas *i* (kg/m-s)

M_i = molecular weight of gas *i* (kg/mol)

The mole fractions, y_i, may be expressed in terms of the gas masses, m_i, using,

$$y_i = \frac{m_i / M_i}{\sum_{i=1}^{n} m_j / M_j} \tag{4-3}$$

yielding,

$$\lambda_{mix} = \sum_{i=1}^{n} \frac{m_i \lambda_i}{\sum_{j=1}^{n} m_j \left(\frac{M_i}{M_i}\right) A_{ij}} \tag{4-4}$$

or

$$\lambda_{mix} = \sqrt{8}\sum_{i=1}^{n} \frac{m_i \lambda_i}{\sum_{j=1}^{n} m_j \left(\frac{M_i}{M_j + M_i}\right)^{1/2}\left[\left(\frac{M_i}{M_j}\right)^{1/4} + \left(\frac{\mu_i}{\mu_j}\right)^{1/2}\right]^2} \tag{4-5}$$

4.4 Equation Fit

Some noncondensible gases use an equation fit to data as the default. This is generally more accurate at higher temperatures than using the Chapman-Enskog equation when considering air-graphite reactions in high temperature gas reactor. This was implemented as default in later versions of MELCOR 2.1. The gases using equation fits are listed below:

- Helium
- Air
- Nitrogen
- Oxygen
- Argon

For all these cases except Helium, a power law fit is used of the form

$$k = AT^B$$

where

k = thermal conductivity (W/m-K),

A = lead coefficient,

B = exponent

The fits are done by applying a linear least squares procedure to the log of the data in the form

$$ln\, y = ln\, A + B\, ln\, T$$

A power law fit of this form is accurate for gas thermal conductivity in the “dilute region”, meaning that the ideal gas law is also applicable, and corrections due to being near the triple point or at extreme high pressures are not necessary.

The thermal conductivity of Helium is modeled with a fit from the KTA Rules [15]:

$$k = 2.682x10^{-3}(1 + 1.123x10^{-8}P)T^{0.71}(1 - 2x10^{-9}P)$$

where

k = thermal conductivity (W/m-K)

P = pressure (Pa)

T = temperature (K)

There is a small variation of thermal conductivity over the range 0.1 MPa to 10 MPa, our range of interest for HTGRs, but not enough to be significant. MELCOR coding presently allows thermal conductivity as a function of temperature only, so the KTA formula is used at a pressure of 0.1 MPa.

Values for the parameters in the fits are shown in Table 4.1. The data sources are also listed in the table.

Table 4.1 Parameters for Gas Thermal Conductivity Power Law Fits

Gas	A (W/m-K)	B	Data Source
Helium	2.685e-3	0.71	Ref.15
Nitrogen	3.704689e-4	0.74842	Ref.16
Oxygen	2.810152e-4	0.80107	Ref.16
Argon	3.518418e-4	0.69561	Ref.16
Air	3.418146e-4	0.76512	Ref.16

Thermal conductivity for gas mixtures is done using the Wassijewa method as done for mixtures of pure gas conductivities calculated using the Eucken equation.

5. Dynamic Viscosity as a Function of Temperature

The dynamic viscosity may be computed from three different methods. One method, used for structural materials in the COR and HS packages, utilizes tabular data that may be either a user-specified tabular function or a MELCOR default table. The second method, used for some noncondensible gases and optionally for steam and air, utilizes the Chapman-Enskog equations for low-pressure gases based on constant Lennard-Jones potential parameters, σ and ε / k, which may be either user-specified or MELCOR default values. The third method uses a power law fit for the noncondensible gases He, Air, O_2, N_2, and Ar.

5.1 Tabular

The user-specified tabular function to define a new material or to override the default table for an existing material is invoked by using a standard tabular function (see the TF Package Users' Guide) to input the viscosity (kg/m-s) as a function of temperature (K).

The following materials have default tables for viscosity:

- Water
- Steam
- Air
- Hydrogen
- Deuterium

The default viscosity values are computed by linear interpolation of the tabulated values listed below. Data sources are given with each table.

5.1.1 Water

The default tabular values of dynamic viscosity as a function of temperature for liquid water are listed below. No extrapolation is allowed.

Water

Temp	Dynamic Viscosity (kg/m-s)	Data Source
255.370	0.00264402	Ref. [6]
283.148	0.00130962	Ref. [6]
310.926	0.000681596	Ref. [6]
338.704	0.000434554	Ref. [6]
366.482	0.000305081	Ref. [6]
394.260	0.000235136	Ref. [6]
422.038	0.000186025	Ref. [6]
449.816	0.000156261	Ref. [6]
477.594	0.000135426	Ref. [6]
499.850	0.000117267	Ref. [6]
522.050	0.000106999	Ref. [6]
544.250	0.0000985165	Ref. [6]
566.450	0.0000915221	Ref. [6]
588.750	0.0000833372	Ref. [6]
610.950	0.0000723248	Ref. [6]
633.150	0.0000581872	Ref. [6]
647.245	0.0000492111	Ref. [7], p. 103

5.1.2 Steam

The default tabular values of dynamic viscosity as a function of temperature for steam are listed below. Linear extrapolation is allowed from the upper end of the tabulated range. No extrapolation is allowed from the lower end of the tabulated range.

Steam

Temp	Dynamic Viscosity (kg/m-s)	Data Source
255.15	0.00000724	Ref. [8]
273.15	0.00000804	Ref. [8]
313.15	0.00000966	Ref. [8]
353.15	0.0000113	Ref. [8]
393.15	0.0000129	Ref. [8]
433.15	0.0000146	Ref. [8]
473.15	0.0000162	Ref. [8]
513.15	0.0000178	Ref. [8]
553.15	0.0000194	Ref. [8]
593.15	0.0000211	Ref. [8]
633.15	0.0000227	Ref. [8]

Steam

Temp	Dynamic Viscosity (kg/m-s)	Data Source
673.15	0.0000243	Ref. [8]
713.15	0.0000260	Ref. [8]
753.15	0.0000276	Ref. [8]
793.15	0.0000292	Ref. [8]
833.15	0.0000308	Ref. [8]
873.15	0.0000325	Ref. [8]
913.15	0.0000341	Ref. [8]
953.15	0.0000357	Ref. [8]
993.15	0.0000375	Ref. [8]
1033.15	0.0000391	Ref. [8]
1073.15	0.0000406	Ref. [8]
1200.00	0.0000454	Ref. [8]
1400.00	0.0000512	Ref. [8]
1600.00	0.0000563	Ref. [8]
1800.00	0.0000612	Ref. [8]
2000.00	0.0000659	Ref. [8]
2200.00	0.0000703	Ref. [8]
2400.00	0.0000742	Ref. [8]
2600.00	0.0000775	Ref. [8]
2800.00	0.0000798	Ref. [8]
3000.00	0.0000810	Ref. [8]
3200.00	0.0000814	Ref. [8]
3400.00	0.0000816	Ref. [8]
3600.00	0.0000825	Ref. [8]
3800.00	0.0000851	Ref. [8]
4000.00	0.0000895	Ref. [8]

5.1.3 Air

Tabular values of dynamic viscosity as a function of temperature for air are listed below. This table was used in early versions of 2.1 as well as 1.8.6 and can be imposed by enabling 1.8.6 defaults through the EXEC_DEFAULT_GLOBAL. Linear extrapolation is allowed from the upper end of the tabulated range. No extrapolation is allowed from the lower end of the tabulated range.

Air

Temp (K)	Dynamic Viscosity (kg/m-s)	Data Source
99.820	0.00000852739	Ref. [8]
299.820	0.0000184686	Ref. [8]
499.820	0.0000267132	Ref. [8]
699.820	0.0000333208	Ref. [8]
899.820	0.0000389908	Ref. [8]

1099.820	0.0000439763	Ref. [8]
1299.820	0.0000484856	Ref. [8]
1499.820	0.0000525781	Ref. [8]
1699.820	0.0000564325	Ref. [8]
1899.820	0.0000599596	Ref. [8]
2099.820	0.0000640075	Ref. [8]
2299.820	0.0000671625	Ref. [8]
2499.820	0.0000698561	Ref. [8]
2699.820	0.0000723414	Ref. [8]

5.1.4 Hydrogen

The default tabular values of dynamic viscosity as a function of temperature for hydrogen are listed below. Linear extrapolation is allowed from the upper end of the tabulated range. No extrapolation is allowed from the lower end of the tabulated range.

Hydrogen

Temp	Dynamic Viscosity (kg/m-s)	Data Source
100.0	0.0000042105	Ref. [9], p.284
200.0	0.0000068129	Ref. [9], p.284
250.0	0.0000079232	Ref. [9], p.284
280.0	0.0000085523	Ref. [9], p.284
300.0	0.0000089594	Ref. [9], p.284
400.0	0.000010867	Ref. [9], p.284
500.0	0.000012642	Ref. [9], p.284
600.0	0.000014290	Ref. [9], p.284
700.0	0.000015846	Ref. [9], p.284
800.0	0.000017335	Ref. [9], p.284
900.0	0.000018756	Ref. [9], p.284
1000.0	0.000020128	Ref. [9], p.284
1100.0	0.000021440	Ref. [9], p.284
1200.0	0.000022754	Ref. [10]
1300.0	0.000024078	Ref. [10]
4000.0	0.000059839	Ref. [10], extrapolated

5.1.5 Deuterium

The default tabular values of dynamic viscosity as a function of temperature for deuterium are listed below. No extrapolation is allowed from the lower end of the tabulated range. Linear extrapolation is allowed from the upper end of the tabulated range.

Deuterium

Temp	Dynamic Viscosity (kg/m-s)	Data Source
100.0	0.00000579	Ref. [11]
120.0	0.00000662	Ref. [11]
140.0	0.00000739	Ref. [11]
160.0	0.00000814	Ref. [11]
180.0	0.00000885	Ref. [11]
200.0	0.00000955	Ref. [11]
220.0	0.00001022	Ref. [11]
240.0	0.00001087	Ref. [11]
260.0	0.00001151	Ref. [11]
280.0	0.00001214	Ref. [11]
300.0	0.00001274	Ref. [11]
320.0	0.00001332	Ref. [11]
340.0	0.00001388	Ref. [11]
360.0	0.00001445	Ref. [11]
380.0	0.00001501	Ref. [11]
400.0	0.00001554	Ref. [11]
420.0	0.00001606	Ref. [11]
440.0	0.00001658	Ref. [11]
460.0	0.00001709	Ref. [11]
480.0	0.00001758	Ref. [11]
500.0	0.00001805	Ref. [11]

5.2 Chapman-Enskog Equation for a Single, Pure Gas

$$\mu_i = 2.6693x10^{-6}\frac{\sqrt{1000\ M\ T}}{\sigma^2\Omega_v} \qquad kg/m-s \tag{5-1}$$

The viscosity, μ_i, of a single, low-pressure gas may be computed using the Chapman-Enskog viscosity equation [12]:

where,

M = molecular weight (kg/mol)

T = gas temperature (K)

σ = collision diameter (Å ≡ 10^{-10}m)

Ω_v = collision integral

$$= \quad 2.785\left(\frac{T^*}{0.3}\right)^{-0.4} \qquad T^* < 0.3 \text{ (extrapolated)}$$

$$= \quad f(T^*) \text{ from Table 5.1 below} \qquad 0.3 \le T^* < 100$$

$$= \quad 0.5882\left(\frac{T^*}{100}\right)^{-0.145} \qquad T^* \ge 100$$

$$T^* = \frac{k\,T}{\varepsilon}$$

ε / k = characteristic energy/Boltzmann's constant (K)

The following materials have default tables for the Lennard-Jones potential parameters, σ and ε / k [13,14]:

	σ (Å)	ε / k (K)
Steam	2.641	809.1
Air	3.711	78.6
Hydrogen	2.827	59.7
Helium	2.551	10.22
Argon	3.542	93.3
Deuterium	2.948	39.3
Oxygen	3.467	106.7
Carbon Dioxide	3.941	195.2
Carbon Monoxide	3.690	91.7
Nitrogen	3.798	71.4
Nitric Oxide	3.492	116.7
Nitrous Oxide	3.828	232.4
Ammonia	2.900	558.3
Acetylene	4.033	231.8
Methane	3.758	148.6
Ethylene	4.163	224.7

The default values for σ and ε / k may be changed using the mnemonics SIG and EPS as described in the NCG Users' Guide.

Table 5.1 Collision Integral, Ω_v, as a Function of the Dimensionless Temperature, T^* [12].

T^*	Ω_v	T^*	Ω_v	T^*	Ω_v
0.30	2.785	1.65	1.264	4.00	0.9700
0.35	2.628	1.70	1.248	4.10	0.9649
0.40	2.492	1.75	1.234	4.20	0.9600
0.45	2.368	1.80	1.221	4.30	0.9553
0.50	2.257	1.85	1.209	4.40	0.9507
0.55	2.156	1.90	1.197	4.50	0.9464
0.60	2.065	1.95	1.186	4.60	0.9422
0.65	1.982	2.00	1.175	4.70	0.9382
0.70	1.908	2.10	1.156	4.80	0.9343
0.75	1.841	2.20	1.138	4.90	0.9305
0.80	1.780	2.30	1.122	5.00	0.9269
0.85	1.725	2.40	1.107	6.00	0.8963
0.90	1.675	2.50	1.093	7.00	0.8727
0.95	1.629	2.60	1.081	8.00	0.8538
1.00	1.587	2.70	1.069	9.00	0.8379
1.05	1.549	2.80	1.058	10.00	0.8242
1.10	1.514	2.90	1.048	20.00	0.7432
1.15	1.482	3.00	1.039	30.00	0.7005
1.20	1.452	3.10	1.030	40.00	0.6718
1.25	1.424	3.20	1.022	50.00	0.6504
1.30	1.399	3.30	1.014	60.00	0.6335
1.35	1.375	3.40	1.007	70.00	0.6194
1.40	1.353	3.50	0.9999	80.00	0.6076
1.45	1.333	3.60	0.9932	90.00	0.5973
1.50	1.314	3.70	0.9870	100.00	0.5882
1.55	1.296	3.80	0.9811		
1.60	1.279	3.90	0.9755		

5.3 Chapman-Enskog Equation for a Combination of Low-Pressure Gases

The viscosity of a mixture of gases can be computed by combining the individual viscosities of the pure substances using the following equation with the Wilkes approximation for the term, φ_{ij} [13]

$$\mu_{mix} = \sum_{i=1}^{n} \frac{y_i \mu_i}{\sum_{j=1}^{n} y_j \varphi_{ij}} \tag{5-2}$$

where,

y_i = mole fraction of gas *i*

μ_i = viscosity of pure gas *i* (see Section 5.2)

$$\varphi_{ij} = \frac{\left[1+(\mu_i/\mu_j)^{1/2}(M_j/M_i)^{1/4}\right]^2}{\left[8(1+M_i/M_j)\right]^{1/2}}$$

$$= \frac{1}{\sqrt{8}}\left(\frac{M_j}{M_i}\right)\left(\frac{M_i}{M_j+M_i}\right)^{1/2}\left[\left(\frac{M_i}{M_j}\right)^{1/4}+\left(\frac{\mu_i}{\mu_j}\right)^{1/2}\right]^2$$

M_i = molecular weight of gas *i* (kg/mol), set by the NCG package

The mole fractions, y_i, may be expressed in terms of the gas masses, m_i, using,

$$y_i = \frac{m_i / M_i}{\sum_{k=1}^{n} m_j / M_j} \tag{5-3}$$

yielding,

$$\mu_{mix} = \sum_{i=1}^{n} \frac{m_i \mu_i}{\sum_{j=1}^{n} m_j \left(\frac{M_i}{M_j}\right)\varphi_{ij}} \tag{5-4}$$

or

$$\mu_{mix} = \sqrt{8}\sum_{i=1}^{n} \frac{m_i \mu_i}{\sum_{j=1}^{n} m_j \left(\frac{M_i}{M_j + M_i}\right)^{1/2} \left[\left(\frac{M_i}{M_j}\right)^{1/4} + \left(\frac{\mu_i}{\mu_j}\right)^{1/2}\right]^2} \qquad (5\text{-}5)$$

5.4 Equation Fit

Some noncondensible gases use an equation fit to data as the default. This is generally more accurate at higher temperatures than using the Chapman-Enskog equation when considering air-graphite reactions in high temperature gas reactor. This was implemented as default in later versions of MELCOR 2.1. The gases using equation fits are listed below:

- Helium
- Air
- Nitrogen
- Oxygen
- Argon

For all these cases, a power law fit is used of the form

$$\mu = AT^B$$

where

μ = viscosity (Pa-s),

A = lead coefficient,

B = exponent

The fits are done by applying a linear least squares procedure to the log of the data in the form

$$ln\, y = ln\, A + B\, ln\, T$$

A power law fit of this form is accurate for gas viscosity in the “dilute region”, meaning that the ideal gas law is also applicable, and corrections due to being near the triple point or at extreme high pressures are not necessary.

Values for the parameters in the fits are shown in Table 5.2. The data sources are also listed in the table.

Table 5.2 Values for Parameters in Power Law Fits

Gas	A (Pa-s)	B	Data Source
Helium	3.674e-7	0.7	Ref.15
Nitrogen	3.9539785e-7	0.67288	Ref.16
Oxygen	4.3304788e-7	0.68343	Ref.16
Argon	4.3908105-7	0.69910	Ref.16
Air	4.0554513e-7	0.67501	Ref.16

The viscosity for gas mixtures is obtained using the Wilkes method, as done for pure gas viscosities calculated from the Chapman-Enskog equation.

6. Binary Mass Diffusion Coefficient

The binary diffusion coefficients are computed using two different methods depending on which MELCOR package requires the information. The diffusion coefficients required for COR, CVH, and HS packages are computed by the MP package using the equations given in Section 6.1, below. RN1 utilizes the MP package noncondensible gas Lennard-Jones parameters for the calculation of fission product vapor binary diffusion coefficients as described in Section 6.2.

6.1 Binary Mass Diffusion Coefficient as a Function of Temperature and Pressure

The diffusion coefficient is computed from different correlations for each pair of materials. The diffusion coefficient (m^2/s) is defined as a function of temperature (K) and pressure (Pa) for two pairs of materials.

For steam and air, the following correlation is used (origin unknown):

$$D = 4.7931x10^{-5}\left(\frac{T^{1.9}}{P}\right) \tag{6-1}$$

For steam and hydrogen, the correlation is taken from Reference [17]:

$$D = 6.60639x10^{-4}\left(\frac{T^{1.68}}{P}\right) \tag{6-2}$$

An error message is printed if the input temperature or pressure is less than zero. There is currently no means by which the user can change these correlations.

6.2 Chapman-Enskog Equation for a Pair of Low-Pressure Gases

The binary diffusion coefficient, D_{AB}, for a pair of low-pressure gases may be computed using the Chapman-Enskog equation [12]:

$$D_{AB} = 1.88292x10^{-2} \frac{\sqrt{T^3\left(\frac{0.001}{M_A}+\frac{0.001}{M_B}\right)}}{P\sigma_{AB}^2\Omega_{D,AB}} \quad m^2/s \qquad (6\text{-}3)$$

where,

M_A = molecular weight of gas A (kg/mol)

M_B = molecular weight of gas B (kg/mol)

T = gas temperature (K)

P = gas pressure (Pa)

σ_A = collision diameter of gas A (Å ≡ 10^{-10}m)

σ_B = collision diameter of gas B (Å ≡ 10^{-10}m)

σ_{AB} = effective collision diameter of gas *A* and *B* (Å ≡ 10^{-10}m)

= $1/2\left(\sigma_A+\sigma_B\right)$

$\Omega_{D,AB}$ = collision integral

= $2.662\ \left(T_{AB}^*/0.3\right)^{-0.5}$ $T_{AB}^* < 0.3$ (extrapolated)

= $f\left(T_{AB}^*\right)$ from Table 6.1 below $0.3 \le T_{AB}^* < 100$

= $0.5170\left(T_{AB}^*/100\right)^{-0.155}$ $T_{AB}^* \ge 100$ (extrapolated)

T_{AB}^* = $\left(k\,T/\varepsilon_{AB}\right)$

ε_{AB}/k = effective characteristic energy/Boltzmann's constant for gas *A* and *B* (K)

= $1/k\left(\varepsilon_A\varepsilon_B\right)^{1/2}$

ε_A/k = effective characteristic energy/Boltzmann's constant for gas *A* (K)

ε_B/k = effective characteristic energy/Boltzmann's constant for gas *B* (K)

The table of Lennard-Jones potential parameters, σ and ε / k, is given in Section 5.2. The default values for σ and ε / k may be changed using the mnemonics SIG and EPS as described in the MP Users' Guide.

Table 6.1 Collision Integral, Ω_D, as a Function of Dimensionless Temperature, T_{AB}^* [12].

T_{AB}^*	Ω_D	T_{AB}^*	Ω_D	T_{AB}^*	Ω_D
0.30	2.662	1.65	1.153	4.00	0.8836
0.35	2.476	1.70	1.140	4.10	0.8788
0.40	2.318	1.75	1.128	4.20	0.8740
0.45	2.184	1.80	1.116	4.30	0.8694
0.50	2.066	1.85	1.105	4.40	0.8652
0.55	1.966	1.90	1.094	4.50	0.8610
0.60	1.877	1.95	1.084	4.60	0.8568
0.65	1.798	2.00	1.075	4.70	0.8530
0.70	1.729	2.10	1.057	4.80	0.8492
0.75	1.667	2.20	1.041	4.90	0.8456
0.80	1.612	2.30	1.026	5.00	0.8422
0.85	1.562	2.40	1.012	6.00	0.8124
0.90	1.517	2.50	0.9996	7.00	0.7896
0.95	1.476	2.60	0.9878	8.00	0.7712
1.00	1.439	2.70	0.9770	9.00	0.7556
1.05	1.406	2.80	0.9672	10.00	0.7424
1.10	1.375	2.90	0.9576	20.00	0.6640
1.15	1.346	3.00	0.9490	30.00	0.6232
1.20	1.320	3.10	0.9406	40.00	0.5960
1.25	1.296	3.20	0.9328	50.00	0.5756
1.30	1.273	3.30	0.9256	60.00	0.5596
1.35	1.253	3.40	0.9186	70.00	0.5464
1.40	1.233	3.50	0.9120	80.00	0.5352
1.45	1.215	3.60	0.9058	90.00	0.5256
1.50	1.198	3.70	0.8998	100.00	0.5170
1.55	1.182	3.80	0.8942		
1.60	1.167	3.90	0.8888		

6.3 Chapman-Enskog Equation for a Combination of Low-Pressure Gases

The effective binary diffusion coefficient, D_{im}, for gas i in a mixture of m low-pressure gases can be computed as [12]:

$$\frac{1-y_i}{D_{im}} = \sum_{j=1,\neq i}^{m} \left(\frac{y_j}{D_{ij}}\right) \tag{6-4}$$

where,

y_i = mole fraction of gas i, and

D_{ij} = binary diffusion coefficient for gas pair *ij* (m^2/s).

7. Density

The density of most materials may be computed as a constant value, a user-specified tabular function or a MELCOR default table. The default function for the densities of air and steam, however, are fixed by the code and cannot be changed through user input.

7.1 Calculated as a Function of Temperature and Pressure

The default density functions for air and steam are described in Sections 7.1.1 and 7.1.2, below. These default functions may not be altered through user input.

7.1.1 Air

The density (kg/m^3) of air is computed from the gas law:

$$\rho = MW \ x \ Pr\,e\,s/(R \ x \ T \ x \ CPRS) \tag{7-1}$$

where,

MW = Molecular weight, 0.028966 kg/mol

Pres = Pressure (Pa)

R = Universal gas constant, 8.31441 J/(mol-K)

T = Temperature (K)

CPRS = Compressibility, 1.0

7.1.2 Steam

The density (kg/m^3) of steam is computed from the gas law:

$$\rho = MW\ x\ Pr\,e\,s/(R\ x\ T\ x\ CPRS) \tag{7-2}$$

where,

MW = Molecular Weight, 0.018016 kg/mol

Pres = Pressure (Pa)

R = Universal gas constant, 8.31441 J/(mol-K)

T = Temperature (K)

CPRS = Given in Table 5.1

The value of CPRS is determined by standard interpolation on T and P for those points bounded by values from Table 5.1. For those points that lie outside the bounds of the table, various methods are used for determining CPRS. Figure 7.1 is a graphic illustration of the values shown on Table 5.1. The figure is divided into 10 regions, each of which has its own method for computing the compressibility.

Region 1: Points in this region are assigned a compressibility of 0.9978. This corresponds to the value of CPRS at (0.0068884 MPa, 311.72 K).

Region 2: CPRS for points in this region are computed by linear interpolation on temperature of the values for the pressure, P = 0.0068884 MPa.

Region 3: Points in this region are assigned a compressibility of 1.0000. This corresponds to the value of CPRS at (0.0068884 MPa, 1033.0 K).

Region 4: CPRS for points in this region are computed by linear interpolation on pressure of the smallest values for the pressures on the left and right sides of (P, T).

Region 5: CPRS for points in this region are computed by linear interpolation, first on temperature, then on pressure, of the bounding values on the left side and the value corresponding to the minimum temperatures on the right side.

Region 6: CPRS for points in this region are computed by linear interpolation, first on temperature, then on pressure, of the bounding values.

Region 7: Points in this region are assigned the maximum value for compressibility, 1.0000.

Region 8: Points in this region are assigned a compressibility of 0.9134. This corresponds to the value of CPRS at (1.3786 MPa, 467.37 K).

Region 9: CPRS for points in this region are computed by linear interpolation on temperature of the values for the pressure, P = 1.3786 MPa.

Region 10: Points in this region are assigned a compressibility of 0.9995. This corresponds to the value at (1.3786 MPa, 1366.33 K).

Table 7.1 Compressibility of Steam as a Function of Temperature (K) and Pressure (MPa) (Ref. 8).

Pressure (MPa)	**0.0068884**	**0.034462**	**0.068953**	**0.10130**	**0.13786**	**0.27572**	**0.41358**
Temp (K)							
311.72	0.9978						
345.34		0.9927					
362.55			0.9881				
366.33	0.9991	0.9946	0.9889				
372.99				0.9846			
381.87					0.9811		
388.55	0.9993	0.9959	0.9916				
388.56				0.9875	0.9825		
403.70						0.9702	
410.77	0.9995	0.9969	0.9936				
410.78				0.9905	0.9866	0.9726	
417.85							0.9610
428.59							
433.00	0.9997	0.9976	0.9950	0.9925	0.9898	0.9786	0.9672
437.37							
444.83							
451.38							
455.22	0.9998	0.9981	0.9960	0.9941	0.9919	0.9830	0.9739
457.22							
462.52							
467.37							
477.44	0.9999	0.9985	0.9967	0.9952	0.9934	0.9862	0.9789
499.67	1.0000	0.9988	0.9974	0.9959	0.9944	0.9886	0.9826
505.22							
533.00	1.0000	0.9991	0.9980	0.9969	0.9959	0.9913	0.9867
560.78							
588.56	1.0000	0.9995	0.9987	0.9981	0.9973	0.9941	0.9911
644.11	1.0000	0.9996	0.9992	0.9986	0.9982	0.9959	0.9938
699.67	1.0000	0.9998		0.9991	0.9987	0.9971	0.9956
755.22							
810.78	1.0000	1.0000	0.9998	0.9996	0.9994	0.9985	0.9976
921.89	1.0000	1.0000	1.0000	0.9999	0.9998	0.9992	0.9987
1033.00	1.0000	1.0000	1.0000	1.0000	0.9999	0.9997	0.9992
1144.11				1.0000	1.0000	0.9998	0.9998
1255.22						1.0000	1.0000
1366.33							1.0000
1477.44							

Table 7.1 Compressibility of Steam as a Function of Temperature (K) and Pressure (MPa) (Ref. 8). (continued)

Pressure (MPa)	0.55145	0.68931	0.82717	0.96502	1.1029	1.2407	1.3786
Temp (K)							
311.72							
345.34							
362.55							
366.33							
372.99							
381.87							
388.55							
388.56							
403.70							
410.77							
410.78							
417.85							
428.59	0.9528						
433.00	0.9552						
437.37		0.9432					
444.83			0.9383				
451.38				0.9316			
455.22	0.9646	0.9550	0.9449	0.9347			
457.22					0.9255		
462.52						0.9193	
467.37							0.9134
477.44	0.9714	0.9637	0.9561	0.9478	0.9397	0.9310	0.9223
499.67	0.9766	0.9703					
505.22			0.9657	0.9595	0.9533	0.9469	0.9406
533.00	0.9822	0.9775	0.9727	0.9681	0.9630	0.9580	0.9532
560.78			0.9779	0.9741	0.9701	0.9663	0.9622
588.56	0.9880	0.9850	0.9819	0.9787	0.9756	0.9723	0.9691
644.11	0.9916	0.9895	0.9872	0.9852	0.9830	0.9807	0.9785
699.67	0.9940	0.9924	0.9909	0.9893	0.9877	0.9861	0.9845
755.22						0.9899	0.9887
810.78	0.9967	0.9958	0.9950	0.9941	0.9933	0.9924	0.9916
921.89	0.9982	0.9976	0.9971	0.9966	0.9962	0.9956	0.9951
1033.00	0.9990	0.9986	0.9983	0.9980	0.9977	0.9974	0.9970
1144.11	0.9994	0.9992	0.9990	0.9989	0.9986	0.9984	0.9983
1255.22	0.9998	0.9996	0.9995	0.9994	0.9993	0.9991	0.9990
1366.33	1.0000	0.9999	0.9998	0.9998	0.9997	0.9996	0.9995
1477.44		1.0000			1.0000		

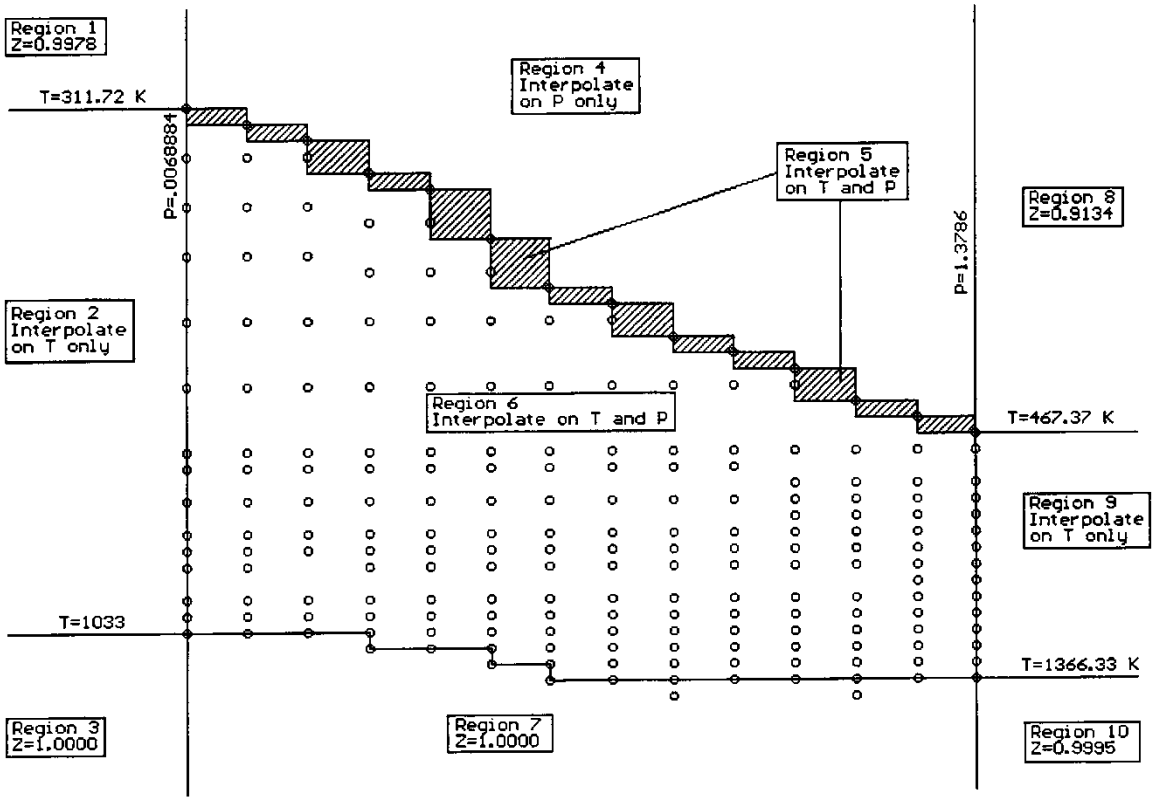


Figure 7.1 Methods for Computing the Compressibility from Table 5.1 data.

8. NCG Library

A library of data for gases of interest is available for use. Any of the numbers may be changed via user input. The available gases and the associated constants are defined below. Ten user-defined gases called GASk, where k is any letter from A to J, can also be used, but the user must define all the values for the associated constants. Units for the parameters are given in the NCG Users' Guide.

Hydrogen (H_2)		
	MELCOR Name: H2	
	Molecular Weight: 0.0020162	
	c_{v0}:	-17849.
	c_{v1}:	11.28298
	c_{v2}:	-2.1081958E-3
	c_{v3}:	1.5635602E-7
	c_{vsqrt}:	865616.
	c_{vm1}:	-8188058.3
	c_{vm2}:	1.925734E8

	T_{low}:	100.
	T_{up}:	6000.
	e_f:	0.
	s_0:	0.

Deuterium (D_2)		
	MELCOR Name: D2	
	Molecular Weight: 0.00400	
	c_{v0}:	5508.8
	c_{v1}:	-2.0277
	c_{v2}:	3.3827E-3
	c_{v3}:	-1.0842E-6
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	600.
	T_{up}:	1500.
	e_f:	0.
	s_0:	0.

Helium (He)		
	MELCOR Name: HE	
	Molecular Weight: 0.004003	
	c_{v0}:	3152.955
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	1.
	T_{up}:	10000.
	e_f:	0.
	s_0:	0.

Nitrogen (N_2)		
	MELCOR Name: N2	
	Molecular Weight: 0.02801	
	c_{v0}:	1.117E3
	c_{v1}:	0.

	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	-2.880E5
	c_{vm2}:	5.348E7
	T_{low}:	300.
	T_{up}:	5000.
	e_f:	0.
	s_0:	0.

Oxygen (O_2)		
	MELCOR Name: O2	
	Molecular Weight: 0.032	
	c_{v0}:	1245.
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	-16763.
	c_{vm1}:	1.111E5
	c_{vm2}:	0.
	T_{low}:	300.
	T_{up}:	2778.
	e_f:	1.7828E7
	s_0:	0.

Argon (Ar)		
	MELCOR Name: AR	
	Molecular Weight: 0.03994	
	c_{v0}:	316.0827
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	1.
	T_{up}:	10000.
	e_f:	0.
	s_0:	0.

Methane (CH_4)		
	MELCOR Name: CH4	
	Molecular Weight: 0.0160324	
	c_{v0}:	660.6
	c_{v1}:	3.462
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	300.
	T_{up}:	833.
	e_f:	-4.5153E6
	s_0:	0.

Carbon Monoxide (CO)		
	MELCOR Name: CO	
	Molecular Weight: 0.028	
	c_{v0}:	1.116E3
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	-2.7312E5
	c_{vm2}:	4.9348E7
	T_{low}:	300.
	T_{up}:	5000.
	e_f:	6.3286E6
	s_0:	0.

Carbon Dioxide (CO_2)		
	MELCOR Name: CO2	
	Molecular Weight: 0.044	
	c_{v0}:	1351.35
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	-3.4497E5
	c_{vm2}:	4.138E7
	T_{low}:	300.
	T_{up}:	3500.
	e_f:	4.0785E6
	s_0:	0.

Acetylene (C_2H_2)		
	MELCOR Name: C2H2	
	Molecular Weight: 0.026016	
	c_{v0}:	1.1457E3
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	1.
	T_{up}:	10000.
	e_f:	8.8104E6
	s_0:	0.

Ethylene ($C_2 H_4$)		
	MELCOR Name: C2H4	
	Molecular Weight: 0.028032	
	c_{v0}:	334.51
	c_{v1}:	1.7568
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	194.
	T_{up}:	611.1
	e_f:	1.9536E6
	s_0:	0.

Ammonia (NH_3)		
	MELCOR Name: NH3	
	Molecular Weight: 0.017029	
	c_{v0}:	1.7012E3
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	1.
	T_{up}:	10000.
	e_f:	-2.557E6
	s_0:	0.

Nitrogen Monoxide (NO)		
	MELCOR Name: NO	
	Molecular Weight: 0.03005	
	c_{v0}:	6.8985E2
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	1.
	T_{up}:	10000.
	e_f:	6.561E6
	s_0:	0.

Nitrous Oxide (N_2O**)**		
	MELCOR Name: N2O	
	Molecular Weight: 0.04401	
	c_{v0}:	736.32
	c_{v1}:	0.
	c_{v2}:	0.
	c_{v3}:	0.
	c_{vsqrt}:	0.
	c_{vm1}:	0.
	c_{vm2}:	0.
	T_{low}:	1.
	T_{up}:	10000.
	e_f:	4.6699E6
	s_0:	0.

User Defined Gases (-)		
	MELCOR Name: GASk, k = A, B, … , J	
	Molecular Weight: -1.	
	c_{v0}:	-1.
	c_{v1}:	-1.
	c_{v2}:	-1.
	c_{v3}:	-1.
	c_{vsqrt}:	-1.
	c_{vm1}:	-1.
	c_{vm2}:	-1.
	T_{low}:	-1.
	T_{up}:	-1.
	e_f:	-1.
	s_0:	-1.

References

1. J. H. Keenan, et al., Steam Tables: Thermodynamic Properties of Water, Including Vapor, Liquid, and Solid Phases (SI Units), John Wiley & Sons, Inc., New York (1978).

2. JANAF Thermochemical Tables, Dow Chemical Company, Thermal Research Laboratory, Midland, MI (1965).

3. N. B. Vargaftik, Handbook of Physical Properties of Liquids and Gases: Pure Substances and Mixtures, 2nd ed., Springer-Verlag (1975).

4. E. R. G. Eckert and R. M. Drake, Jr., Heat and Mass Transfer, McGraw-Hill, New York (1959).

5. R. Reid, et al., The Properties of Gases and Liquids, McGraw-Hill, New York (1970).

6. M. M. Wakil, Nuclear Heat Transport, International Textbook Co., New York (1971).

7. 1967 Steam Tables, Electrical Research Association, St. Martin's Press, New York (1967).

8. N. B. Vargaftik, Handbook of Physical Properties of Liquids and Gases: Pure Substances and Mixtures, 2nd ed., Springer-Verlag (1975).

9. J. Hilsenrath, et al., Tables of Thermodynamic and Transport Properties, Pergamon Press, New York (1960).

10. D. R. Pitts and L. E. Sissom, Schaum's Outline of Theory and Problems of Heat Transfer, McGraw-Hill, New York (1977).

11. R. C. Nause and M. T. Leonard, Thermophysical Property Assessment Report for Savannah River Site Production Reactor Materials, Science Applications International Corporation, SAIC Report No. 89/6507, Rev. 1 (May 1990).

12. R. Bird, et al., Transport Phenomena, John Wiley and Sons, New York (1960).

13. R. Reid, et al., The Properties of Gases and Liquids, McGraw-Hill, New York (1970).

14. J. K. Fink, R. Simms and B. A. Brock, Material Properties for HWR-NPR Severe Accident Studies, Argonne National Laboratory, ANL/NPR-90/005 (March 1990).

15. Kerntechnischer Ausschuss, *SAFETY STANDARDS of the Nuclear Safety Standards Commission (KTA)*, “KTA 3102.1, Reactor Core Design for High-temperature Gas-Cooled Reactor, Part 1: Calculations of the Material Properties of Helium”, Carl Heymanns Verlag KG (1978).

16. E.W. Lemmon, M.L. Huber, M.O. McLinden, *NIST Reference Fluid Thermodynamic and Transport Properties--REFPROP Version 8.0*, NIST, Boulder, CO (2007).

17. L. Baker and C. Just, Studies of Metal-Water Reactions at High temperatures; III. Experimental and Theoretical Studies of the Zirconium-Water Reaction, ANL-6548, Argonne National Laboratory, Chicago (May 1962).

Passive Autocatalytic Recombiner (PAR) Package

The MELCOR ESF package models the physics for the various engineered safety features (ESFs) in a nuclear power plant. The Passive Autocatalytic Recombiner (PAR) package constitutes a subpackage within the ESF package and calculates the hydrogen removal rate from the operation of hydrogen recombiners. This reference manual gives a description of the physical models and numerical solutions implemented in the PAR package.

User input for running MELGEN and MELCOR with the PAR package activated is described separately in the Passive Autocatalytic Recombiner section of the Users' Guide.

Contents

1. Introduction

The MELCOR ESF package models the thermal-hydraulic behavior of various engineered safety features (ESFs) in nuclear power plants. One ESF is designed to react hydrogen in a reactor containment in a continuous manner with the goal of preventing hydrogen concentrations from increasing to levels that could produce large scale hydrogen deflagration or even detonations. There are several methods for achieving hydrogen removal. The most common method is the use of igniters, which provide local ignition sources that can precipitate hydrogen burns near the lower burn limits. However, these systems depend on the availability of a power source, which in certain accident sequences may be lost. In addition, these systems do not operate under certain steam inerted conditions that can lead the igniter system to precipitate a large burn when inerted conditions are removed. The passive autocatalytic systems, however, do not require a power source and are not strongly affected by inerted conditions. The benefits derived from this type of hydrogen control system are obvious and are under study for possible backfitting to existing power plants.

The Passive Autocatalytic Recombiner (PAR) package constitutes a subpackage within the ESF package and calculates the rate of hydrogen removal generated by PAR type hydrogen removal systems. The default MELCOR model is based on the Fischer model [1], which is a parametric model developed for a specific PAR unit. The user input provides correlation coefficients for the general mathematical form of the model. These coefficients are used by the code to calculate the total gas flow rate through a PAR unit. From the PAR gas flow rate together with user provided PAR efficiencies, transient relaxation times, delay times, and the internally calculated hydrogen mole fractions, a per-PAR-unit hydrogen reaction rate is calculated. This rate is then multiplied by the current timestep and the user provided number of active PAR units to determine the change in hydrogen, oxygen, and steam masses. These differential masses are then passed to CVH as sources/sinks.

It is noted that a PAR design has been developed, studied, tested, and reported on in the technical literature. The NIS Company in Hanau, Germany developed this type of PAR. The design consists of parallel plate cartridges containing palladium-coated aluminum micro pellets. It has been tested in a series of experiments as described in References [1], [2] and [3]. The Reference [3] tests are of particular interest because these tests of the NIS PAR were performed at the Sandia National Laboratories/NM *for the NRC*.

The type of PAR used as the default model in the MELCOR model is the NIS type of PAR. This type of PAR was chosen as the default model because of the literature available and because this was the type studied by the NRC; see Reference [3]. It should be noted that other PAR designs are available. However, it is likely that many other specific designs can be modeled within the parametric framework described here. In the event that it is desired to study a design concept sufficiently different from the type described here, it was necessary to provide a more general input option. This provides the user with the

option to specify a control function with which the PAR flow rate can be calculated as a function of one or more system variables. In addition, an option is provided that allows the user to specify the PAR efficiency using a control function. A more detailed description of the model is provided in the next section.

2. Model Description

The chemical recombination of hydrogen and oxygen to produce steam and release energy is described by the equation

$$2\mathrm{H}_2 + \mathrm{O}_2 \rightarrow 2\mathrm{H}_2O + 1.2 \times 10^8 (\mathrm{J/kg}) \qquad (2\text{-}1)$$

The hydrogen reaction rate for a single PAR unit may be expressed in terms of the total volumetric flow rate passing through the unit as follows:

$$R_H = \eta \ \ \rho_H \ Q \ \ f(t) \qquad (2\text{-}2)$$

where

R_H = hydrogen reaction rate (kg/sec)

ρ_H = hydrogen density of entering gas (kg/m^3)

η = hydrogen reaction efficiency (~0.85)

Q = total gas-phase volumetric flow rate through the unit (m^3/sec)

$f(t)$ = $\left[1 - e^{-\left[\frac{t-t_0}{\tau}\right]}\right]$ = relaxation time function during the initial PAR heat-up

τ = characteristic heat-up time (~1800 sec)

t_0 = time of PAR initiation (s)

t = time after PAR initiation (s)

The relaxation time function is intended to account for the observed transient interval before the PAR attains steady-state operation. It is thought that the primary transient effect is due to the time required for the catalytic elements to come up to operating temperature. For the configuration tested in Reference [1], the relaxation time,τ, was determined to be on the order of half an hour.

The user through control functions may in the general case, provide the total volumetric flow rate through a PAR unit. However, an expression for NIS type PAR units [1,4] has been found to accurately describe the flow rate, and is given by:

$$Q = a\ C_H^b \tag{2-3}$$

where

C_H = hydrogen concentration (mole fraction)

a = constant that depends on PAR unit design parameters (~0.67 kg/sec)

b = exponent that depends on PAR unit design parameters (~0.307)

Some of the parameters in Equations (2-2) and (2-3) are provided by the CVH package in the MELCOR code. For the NIS model, other than the current time, the hydrogen density and mole fraction are the only two required parameters. Depending on the specifics of the model, however, user-defined PAR flow rates may depend on other CVH parameters such as temperature or pressure.

For typical containment volumes, several PAR units are required to control the H_2 concentration. The total hydrogen depletion rate is then found simply by summing the rates from the individual units in a control volume. The user may specify more than one type of PAR and specify the number of units of each PAR type in a single control volume or distributed in several control volumes.

The transient effects described by the term $f(t)$ in Equation (2-2) derives from the solution for a single step function in hydrogen concentration with the initial concentration being zero, and the hydrogen concentration in a control volume remaining constant.

In general, however, the hydrogen concentration does not remain constant and may in fact involve multiple ‘bursts’ of hydrogen injection into a containment volume combined with continued releases from the vessel or from ex-vessel fuel/metal, fuel/concrete interactions. It is assumed here that if there is an increase in H_2 concentration, which implies an eventual increase in volumetric flow through the PAR (Equation (2-3)), then the time dependence of the change in flow follows the same relaxation behavior implied by Equation (2-4). Based on this assumption, a more general approach to the transient effects is employed in the MELCOR model. The transient effects are described by the differential equation:

$$\frac{dQ}{dt} = \frac{1}{\tau}[Q_{ss} - Q] \tag{2-4}$$

where Q is the volumetric flow rate, and Q_{ss} is the steady-state flow rate implied by the hydrogen concentration found from Equation (2-3). Integrating Equation (2-4) over timestep, Δt , in which the flow rate changes from Q_{old} to Q_{new}, gives the following result:

$$Q_{new} = Q_{ss}\left[1 - e^{-\Delta t/\tau}\right] + Q_{old}e^{-\Delta t/\tau} \tag{2-5}$$

Equation (2-5) provides for transient effects, but requires that the hydrogen reaction rate be carried as a dynamic variable with the old and new values stored in the main variable array. Note that the transient term $f(t)$ is now implicit in the flow rate equation. It should also be noted that Equation (2-5) is applied on a timestep-by-timestep basis so that the flow rates and H_2 burn rates respond in a continuous manner to transient conditions such as increases or decreases in the hydrogen concentration. It is thus, not necessary to track the thermal response of the PAR catalytic elements. The temperature of the PAR catalytic elements is implicit in the correlations used in the Fischer model, but is not explicitly available as an output parameter.

As described by Equation (2-1), the recombination of hydrogen results in an oxygen depletion rate and a steam mass increase rate. Since the reaction is exothermic, there is an associated change in gas temperature. These rates are given as follows for the control volume in which the PAR unit is located:

$$\frac{dm(H_2)}{dt} = -R_H, (\mathrm{kg/s}) \tag{2-6}$$

$$\frac{dm(O_2)}{dt} = -\frac{M_{O_2}}{2M_{H_2}} * R_H, (\mathrm{kg/s}) \tag{2-7}$$

$$\frac{dm(H_2O)}{dt} = -\frac{M_{O_2}}{2M_{H_2}} * R_H, (\mathrm{kg/s}) \tag{2-8}$$

$$\frac{dH}{dt} = \sum_{i=1}^{N} w_{i,in}h_{i,in} - \sum_{i=1}^{N} w_{i,out}h_{i,out}, (\mathrm{W}) \tag{2-9}$$

Note that in Equations (2-6) through (2-8), M refers to the molecular weight of the species. The indices on the sums in Equation (2-9), for the change in total enthalpy H, refer to the specific gas constituents (H_2, O_2, H_2O, CO, CO_2, etc.), while w and h refer to the mass flow rate and specific enthalpy of each constituent. Also, because MELCOR uses a consistent reference point (JANIF Convention) for all gas-phase thermodynamic properties, the heating rate given by Equation (2-9) is not needed as a source in the CVH package. JANAF refers to a set of thermochemical tables [5]. The JANAF convention implicitly includes all heats of formation in the enthalpy functions for each material. In so doing, the heat of reaction, for example in the burning of hydrogen and oxygen, is included in the enthalpy of the reaction product (steam in this example). The advantage is that all chemical reactions, such as those generated in this PAR Package, can be treated simply as changes in the masses of the reactants and products, and the heat effects are accounted for automatically through the equations of state.

The mass rates computed by the PAR Package in Equations (2-6) through (2-8) are multiplied by the current timestep and the differential masses passed to the CVH package.

The PAR testing discussed in Reference [3] did not identify any problem regarding "remaining capacity" in terms of possible degradation of the catalytic elements.

In addition, it is important to note that the literature [2] does address the investigations into possible decrease of the PAR performance because of catalyst inhibitors and poisons. The investigations performed indicate that the effects of catalyst inhibitors and poisons are negligible. However, if further studies provide evidence that catalytic elements can be degraded, an option is provided that allows the user to specify the PAR efficiency, η, by a control function that can be a function of time, aerosol concentration, etc.

The change of gas temperature as it passes through the PAR can be estimated by noting that, since no mass or energy has been added from external sources to the gas stream (with the exception noted later), the enthalpy flow rates in and out of the PAR unit must be equal. Thus, from Equation (2-10), the following relationship between the inlet and outlet conditions must exist,

$$\sum_{i=1}^{N} w_{i,in} h_{i,in} = \sum_{i=1}^{N} w_{i,out} h_{i,out} \qquad (2\text{-}10)$$

$w_{i,out}$ = the mass flow rate of the i^{th} gas-phase species exiting the PAR (kg/s),

$w_{i,in}$ = the mass flow rate of the i^{th} gas-phase species entering the PAR (kg/s),

$h_{i,out}$ = the specific enthalpy of the i^{th} gas-phase species exiting the PAR (J/kg), and

$h_{i,in}$ = the specific enthalpy of the i^{th} gas-phase species entering the PAR (J/kg).

The outlet temperature is then evaluated by a Newton's method iteration in which the inlet enthalpy flow (the left side of Equation (2-10)) is evaluated and successive estimates of the outlet temperature are calculated until the difference between the inlet enthalpy rate and the outlet enthalpy rate are less than a specified limit.

It should be noted that the estimate of outlet temperature is not accurate during rapid transient situations. During periods in which the PAR elements are heating up or cooling down some fraction of the energy is transferred to or from the PAR elements. In the former case the outlet temperature is over-predicted and in the latter under-predicted.

Because the outlet temperature is only an output variable and does not affect any other calculations in the model or in the code, this is not a serious deficiency.

3. Discussion and Development Plans

Although the proposed model is rather simple and was developed in the form of a correlation for a specific design configuration, it is thought that it provides the capability to accurately model the operation of the NIS type PAR unit. It should also provide the capability to treat a wide variety of similar catalytic reactions given the required performance characteristics. In addition, the options that provide for a user-specified flow rate and efficiency using Control Functions provide the required additional flexibility and utility to model essentially any type of PAR unit.

For lack of sufficient data regarding other catalytically induced reactions, the current model does not provide for the reaction of CO or other combustible species. Future improvements to the PAR models may consider these reactions.

References

1. K. Fischer, “Qualification of Passive Catalytic Module for Hydrogen Mitigation,” Nuclear Technology, Vol. 112, p. 58 (October 1995).

2. EPRI report, “Qualification of Passive Autocatalytic Recombiners for Combustion Gas Control in ALWR Containments,” EPRI Electric Power Research Institute, Palo Alto CA (April 8, 1993).

3. T. K. Blanchat and A. C. Malliakos (NRC/RES), “Passive Autocatalytic Recombiner (PAR) Tests at the SNL,” CSARP Meeting (May 1998).

4. R. Sher, J. Li, and D. E. Leaver, “Models for Evaluating the Performance of Passive Autocatalytic Recombiners (PARs),” 1995 National Heat Transfer Conference, Portland OR, ANS Proceedings, HTC-Vol. 8 (August 5 – 9, 1995).

5. JANAF Thermochemical Tables, DOW Chemical Company, Thermal Research Laboratory, Midland, MI (1965.)

RadioNuclide (RN) Package

The RadioNuclide (RN) package models the behavior of fission product aerosols and vapors and other trace species, including release from fuel and debris, aerosol dynamics with vapor condensation and revaporization, deposition on structure surfaces, transport through flow paths, and removal by engineered safety features. The package also allows for simplified chemistry controlled by the user.

Boundary conditions for the various models are obtained from other MELCOR packages: fluid conditions are obtained from the Control Volume Hydrodynamics (CVH) package, fuel and debris temperatures are obtained from the Core (COR) and Cavity (CAV) packages, and structure surface temperatures are obtained from the Heat Structures (HS) package. The COR and CAV packages also provide information regarding bulk debris relocation, allowing the RN package to perform relocation of unreleased fission products in parallel. Likewise, advection of radionuclides between control volumes is done using CVH flows, and wash-off of radionuclides deposited on heat structures is determined from drainage of water films calculated by the HS package. The RN package determines decay heat power for current radionuclide inventories from the Decay Heat (DCH) package when requested by each of these packages.

This document describes in detail the various models incorporated in the RN package in MELCOR. Details on input to the RN package can be found in the RN Users' Guide.

Contents

List of Figures

List of Tables

1. Introduction

Since MELCOR is intended as a tool for probabilistic risk assessment (PRA), it must account for the release and transport of radioactive fission products that upon release to the environment become the *source term*, which is one major product of the overall accident calculation in MELCOR. Source terms are then used to calculate consequences, an important input to the PRA. Such processes as thermal-hydraulics and core degradation are calculated in MELCOR to support calculation of the source term.

The RadioNuclide (RN) package in MELCOR calculates the release and transport behavior of fission product vapors and aerosols. Most of the models and concepts included in the RN package are discussed in detail in the fission product phenomena assessment report prepared at the beginning of MELCOR development [1]. Only a brief overview of the concepts and models is included in this section; Section 2 contains detailed descriptions of the models used in the RN package.

As a source term code, MELCOR is especially concerned with those fission products (and daughters) released during an accident, which are particularly important for determining consequences and risks. However, to model the transport of these important fission products properly, it is necessary to model the transport of other mass that affects the transport of radionuclide mass. For example, radiocesium exists as CsOH, so the mass of the hydroxide must be modeled, and if the CsOH aerosol interacts with concrete or water aerosols, the transport and thus the mass of the latter must also be modeled. Accordingly, MELCOR treats the molecular forms of all important fission products and also models the transport of all nonradioactive masses (water and concrete or other structural aerosols) with which fission products may interact. Therefore, in this manual the term *radionuclide* is generally taken to mean all masses, both radioactive and nonradioactive, that affect fission product transport.

Rather than tracking all fission product isotopes, the masses of all the isotopes of an element are modeled as a sum; that is, the total element mass, not its individual isotopes, is modeled. Furthermore, elements are combined into material *classes*, groups of elements with similar chemical behavior. Fifteen material classes are typically used, thirteen containing fission products, plus water, and concrete oxides. Combination of classes to form new classes upon release, such as Cs + I to CsI, is permitted. The decay heat power per unit initial mass for each class is determined by the Decay Heat (DCH) package based on the class compositions.

Initial radionuclide inventories for each class are generally based on whole-core inventories calculated using the ORIGEN code [2,3], and distributions may be specified for the fuel in the core, the fuel-cladding gap, any initial cavity debris, and the atmosphere and pool of any control volume. Until released as vapors or aerosols, fission products within the fuel are transported with the fuel as it relocates from core cell to core cell or is

ejected to the reactor cavity. The decay heat power from radionuclides contained in a control volume, both those that are gas borne and those deposited on heat structure surfaces or contained in water pools, can be apportioned among the atmosphere, surfaces, and pools according to specifications supplied by the user, thus allowing the different penetrating powers of α, β, and γ radiation to be modeled appropriately. Radiation is allocated to various surfaces in the control volume on the basis of area.

Release of radionuclides can occur from the fuel-cladding gap by exceeding a failure temperature criterion or losing intact geometry, from material in the core using the various CORSOR empirical release correlations [4,5] based on fuel temperatures, and during core-concrete interactions in the reactor cavity using the VANESA [6] release model. After release to a control volume, masses may exist as aerosols and/or vapors, depending on the vapor pressure of the radionuclide class and the volume temperature.

Aerosol dynamic processes and the condensation and evaporation of fission product vapors after release from fuel are considered within each MELCOR control volume. The aerosol dynamics models are based on MAEROS [7], a multisection, multicomponent aerosol dynamics code, but without calculation of condensation. Aerosols can deposit directly on surfaces such as heat structures and water pools, or can agglomerate and eventually fall out once they exceed the largest size specified by the user for the aerosol size distribution. Aerosols deposited on surfaces can be vaporized (if they are volatile) but can only be resuspended if the resuspension model is enabled.

The condensation and evaporation of radionuclide vapors at the aerosol surfaces, pool surfaces, and heat structure surfaces are decoupled from MAEROS. These processes are evaluated by the rate equations from the TRAP-MELT2 code [8], which are based on the surface area, mass transfer coefficients, and the difference between the present surface concentration and the saturation surface concentration.

The steam condensation/evaporation is also decoupled from the MAEROS solution for agglomeration and deposition in order to reduce the stiffness of the differential equation set. The amount of steam condensed or aerosol water evaporated is calculated by thermodynamics routines called by the Control Volume Hydrodynamics (CVH) package.

Water droplets are transported as *fog* by the CVH package and treated as water-class aerosol by the RN package. (Water in pools or condensed on surfaces is not treated by the RN package.) Other radionuclide aerosols and vapors are transported between control volumes by bulk fluid flow of the atmosphere and the pool, assuming zero slip between the radionuclides and the host medium (steam, water, etc.). In addition, in the absence of bulk flow, aerosols may move by Brownian motion or by gravitational settling through openings between control volumes.

The difference between CVH fog and RN water-class masses in a control volume at the end of the CVH advancement represents net condensation of water onto or evaporation

from the aerosols in that volume. The net change in water mass is imposed on the water-class inventory in the RN package, which then uses the Mason equation [9] to distribute the mass change over the aerosol size distribution in the control volume.

Models are available for the removal of radionuclides by pool scrubbing, filter trapping, and containment spray scrubbing. The pool scrubbing model is based on the SPARC code [10] and treats both spherical and elliptical bubbles. The model includes condensation at the pool entrance, Brownian diffusion, gravitational settling, inertial impaction, and evaporative forces for the rising bubble. Currently, only aerosols are removed by pool scrubbing in the RN package. Water condensation and evaporation are calculated within the CVH package using its own implementation of SPARC modeling (see the CVH Reference Manual). The filter model can remove aerosols and fission product vapors with a specified maximum mass loading. The containment spray model is based on the model in HECTR 1.5 [11] and removes both vapors and aerosols from the atmosphere.

Chemistry effects can be simulated in MELCOR through the *class reaction* and *class transfer* models, which are controlled entirely by user-specified parameters. The class reaction process uses a first-order reaction equation to simulate reversible chemical reactions. The class transfer process, which can instantly change the material class or location of a radionuclide mass, can be used to simulate fast, irreversible chemical reactions. With these two processes, phenomena including adsorption, chemisorption and other important chemical reactions can be simulated. Only fission product vapors are currently treated with these mechanisms. In addition, chemisorption of radionuclides on surfaces can be simulated with the chemisorption model.

Most intravolume processes involving radionuclides are calculated first in the RN package, including fission product release, aerosol agglomeration and deposition, fission product condensation and evaporation, distribution of decay heat, and chemical interactions. The effects of these processes are included in the hydrodynamic transport and thermodynamic calculations performed in the CVH package, executed subsequently.

The transport of fission products is inferred from the transport of hydrodynamic materials, but the CVH package may subcycle during a MELCOR timestep. Since radionuclide advection must also abide by the Courant limit, the transport calculations are performed by RN package utility routines called from within the CVH subcycle loop. Part of this transport process includes removal of fission product aerosols and vapors, for example, by filters.

After CVH has advanced through the full MELCOR system timestep, the additional intervolume process of pool scrubbing is calculated. While water condensation/ evaporation is an intravolume process, it also is calculated after the CVH package thermodynamics calculations have been performed so that the mass of water condensed in a control volume during the timestep is known.

2. Detailed Models

2.1 General Framework

The RN package operates on the principle of material classes, which are groups of elements that have similar chemical properties. The number of classes is specified on the RN1_DIM input record, with a default of 17 classes. Classes are generally referred to by their class name or representative element. Combination of masses in these classes upon release to form compounds in other classes, such as Cs + I to CsI, is permitted subject to stoichiometric constraints (e.g., excess Cs is retained in the Cs class). For the RN package, the classes must be in numerical order without any gaps. A maximum of 30 classes can presently be employed.

Each class is described by the following set of properties for use in various models:

1	release rates in core	(see Section 2.3)
2	molecular weights	(see Section 2.3)
3	vapor pressure	(see Section 2.5)
4	vapor diffusivity	(see Section 2.5)
5	decay heat power	(see DCH Package Users' Guide)

Two molecular weight values are used for each class; the *elemental* molecular weight (i.e., the element's atomic weight) and the *compound* molecular weight, which are specified in sensitivity coefficient array 7120 (see Appendix A). The elemental molecular weight is used to determine the number of moles of *radioactive* material that are released and available for combination with other RN classes. The compound molecular weight is used to increase the released mass due to combination upon release with *nonradioactive* materials if that is expected to occur (e.g., Cs with H_2O to form CsOH). *Total* class masses after release therefore include both radioactive and nonradioactive masses. In addition, nonradioactive masses from bulk materials in the Core or Cavity package (e.g., cladding Zircaloy, structural steel, control poison, or concrete) may be released as vapors or aerosols and added to the total class masses but not to the radioactive masses of the class to which the materials are assigned.

Some models in the RN package use groupings of elements different from the groupings defined in Table 2.1. Transfers of masses between various models must therefore use *mapping* strategies.

For the transfer of bulk, nonradioactive, Core package structural masses released by the CORSOR models to the RN classes (see Section 2.3); the default mapping defined in Table 2.2 is employed. This mapping may be changed with input records RN1_CRCL, but this practice is discouraged. Note from Table 2.2 that B_4C control poison in BWRs is

mapped totally into the boron class, whereas Ag-In-Cd control poison in PWRs is split between the Cd and Ag classes using the percentages shown.

The VANESA model for radionuclide releases from debris in the cavity (see Section 2.3) recognizes 25 different species groups (for most, several different compounds of one element), and mapping must be used both to transfer RN class masses in the debris (as initially specified and as transferred from the COR and/or Fuel Dispersal Interactions [FDI] packages) to the VANESA groups and also to transfer them back again into the RN classes as VANESA calculates releases. The default mappings for to-VANESA and from-VANESA transfers are defined in Table 2.3 and Table 2.4, respectively. These mappings may be changed with input records RN1_CLVN and RN1_VNCL.

Table 2.1 RN Class Compositions

Class Name	Representative	Member Elements
1. Noble Gases	Xe	He, Ne, Ar, Kr, Xe, Rn, H, N
2. Alkali Metals	Cs	Li, Na, K, Rb, Cs, Fr, Cu
3. Alkaline Earths	Ba	Be, Mg, Ca, Sr, Ba, Ra, Es, Fm
4. Halogens	I	F, Cl, Br, I, At
5. Chalcogens	Te	O, S, Se, Te, Po
6. Platinoids	Ru	Ru, Rh, Pd, Re, Os, Ir, Pt, Au, Ni
7. Early Transition Elements	Mo	V, Cr, Fe, Co, Mn, Nb, Mo, Tc, Ta, W
8. Tetravalent	Ce	Ti, Zr, Hf, Ce, Th, Pa, Np, Pu, C
9. Trivalents	La	Al, Sc, Y, La, Ac, Pr, Nd, Pm, Sm, Eu, Gd, Tb, Dy, Ho, Er, Tm, Yb, Lu, Am, Cm, Bk, Cf
10. Uranium	U	U
11. More Volatile Main Group	Cd	Cd, Hg, Zn, As, Sb, Pb, Tl, Bi
12. Less Volatile Main Group	Ag	Ga, Ge, In, Sn, Ag
13. Boron	B	B, Si, P
14. Water	H_2O	H_2O
15. Concrete	- -	- -
16. Cesium Iodide	CsI	Classes 2 and 4
17. Cesium Molybdate	CsM	Classed 2 and 7

In addition to the 25 VANESA groups, two additional groups can be transferred to VANESA but are changed before VANESA uses them. They are I (VANESA group 26), which is combined automatically with Cs, and Xe (VANESA group 27), which VANESA releases immediately. VANESA assumes that Cs is in excess so that no elemental I

remains as debris is added to the cavity. Also, aerosol products from concrete ablation (VANESA groups 12 through 16) are automatically transferred to the RN concrete class, and bulk gases (VANESA group 1) are transferred directly to the CVH package. The user should not specify mapping values for any of these VANESA groups.

Table 2.2 COR Material to RN Class Mapping

COR Material		**RN Class (Rep. Element)**		
1	UO2	10	U	
2	Zr	8	Ce	
3	ZrO2	8	Ce	
4	Steel	7	Mo	
5	Steel Oxide	7	Mo	
6	Control Rod Poison	13	B	100% BWR / 0% PWR
		11	Cd	0% BWR / 5% PWR
		12	Ag	0% BWR / 95% PWR

Table 2.3 RN Class to VANESA Species Mapping

RN Class		**VANESA Species**	
1	Xe	27	Xe (released instantaneously)
2	Cs	19	Cs
3	Ba	20	Ba
4	I	26	I (immediately forms CsI)
5	Te	9	Te
6	Ru	6	Ru
7	Mo	5	Mo
8	Ce	23	Ce
9	La	22	La
10	U	17	U
11	Cd	8	Sb
12	Ag	7	Sn
13	B	0	(RN class not present in fuel)
14	H2O	0	(RN class not present in fuel)
15	Concrete	0	(RN class not present in fuel)
16	CsI	25	CsI

RN Class		VANESA Species	
17	CsM	19 & 25	Cs and Mo

Warning: If a class is redefined from the default values, or if a new class is added, all of the properties, including mappings, should be evaluated and possibly redefined through the RN sensitivity coefficients. Default values for these properties are defined based on the elements in each class. Whether default values are appropriate when classes are modified must be determined by the user. Note that the DCH package might also have to be redefined in a consistent manner.

Table 2.4 VANESA Species to RN Class Mapping

VANESA Species		RN Class	
1	bulk gases (from CORCON)	(released by CAV pkg to CVH)	
2	Fe	7	Mo
3	Cr	7	Mo
4	Ni	6	Ru
5	Mo	7	Mo
6	Ru	6	Ru
7	Sn	12	Ag
8	Sb	11	Cd
9	Te	5	Te
10	Ag	12	Ag
11	Mn	7	Mo
12	Ca (from concrete ablation)	15	Concrete
13	Al (from concrete ablation)	15	Concrete
14	Na (from concrete ablation)	15	Concrete
15	K (from concrete ablation)	15	Concrete
16	Si (from concrete ablation)	15	Concrete
17	U	10	U
18	Zr	8	Ce
19	Cs	2	Cs
20	Ba	3	Ba
21	Sr	3	Ba
22	La	9	La
23	Ce	8	Ce
24	Nb	7	Mo
25	CsI	2	Cs and 4 I

VANESA Species		RN Class
26	I	(combined with Cs by VANESA)
27	Xe	(released by VANESA)

2.2 Initial Radionuclide Inventories

Initial inventories and distributions of radionuclides must be specified for the core, for the cavity, and for control volume pools and atmospheres. (Inventories for some locations may be zero initially.) Masses can be distributed among core cells according to radial and axial decay heat power profiles in the core. In addition, a fraction of the radionuclides in a core cell can be designated as residing in the fuel-cladding gap.

Total radioactive class masses are normally determined by the DCH package from the operating power of the reactor and the mass of each element in the class per unit of operating power (see the DCH Package Reference Manual and Users' Guide). RN package input generally defines only the initial distribution of these masses in the core and cavity through reference values and multipliers specified on the RN1_FPN input records. However, options are provided to use these records to specify the class masses directly. These options are useful for analysis of experiments.

The total mass inventories for all RN classes in a particular core cell or in a cavity are normally calculated from user-specified multipliers r_1 and r_2 as:

$$M_x = r_1 \, r_2 M_{x,ref} \tag{2-1}$$

where $M_{x,ref}$ is a reference value for class x that may be taken as the total class mass defined by the DCH package or as the inventory in some other core cell or cavity location, depending on the option chosen. For core cells, r_1 and r_2 typically represent axial and radial multipliers to specify the decay heat power profile in the core, while for cavities they are arbitrary. If the DCH package option is chosen, however, the mass of the uranium class (default class 10) is calculated by decrementing the total uranium mass in the Core package, $M_{U,COR}$ by the sum of the masses in the remaining classes, i.e.:

$$M_{U,RN} = M_{U,COR} - \sum_{i \neq U} M_i \text{ (Uranium class only)} \tag{2-2}$$

Optionally, as specified on the RN1_FPN records, the mass for a specified class in a particular core cell or cavity location may be input directly as:

$$M_x = r_1 \, r_2 \tag{2-3}$$

where r_1 is typically chosen as the total mass, with r_2 defined as the fraction of that mass in the core cell or cavity location. The various options are additive and may be combined as convenient. Note that masses can also be reduced if a negative multiplier is used.

The masses given by Equations (2-1) through (2-3) determine the total radioactive mass of radionuclides in a particular core cell, including the fuel-cladding gap. The fraction of radioactive mass that resides in the gap is determined by the parameter r_1 input on the RN1_GAP input record series (different from r_1 input on RN1_FPN). Depending on the input option chosen, the gap fraction F_x may be specified directly for each class as:

$$F_x = r_1 \tag{2-4}$$

or it may be calculated as a proportion of the gap fraction $F_{x,ref}$ at some other location.

$$F_x = r_1 F_{x,ref} \tag{2-5}$$

For a core cell, the radioactive masses residing in the fuel and gap, $M_{x,fuel}$ and $M_{x,gap,R}$, respectively, are thus given by:

$$M_{x,fuel} = (1 - F_x)\, M_x \tag{2-6}$$

$$M_{x,gap,R} = F_x M_x \tag{2-7}$$

The total masses residing in the gap must be calculated to account for the addition of nonradioactive material from presumed chemical reactions following release from the fuel. (See the discussion of total vs. radioactive masses in Section 2.1.) If the gap fraction has been specified directly from Equation (2-4), the total gap mass $M_{x,gap,T}$ is given by

$$M_{x,gap,T} = r_2 M_{x,gap,R} \tag{2-8}$$

where r_2 is the ratio of total mass to radioactive mass (usually the ratio of compound to elemental molecular weights, matching the values in sensitivity coefficient array 7120; see Appendix A), whereas if the gap fraction has been specified as a proportion r_1 of the gap fraction at some other location with Equation (2-6), the total gap mass is that same fraction of the total gap mass $M_{x,gap,T,ref}$ at the other location,

$$M_{x,gap,T} = r_1\, M_{x,gap,T,ref} \tag{2-9}$$

in which case no value is needed for r_2 since it is already reflected in $M_{x,gap,T,ref}$ (any value input for r_2 is ignored).

The distribution of radionuclide masses between fuel and gap in a core cell changes with time due to release and the relocation of fuel. When fuel is relocated by the COR package, the radionuclides still residing in the fuel are transported with it. Relocation of the gap

radionuclide mass is not necessary since cladding failure and gap release always occur before fuel relocates (see Section 2.3.2).

In addition to the radioactive masses initially residing in the fuel or fuel-cladding gap, nonradioactive bulk masses in other packages, such as Zircaloy fuel rod cladding, may be released as vapors or aerosols by the RN package release models. Initial inventories for these bulk masses are already available in the appropriate package database and no additional input is needed for the RN package. Release of core or cavity masses by the RN package does not change the mass values in the other packages. For example, the mass of Zircaloy in the COR package is not modified by release of Zircaloy aerosols in the RN package. The errors introduced by this assumption should be very small since the fractions of core and cavity materials that are released as vapors and aerosols are very small. Nevertheless, the user should be aware that mass is not explicitly conserved in this modeling.

The user may also directly specify the initial radionuclide aerosol and/or vapor inventory for any class in any control volume by using the RN1_AG, RN1_AL, RN1_VG, and RN1_VL input record series.

2.3 Release of Radionuclides

Release of radionuclides can occur from the core fuel (with nonradioactive releases from other core structures), from the fuel-cladding gap, and from material in the cavity. At present, no material can be released from the reactions treated in the FDI package. The release models used in each of these areas are discussed below.

2.3.1 Core Release

Radioactive and nonradioactive material may be released from the core. As described in Sections 2.1 and 2.2, the radionuclides residing in the COR package fuel are assumed to be in elemental form and therefore to have only radioactive mass (no associated molecular mass). Upon release from fuel, the total class masses are converted to compound form with a corresponding increase in mass from the added nonradioactive material (e.g., the hydroxide mass in CsOH). By default, the release models are used to calculate the release of radioactive radionuclides from core fuel material (i.e. UO_2) only, which exists in the intact fuel component, in refrozen fuel material on other components and in particulate debris.

In order to apply the release models to core materials other than fuel, such as the fuel rod cladding, the user must change the default values of the core material release multipliers contained in sensitivity coefficient array 7100. For these other core materials, the mapping scheme described in Section 2.1 (with defaults in Table 2.2) determines the apportioning of the core masses among the RN classes, and the entire masses are considered nonradioactive. Hence, by changing the release multiplier for Zr from 0.0 to 0.5, for example, the user obtains half the fractional release rates calculated by the

release correlations for Zr in the cladding, canisters and particulate debris. However, because the mass of structural Zr in the cladding component is enormous compared to the mass of Zr class fission products in the fuel component, the actual release rate (fractional rate times the available mass) from the cladding may be quite large. Because the core release models were developed for fuel releases, their use to calculate the release of structural materials in other components is questionable.

Before cladding failure has occurred, radionuclides released from the fuel in the core are transferred to the gap inventory and are released to the surrounding atmosphere of control volume only upon cladding failure. (However, they are reported by RN output as "released.") After cladding failure, radionuclides released from the core are transferred to the atmosphere of control volumes as specified in the Core package input, which defines channel and bypass control volumes for each core cell. These volumes are used by the RN package as follows:

Core Component	RN Release Volume
Intact:	
Fuel	Channel
Cladding	Channel
Control Rods	Bypass
Canisters	Split Equally Between Channel and Bypass
Conglomerate Debris:	
Refrozen on Cladding	Channel
Refrozen on Control Rods	Bypass
Refrozen on Canisters	Channel
Particulate Debris:	
All	Channel

In addition to releases in the core calculated by the RN package, the reaction modeled in the Core package of B_4C in control rods with steam can release B_2O_3 to the RN package. The class specified on the RN1_DIM input record for B_2O_3 receives this mass in the bypass control volume defined for that core cell.

Three options are currently available for the release of radionuclides from the core fuel component; the CORSOR, CORSOR-M [4] or CORSOR-Booth [5] model may be specified on Input Record RN1_FP00. The CORSOR-BOOTH model contains low and high burn-up options. In addition, the CORSOR and CORSOR-M release rates can be modified to be a function of the component surface-to-volume ratio as compared to a base value, derived from the experimental data on which CORSOR are based. The surface

areas, volumes, and temperatures of the components used in the calculation are obtained from the COR package database. Because none of these radionuclide release models can be considered truly general or universally applicable, it is recommended that concerned users refer to the release model references [4, 5] for a more complete description of modeling assumptions and limitations.

The reduction in release rate of the tellurium class by the presence of unoxidized zirconium can be modeled if desired. The parameters affecting this option are controlled by sensitivity coefficient array 7105 for CORSOR and CORSOR-M and within array 7107 for CORSOR-Booth (see Appendix A). The release rate of Te is reduced by a release rate multiplier (with a default value of 1/40 = 0.025) until the mass of unoxidized intact metal cladding falls below a cut-off fraction (default value of 0.7) of the total mass of intact cladding (including the oxide mass). The default values are based on discussion in Reference [12].

Note that for each core component, the same correlation is used to calculate the release rate for a given class using the individual temperature of that component. That is, the calculation of release of radionuclides from fuel, cladding, canisters, control rods, and particulate debris differs only in the temperature used. Separate correlations for these components are not employed since their form is not compatible with the MELCOR structure.

2.3.1.1 CORSOR

The original CORSOR model correlates the fractional release rate in exponential form,

$$\dot{f} = A \ \ exp(B\,T) \qquad for\ T \geq T_i \tag{2-10}$$

where $\dot{f}$ is the release rate (fraction per minute), A and B are empirical coefficients based on experimental data, and T is the core cell component temperature in Kelvin. Different values for A and B are specified for three separate temperature ranges. The lower temperature limit T_i for each temperature range and the A and B values for that range are defined for each class in sensitivity coefficient array 7101 (see Appendix A). If the cell temperature is below the lowest temperature limit specified, no release is calculated.

2.3.1.2 CORSOR-M

The CORSOR-M model correlates the same release data used for the CORSOR model using an Arrhenius form:

$$\dot{f} = k_o\, exp(-Q/RT) \tag{2-11}$$

The values of k_o, Q, and T are in units of min^{-1}, kcal/mole, and K, respectively. The value of R is 1.987 x 10^{-3} in (kcal/mole)K^{-1}. The values of k_o and Q for each class are implemented in sensitivity coefficient array 7102 (see Appendix A).

2.3.1.3 CORSOR-Booth

The CORSOR-Booth model considers mass transport limitations to radionuclide releases and uses the Booth model for diffusion with empirical diffusion coefficients for cesium releases. Release fractions for other classes are calculated relative to that for cesium. The classical or effective diffusion coefficient for cesium in the fuel matrix is given by:

$$D = D_o \, exp(-Q/RT) \tag{2-12}$$

where R is the universal gas constant, T is the temperature, Q is the activation energy, and the pre-exponential factor D_0 is a function of the fuel burn-up. For fuel with burn-up in excess of 30,000 MWD/MTU the model uses a value for D_0 five times larger than the value it uses for fuels with lower burn-up. The two default values for D_0, the transition burn-up value, and the activation energy Q, based on experimental data for the release of fission gases from fuel test samples [13], are all given in sensitivity coefficient array 7106 (see Appendix A).

The cesium release fraction, f, at time t is calculated from an approximate solution of Fick's law for fuel grains of spherical geometry [14],

$$f = 6\sqrt{\frac{D' t}{\pi}} - 3\,D' t \qquad \text{for } D' t < 1/\pi^2 \tag{2-13}$$

$$f = 1 - \frac{6}{\pi^2}\exp(-\pi^2 D' t) \qquad \text{for } D' t > 1/\pi^2 \tag{2-14}$$

where

$D' t$ = $D t / a^2$ (dimensionless)

a = equivalent sphere radius for the fuel grain

The release rate (in mole/s) of Cs during a time interval t to $t + \Delta t$ from the fuel grain is calculated as:

$$Release\ rate_{Cs} = \frac{[(f \sum D' \Delta t)_{t+\Delta t} - (f \sum D' \Delta t)_t] V \rho}{(1-f)\Delta t} \tag{2-15}$$

where ρ is the molar density of UO_2 in the fuel, V is the fuel volume and the summations are done over the timesteps up to time $(t + \Delta t)$ and t, respectively.

The release rate formulation in the CORSOR-Booth model is also limited by mass transfer through the gas-phase. The gas-phase mass transport release rate from the fuel rod for species k, $\dot{m}_k$, is calculated using an analogy from heat transfer as:

$$\frac{1}{\dot{m}_k} = \frac{D_{fuel} RT}{A_{fuel} Nu\, D_{k,gas} P_{k,eq}} \tag{2-16}$$

where

D_{fuel} = diameter of fuel pellet

A_{fuel} = fuel rod flow contact area

$D_{k,gas}$ = diffusivity of class k in the gas mixture

Nu = Nusselt number (evaluation described in Section 2.2 of the COR reference manual.)

$P_{k,eq}$ = equilibrium vapor pressure of class k at temperature T

The effective release rate for Cs given by Equation (2-16) is a combination of the rates given by diffusion and by gas-phase mass transport. Therefore, the contribution from diffusion only is taken as:

$$DIFF_{Cs} = \left[\frac{1}{\text{Release rate}_{Cs}} - \frac{1}{\dot{m}_{Cs}} \right]^{-1} \tag{2-17}$$

The diffusion release rate for species other than cesium is given by multiplying the cesium release rate by an appropriate scaling factor S_k for each *RN* class k:

$$DIFF_k = DIFF_{Cs} S_k \tag{2-18}$$

Nominal values for S_k are given in sensitivity coefficient array 7103. For certain conditions of cladding oxidation and temperature, the scaling factors must be modified for some classes. When the oxide mass fraction exceeds a critical value F_{k1} and the temperature exceeds a critical value T_{k1}, the class scaling factor is given by:

$$S_k = S_{k1}\, exp(C_k T) \tag{2-19}$$

where T is not allowed to exceed a maximum value T_{max}. When the oxide mass fraction is below a minimum value F_{k2}, the class scaling factor is given by:

$$S_k = S_{k2} \tag{2-20}$$

Values for F_{k1}, T_{k1}, S_{k1}, C_k, T_{max}, F_{k2}, and S_{k2} are all contained within sensitivity coefficient array 7107.

The combined mass transport and diffusion release rate $\dot{m}_{tot,k}$ for class k is then:

$$\dot{m}_{tot,k} = \frac{1}{DIFF_k^{-1} + \dot{m}_k^{-1}} \quad (2\text{-}21)$$

This model assumes that the two processes are in series without any storage capacity between them.

The fractional release rate for the inventory of class *k* is calculated as:

$$\dot{f}_k \text{ (fraction/s)} = \frac{\dot{m}_{tot,k}}{\rho V}\left[F - \frac{P_{k,bulk}}{P_{k,eq}}\right] \quad (2\text{-}22)$$

2.3.1.4 Surface-to-Volume Ratio

In the CORSOR and CORSOR-M release expressions, the effect on the release rate of the surface-to-volume ratio of the material from which release occurs is not treated. An option has been added to include the effect of this ratio as follows:

$$\dot{f} = \dot{f}_{CORSOR(-M)}\ (S/V)_{structure}/(S/V)_{base} \quad (2\text{-}23)$$

where the $(S/V)_{base}$ value has been derived from the original CORSOR data with a value of 422.5 m^{-1} that is stored in sensitivity coefficient array 7104 (see Appendix A). Values for $(S/V)_{structure}$ are calculated from component surfaces and volumes in the Core package (see Section 3 of the COR Package Reference Manual) and thus reflect the effects of core degradation on the surface-to-volume ratios of core components.

2.3.1.5 Generalized Release Model

The generalized release model is an alternate release model that can be easily customized by the user to allow both diffusion and burst component. The cumulative burst fission product release fraction is described by the following equation:

$$FB_{j,i} = a\_burst_j\left(c_0 + c_1 * T_i + c_2 * T_i^2 + c_3 * T_i^3\right) \quad (2\text{-}24)$$

where

T_i = fuel temperature that existed during the time interval Δt_i

c_n = constant coefficients provided by user input

a_burst_j = class j constant coefficient provided by user input.

A cumulative diffusive fission product release fraction is described by the following equation:

$$FD_{j,i} = b\_diff_j\left(FD_{j,i-1} + \left(1 - FB_{j,i-1} - FD_{j,i-1}\right) \cdot \left[1 - e^{-kd_{j,i} \cdot \Delta t_i}\right]\right) \quad (2\text{-}25)$$

where

$FD_{j,i}$ = cumulative fraction of diffusive fission product released up to time t_i

b_diff_j = constant class dependent coefficient provided by user input

$FD_{j,i-1}$ = cumulative fraction of diffusive fission product released up to time t_{i-1}

$FB_{j,i}$ = cumulative fraction of burst fission product released up to time t_i

$FB_{j,i-1}$ = cumulative fraction of burst fission product released up to time t_{i-1}

So $\left(1 - FB_{j,i-1} - FD_{j,i-1}\right)$ is the fission product fraction remaining at time t_i following the current burst release and the previous diffusive releases. $\left[1 - e^{-kd_{j,i} \cdot \Delta t_i}\right]$ is the fractional release due to diffusion during the time interval Δt_i.

where

$kd_{j,i}$ = release rate coefficient for fission product class j calculated using the temperature, T_i, that existed during the time interval Δt_i

$$kd_{j,i} = a\_coeff_j * e^{-b\_diff_j/(RT_i)} \quad (2\text{-}26)$$

where

a_coeff_j = class dependent coefficients provided by user input.

R = 1.987E-3 (kcal/mole)K^{-1}

The total cumulative fission product release fraction at time t_i for fission product j is determined by:

$$F_{j,i} = d\_total_j \cdot \left(FB_{j,i} + FD_{j,i}\right) \quad (2\text{-}27)$$

where

d_total_j = class dependent multiplier provided by user input.

These equations are modified and constrained. The cumulative release fraction cannot exceed the amount of fission product available.

$$FB_{j,i} = FB_{j,i-1} \; and \; FD_{j,i} = FD_{j,i-1} when \; FD_{j,i} \geq 1.0 \quad (2\text{-}28)$$

There is no burst release below TB-init.

$$FB_{j,i} = 0.0 \; when \; T_{i-1} - T_i \leq 0 \qquad (2\text{-}29)$$

The derivative of the cumulative burst release with respect to time cannot be less than zero; if the temperature decreases, the cumulative burst release remains constant.

$$FB_{j,i} = FB_{j,i-1} \; when \; T_{i-1} \geq T_{B-max} \; or \; T_{melting} \qquad (2\text{-}30)$$

The cumulative burst release reaches its maximum when the fuel temperature reaches $T_{B\text{-}max}$ or $T_{melting}$ whichever is lower

$$FB_{j,i} = FB_{j,i} \; when \; T_i \geq T_{B-max} \; or \; T_{melting} \qquad (2\text{-}31)$$

After the fuel melts, fission product release continues in the debris and molten pool and is calculated using the same set of equations.

2.3.1.6 Class Combination at Release

The release model also can provide for the combination of different donor classes into a new class based on the elemental molecular weights. An example is the combination upon release of Cs and I atoms to form CsI molecules, which is modeled by moving stoichiometric amounts of Cs and I mass from the Cs and I classes into a new CsI class. The number of moles of each class that combine is defined by RN1_CLS input data. This combination occurs instantaneously upon release and is only limited by the availability of the released mass during that timestep. If there is an excess of any donor class during the timestep, that excess material stays in the original class. Chemical reactions that take place once release has been completed can be approximated using the models discussed in Section 2.8.

Note: The class combination model is only used for release from the fuel in the core and not for tabular or control function input sources defined using the RN1_AS or RN1_VS records.

2.3.2 Fuel-Cladding Gap

Release of the radionuclides in the fuel-cladding gap (initial inventory plus masses from fuel release) occurs on cladding failure. Cladding failure is assumed to occur if either a temperature criterion is exceeded or if the intact cladding geometry has been lost due to candling or oxidation. It is assumed that the gaps in each radial ring can communicate axially between core cells, so when cladding in one axial level in a radial ring fails, the gap inventory for that entire ring is released. The cladding failure temperature for each core cell is specified on the RN1_GAP00 input record, with a default value of 1173 K (900° C) [15]. The control volume that receives the gap release is the channel control volume associated with the core cell where failure occurs, as defined by the COR_RBV input records (see the COR Package Users' Guide).

2.3.3 Cavity Release

For release of radionuclides from the cavity due to core-concrete interactions, the VANESA model [6] has been implemented in MELCOR and is coupled to CORCON [16] during every timestep. The control volume for cavity releases is specified in the Cavity package input. If a water pool is present, pool scrubbing calculations are performed to apportion the released mass between the pool and the atmosphere.

A number of changes have been made to the stand-alone VANESA program to allow it to function within the MELCOR framework. The major changes are:

(1) The concrete composition used in VANESA is converted from CORCON input using the following mapping, rather than input independently:

CORCON Mass Fraction	VANESA Mass Fraction
CaO + MgO	CaO
Al_2O_3	Al_2O_3
Na_2O	Na_2O
K_2O	K_2O
SiO_2	SiO_2
Fe_2O_3 (converted to FeO) + MnO	FeO
Ti_2O	Ti_2O
Cr_2O_3	Cr_2O_3
Rebar	Fe

(2) To ensure conservation of mass in the calculations, the rate of addition of concrete decomposition products (gases and condensed-phase oxides) is now derived from CORCON results by forward differences, rather than the central difference scheme originally in VANESA.

(3) Core debris masses (and associated radionuclides) may be added as a function of time throughout the transient.

(4) Both radioactive and total masses are tracked. The fraction of the radioactive mass released is assumed to be the same as the fraction of total mass released.

(5) The radioactive inventory is used (by default) to calculate the decay heat in the Cavity package. It is partitioned between the metallic and oxidic phases according to the assumed chemical state of the VANESA class. This partitioning accounts for the difference between elemental mass (e.g., Ba) and compound mass

(e.g., BaO) and the mapping of released structural materials (e.g., Fe) into the nonradioactive portions of RN inventories.

(6) Pool scrubbing calculations are done by the RN package rather than the model in stand-alone VANESA.

2.4 Aerosol Dynamics

This section describes the models used in the RN package to predict the behavior of aerosols during an accident in a LWR. Fission products may be aerosolized as they are released from fuel early in a LWR accident and later expelled from the reactor coolant system. Other events and processes that occur late in the accident, such as core-concrete interactions, pool boiling, direct containment heating, deflagrations, and resuspension may also generate aerosols. High structural temperatures may also result in aerosolization of nonradioactive materials.

The principal aerosol quantities of interest are the mass and composition of aerosol particles and their distribution throughout the reactor coolant system and containment. The calculation of aerosol agglomeration and deposition processes is based on the MAEROS [7] computer code, but without direct inclusion of condensation or evaporation within the MAEROS solution framework. Vapor condensation on and evaporation from aerosol particles are handled separately to reduce the stiffness of the differential equation set and to ensure consistency with the calculation of these processes by other models and packages, as described later.

MAEROS is a multisectional, multicomponent aerosol dynamics code that evaluates the size distribution of each type of aerosol mass, or *component*, as a function of time. This size distribution is described by the mass in each size bin, or *section*. Each section may have a different chemical composition as described by the masses of various components for that section. In other words, a section is an aerosol size group and a component is a particular type of aerosol material. Since MELCOR operates on a radionuclide class structure, as discussed earlier, a mapping between RN classes and MAEROS aerosol components must be specified by the user.

Figure 2.1 illustrates the sectional representation of a two-component aerosol with five sections. The mass concentrations of component 1 in the five sections are given by the stair-stepped line that bounds the lower crosshatched region. The total aerosol mass concentrations in the five sections are given by the uppermost stair-stepped line. Therefore, the mass concentrations of component 2 in the five sections are given by the upper shaded region.

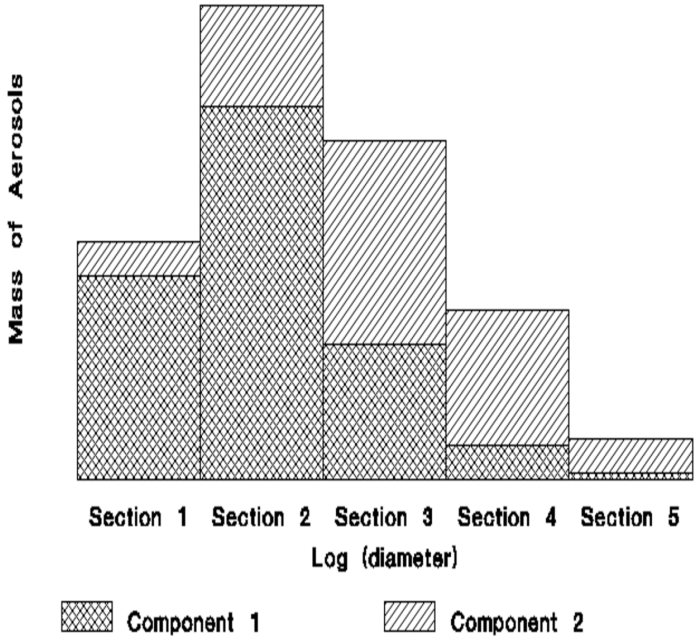


Figure 2.1 MAEROS Aerosol Model

One powerful feature of MELCOR is that water condensation onto and evaporation from aerosols is modeled in a manner consistent with the thermal/hydraulic calculations in the CVH and HS packages. That is, the latent heat associated with the coolant mass transfer between the atmosphere and aerosol surfaces is incorporated in the total internal energy transfer to and from the atmosphere. In addition, condensation and evaporation of fission product vapors onto aerosols is calculated in parallel with condensation onto and evaporation from heat structure surfaces, but without consideration of the latent heat of condensation of the vapor, since it is negligible compared to the energy of the atmosphere and the heat structure.

The MELCOR calculation of changes in aerosol distribution and location within a plant considers the following general processes:

(1) aerosol phenomenological sources from other packages, such as release from fuel rods or during core-concrete interactions, and/or arbitrary user-specified sources;

(2) condensation and evaporation of water and fission products to and from aerosol particles;

(3) particle agglomeration (or coagulation), whereby two particles collide and form one larger particle;

(4) particle deposition onto surfaces or settling through flow paths into lower control volumes;

(5) advection of aerosols between control volumes by bulk fluid flows; and

(6) removal of aerosol particles by engineered safety features (ESFs), such as filter trapping, pool scrubbing, and spray washout.

The RN package includes models to simulate each of these processes, but only user-defined aerosol sources and agglomeration and deposition processes are formally coupled in the MAEROS integrated solution framework. Aerosol sources from other phenomenological packages in MELCOR and condensation onto and evaporation from aerosols are decoupled and treated outside the MAEROS solution. This section describes the details of the implementation of MAEROS within MELCOR. Section 2.4.1 describes in more detail how the component/class mapping scheme works and how the particle size distribution is represented in MELCOR. The general MAEROS equations and the specific models for aerosol agglomeration and deposition are described in Section 2.4.2. Section 2.4.3 provides information on how various aerosol sources are treated, and Section 2.4.4 discusses the MELCOR aerosol resuspension model.

Condensation and evaporation processes for both aerosols and heat structure surfaces are described later in Section 2.5, and Section 2.10 describes the modifications for hygroscopic aerosols. Advection of aerosols between control volumes is based on transport with the hosting fluid (pool or atmosphere) without slip. Section 2.7 describes the removal of aerosols by ESFs.

2.4.1 Aerosol Mass and Size Distributions

In MELCOR, one or more RN classes can be assigned to a component, as specified on the RN1_CC input records, but a particular class cannot be assigned to more than one component. For each control volume, the fractions within a particular component of each class assigned to that component are determined before the aerosol dynamics calculation is performed to determine the new size distribution. These fractions necessarily sum to unity. After the aerosol dynamics calculation, the masses for each aerosol size, the deposited masses, and the fallout masses for each class are determined by multiplying the appropriate component mass values by the previously calculated class mass fraction. In effect, all classes assigned to the same component are assumed to have the same size distribution.

The aerosol particle size distribution is discretized into particle size bins called *sections*. The distribution of aerosol mass within a section is treated as constant with respect to the logarithm of particle mass. The user may input any arbitrary initial aerosol size distribution for any fission product class by specifying the mass in each size section at the initial time (see the RN1_AG and RN1_AL input records). The initial aerosol water mass (fog) is determined from the CVH package input data only and is put in the smallest aerosol section; an error message is generated if an attempt is made to initialize water aerosol mass through RN input.

The numbers of sections and components to be used in the aerosol calculations, as well as the minimum and maximum aerosol diameters, are specified by the user (see input records RN1_DIM and RN1_ASP). Individual section boundaries are calculated from these values so that the ratio of the upper and lower bound diameter of each section is the same. A check is also made that the ratio of the upper to lower mass boundary for each section is greater than or equal to two to assure that the calculations conform to the assumptions made in the derivation of the MAEROS equations. If this constraint is not met, an error message is generated and the calculation terminates.

Although the aerosol component distributions from the MAEROS calculation are not stored permanently, the class distributions are used to calculate the mass median diameter and geometric standard deviation for the wet, dry and component distributions in each control volume for editing. The wet distribution is the sum over all classes including water; the dry distribution, which is commonly determined experimentally, is the sum over all classes excluding water; and the component distribution is the sum over all classes assigned to the component. The mass median diameter is defined to be the diameter above and below which half the total mass (wet, dry or component mass) in the distribution occurs,

$$0.5 \times \int_0^{\infty} f_m(D)dD = \int_0^{D'} f_m(D)dD \qquad (2\text{-}32)$$

where D' is the mass median diameter and $f_m(D)\ dD$ is the mass in the distribution between diameter D and $D + dD$. The geometric standard deviation, σ_G, is defined as:

$$(ln\,\sigma_G)^2 = \frac{\int_0^{\infty} ln^2(D/\bar{D})\, f_m(D)dD}{\int_0^{\infty} f_m(D)dD} \qquad (2\text{-}33)$$

where $\overline{D}$ is the logarithmic mass mean diameter defined by:

$$\ln\left(\overline{D}\right) = \frac{\int_0^{\infty} \ln(D) f_m(D)dD}{\int_0^{\infty} f_m(D)dD} \qquad (2\text{-}34)$$

In MELCOR, any aerosol particles that are calculated to grow larger (by agglomeration or condensation) than the maximum size section, are assumed to *fall out* onto either floor-type heat structures or into adjacent lower control volumes (apportioned by heat structure or flowthrough area). Aerosols that fall out into a lower control volume are put in the largest size section of the aerosol distribution in that control volume and thus should quickly deposit or fall out onto floor structures. This is described in more detail in Section 2.4.2.2.

2.4.2 MAEROS Equations

The aerosol agglomeration and deposition models from MAEROS are used to calculate the changing aerosol size distributions as these processes affect the aerosol in each control volume at each timestep. Particle agglomeration, deposition onto heat structure surfaces, fallout onto floors or into lower control volumes, and the effects of user-defined aerosol sources are all integrated in the MAEROS calculation.

The modeling of the aerosol size distribution is governed by a complex integro-differential equation. MAEROS was developed as a method of discretizing this equation into a form that can be solved numerically. In their method (and using their notation), the full range of aerosol masses is divided into *m* contiguous arbitrarily sized sections, and Q_ℓ is defined as the total mass of aerosol per unit volume of fluid in section ℓ at time *t*. Thus,

$$Q_\ell(t) = \sum_{k=1}^{s} Q_{\ell,k}(t) \tag{2-35}$$

where $Q_{\ell,k}(t)$ is the mass of component *k* in section ℓ, and *s* is the total number of components. The upper bound of section $\ell - 1$ is equal to the lower bound of section ℓ for ℓ = 2, 3, ... *m*. These equations can be written

$$\begin{aligned}
\frac{dQ_{\ell,k}}{dt} \;=\; & \tfrac{1}{2}\sum_{i=1}^{\ell-1}\sum_{j=1}^{\ell-1}\left[{}^{1a}\overline{\beta}_{i,j,\ell}\, Q_{j,k} Q_i + {}^{1b}\overline{\beta}_{i,j,\ell}\, Q_{i,k} Q_j \right] \\
& - \sum_{i=1}^{\ell-1}\left[{}^{2a}\overline{\beta}_{i,\ell}\, Q_i Q_{\ell,k} - {}^{2b}\overline{\beta}_{i,\ell}\, Q_\ell Q_{i,k} \right] \\
& - \tfrac{1}{2}{}^{3}\overline{\beta}_{\ell,\ell}\, Q_\ell Q_{\ell,k} - Q_{\ell,k}\sum_{i=\ell+1}^{m} {}^{4}\overline{\beta}_{i,\ell}\, Q_i + {}^{1}\overline{G}_{\ell,k} Q_\ell \\
& - \sum_{i=1}^{N_a}\left[{}^{2}\overline{G}_{\ell,k} Q_{\ell,k} - {}^{2}\overline{G}_{\ell\pm1,i} Q_{\ell\pm1,k} \right] + {}^{3}\overline{G}_{\ell\pm1,k}\, Q_{\ell\pm1} + \overline{S}_{\ell,k} - \overline{\Re}_{\ell,k}
\end{aligned} \tag{2-36}$$

where

$dQ_{\ell,k}(t)/dt$	=	time rate of change of aerosol mass of component *k* (per unit volume) in section ℓ at time *t*
k	=	aerosol component (for example, water or a specific FP) = 1, 2,...N_a
ℓ	=	discretized section (or physical size range) of the aerosol = 1, 2, 3,...*m*
$\ell \pm 1$	=	$\ell - 1$ for condensation, or ℓ + 1 for evaporation

Each term in Equation (2-36) represents a distinct mechanism for changes in mass concentration of component *k* in a particular section. Time integration of Equation (2-36)

requires that the coefficients used in each term be known on a sectional basis. These sectional coefficients correspond to the following mechanisms:

$\overline{\beta}$ = agglomeration (or coagulation), m^3/s-kg

$\overline{G}$ = gas-to-particle conversion (condensation/evaporation), s^{-1}

$\overline{S}$ = sources, kg/m^3-s

$\overline{\Re}$ = removal (deposition) kg/m^3-s

The $\overline{\beta}$'s are called sectional coagulation coefficients, and they can be evaluated by using a variety of formulas that incorporate the effects of the different physical processes. These processes include gravitational agglomeration (a larger particle overtakes a smaller one as they both fall) and agglomeration through diffusion (either Brownian or turbulent), and are described in more detail in Section 2.4.2.1. The six agglomeration terms in the Gelbard-Seinfeld approach refer respectively to the following processes:

1a $\overline{\beta}_{i,j,\ell}$	addition of component *k* in section ℓ, by removal of component *k* in section *j* when a particle in section *j* coagulates with a particle in section *i* to form a particle in section ℓ.
1b $\overline{\beta}_{i,j,\ell}$	addition of component *k* in section ℓ, by removal of component *k* in section *i* when a particle in section *i* coagulates with a particle in section *j* to form a particle in section ℓ.
2a $\overline{\beta}_{i,\ell}$	removal of component *k* in section ℓ, resulting from a particle in section *i* coagulating with a particle in section ℓ.
2b $\overline{\beta}_{i,\ell}$	addition of component *k* in section ℓ, resulting from a particle in section *i* coagulating with a particle in section ℓ, with the resulting particle remaining in section ℓ.
3 $\overline{\beta}_{\ell,\ell}$	removal of component *k* in section ℓ, by two particles in section ℓ coagulating and the resulting particle is in a section higher than ℓ.
4 $\overline{\beta}_{i,\ell}$	removal of component *k* in section ℓ, by a particle in section ℓ coagulating with a particle in section *i*, where i > ℓ.

The four condensation terms represented by the $\overline{G}$ coefficients correspond to the following processes:

${}^1\overline{G}_{\ell,k}$	addition (removal) of component *k* within section ℓ by condensation (evaporation) of component *k* onto (from) particles in that section;
$2\overline{G}_{\ell,j}$	transfer of existing component *k* from section ℓ to section $\ell + 1$ ($\ell - 1$) by condensation (evaporation) of component *i* onto (from) particles in section ℓ;
${}^2\overline{G}_{\ell\pm1,j}$	transfer of existing component *k* from section $\ell - 1$ ($\ell + 1$) to section ℓ by condensation (evaporation) of component *i* onto (from) particles in section $\ell - 1$ ($\ell + 1$); and
${}^3\overline{G}_{\ell\pm1,k}$	transfer of changed mass of component *k* from section $\ell - 1$ ($\ell + 1$) to section ℓ by condensation (evaporation) of component *k* onto (from) particles in section $\ell - 1$ ($\ell + 1$). This term vanishes in the limit that aerosol masses are large compared to molecular masses.

Water condensation onto and evaporation from aerosol particles are the principal couplings between thermal-hydraulics and aerosol behavior. However, these terms are not used directly in the MELCOR implementation of MAEROS. As described in Section 2.5.1, water condensation and evaporation are treated separately (but still using the MAEROS-calculated coefficients for water, as discussed in Section 2.5.1) for consistency with the water thermodynamics calculated in the CVH package.

Furthermore, fission product condensation onto and evaporation from aerosols are also integrated with the calculation of fission product condensation and evaporation on heat structure surfaces by the TRAP-MELT model, as described in Section 2.5.2, and are thus treated outside the MAEROS framework as well.

In Equation (2-36), particle removal (or deposition) is addressed by the $\overline{\mathfrak{R}}$ term. Deposition occurs through a number of processes, including gravitational settling, diffusion to surfaces, thermophoresis (a Brownian process causing migration of particles toward lower temperatures), and diffusiophoresis (deposition induced by condensation of water vapor onto structural surfaces). The sectional deposition coefficients are described in more detail in Section 2.4.2.2.

Aerosol sources are included by the $\overline{S}$ term in Equation (2-36). Currently, only sources defined by the user as tabular functions of time are directly included in the MAEROS equations. Sources from phenomenological models are added directly to the aerosol sectional distributions as described later in Section 2.4.3.

Intraparticle chemical reactions can occur between constituents of the aerosol. The modeling of aerosol size/composition changes resulting from chemical reactions is not currently implemented in MELCOR, but this phenomenon could easily be included in the sectional model.

Simplifications in the coefficients and in Equation (2-36) occur if the geometric constraint

$$m_{i+1}/m_i > 2 \qquad (2\text{-}37)$$

is satisfied, where m_i is the particle mass at the lower boundary of section *i*. The geometric constraint ensures that the agglomeration of two particles results in a new particle that fits into either the section that contains the larger of the two original particles or the section just above it. This constraint thus reduces the number of sectional agglomeration coefficients. As stated earlier in Section 2.4.1, input specifying the section boundaries is checked to verify that this constraint is met.

Equation (2-36) is used in MELCOR to describe the evolution of the aerosol size and composition distributions within each control volume. Each control volume has its own particle size and chemical composition distributions, and the aerosols are carried from one control volume to another by gas flow and may be removed by ESFs, as described in Section 2.7.

2.4.2.1 Agglomeration

When two aerosol particles collide, they can combine to form a larger particle. This process is known as agglomeration or coagulation. The sectional method used in MAEROS treats four agglomeration processes: Brownian diffusion, differential gravitational settling, and turbulent agglomeration by shear and inertial forces. A basic assumption about these processes is that simultaneous agglomeration of three or more particles is negligible.

The full dependence of the agglomeration coefficients β (m^3/s) upon the aerosol and atmosphere properties as implemented in MELCOR is given in the equations in Appendix B. The dependence on atmosphere properties is not considered to be a major source of uncertainty in the aerosol calculations. The dependence on particle diameter and key modeling parameters can be summarized as follows:

Brownian:	$\beta_B \propto \gamma\, \chi^{-1}\, f\left(d_i, d_j\right)$
Gravitational:	$\beta_{grav} \propto \varepsilon_g\, \gamma^2\, \chi^{-1} \left(d_i + d_j\right)^2 \left(d_i^2 - d_j^2\right)$
Turbulent, Shear:	$\beta_{Tl} \propto \gamma^3\, \varepsilon^{1/2} \left(d_i + d_j\right)^3$
Turbulent, Inertial:	$\beta_{T2} \propto \gamma^2\, \chi^{-1}\, \varepsilon^{3/4} \left(d_i + d_j\right)^2 \left(d_i^2 - d_j^2\right)$

In these proportionalities, γ and χ are the agglomeration and dynamic shape factors, respectively, and ε is the turbulent energy dissipation density, all of which are specified on user input records RN1_MS00 and RN1_MS01. Variables d_i and d_j are the diameters of the two interacting particles, with $d_i > d_j$. The collision efficiency for gravitational agglomeration is represented by ε_g, with a specific value (discussed below) calculated in the code. The magnitude of the Brownian kernel increases with increasing values of the size ratio d_i / d_j. The role of the various parameters appearing in the kernels is also discussed below.

Except when they include significant amounts of liquid, aerosol particles are not usually assumed to be spherical, and the effective aerosol densities may be significantly less than the bulk density of the materials of which the aerosols are composed. In aerosol codes, these effects may be taken into account by using a formalism based on fully dense spherical aerosols modified through the use of the agglomeration shape factor γ and the dynamic shape factor χ. The shape factors γ and χ are input by the user to represent the effect of nonspherical shape upon aerosol collision cross sections and aerosol-atmosphere drag forces, respectively. Unit values of the shape factors correspond to dense aerosol of spherical shape, while porous spherical agglomerates lead, in theory, to values somewhat greater than unity. Highly irregular aerosols and agglomerates can have shape factors substantially greater than unity, often with γ and χ being quite unequal.

Given experimental data for aerosol shapes and densities applicable to LWR accidents, shape factors could, in principle, be derived. Because this is not practical, empirical values are obtained by fitting code calculations to the results of aerosol experiments. The values obtained may be sensitive to aerosol composition and to atmospheric conditions, especially to relative humidity. Humid conditions tend to produce more nearly spherical aerosols due to condensation of water onto aerosol agglomerates. Only limited information is available concerning the dependence of shape 4factors upon the relevant parameters (for example, particle characteristics and atmospheric conditions), and these parameters are themselves quite uncertain under accident conditions. Default values of unity are set for both factors in MELCOR.

Agglomeration rates can be enhanced by turbulence in the atmosphere. In the past, very little attention has been given to estimating values of turbulent energy dissipation density ε appropriate for accident conditions, and uncertainty in its value may contribute to uncertainty in the aerosol agglomeration rates. In MELCOR, the user can input the value of ε or use the default value of 0.001 m^2/s^3.

The gravitational collision efficiency ε_g of unity corresponds to the assumption that collision cross sections are equal to the geometric cross sections. It is well known that hydrodynamic interactions between particles (i.e., the tendency of a particle to follow streamlines in flow around another particle) can yield collision efficiencies much less than

unity, especially for particles that are unequal in size. The problem of collisions between falling (spherical) aerosols has been the object of much detailed theoretical and experimental study, and may be more complex than can be represented by the simple expressions normally used in aerosol codes. In MELCOR, the value of ε_g is given by

$$\varepsilon_g = 1.5 d_j^2 / \left(d_i + d_j\right)^2 \qquad (2\text{-}38)$$

where d_j is the smaller of the two aerosol particle diameters. It has been argued [17, 18] that using 0.5 instead of 1.5 as the coefficient in Equation (2-38) gives a better representation and that other corrections are needed when the size ratio d_i/d_j is less than about 2 and/or d_i is greater than about $20 \mu m$. However, more recent experimental measurements of collision efficiencies by Gelbard et al. [19] do not support these proposed revisions and, instead, gave collision efficiencies in reasonable agreement with Equation (2-38). These measurements involved studying the collisions of spheres at higher Reynolds numbers than those typical of aerosols and the results therefore may not be totally conclusive; however, arguments for modifying Equation (2-38) are not judged to be any more convincing.

The agglomeration model used in MELCOR receives temperature, pressure, and mass flow rate information from the CVH package. The turbulent agglomeration kernels are combined as

$$\beta_{TT} = c_s \left(\beta_{T1} + \beta_{T2}\right)^{1/2} \qquad (2\text{-}39)$$

where c_s is a particle sticking coefficient (default value of unity), which may be specified on input record RN1_MS00. (This sticking coefficient also appears in the other Brownian and gravitational agglomeration kernels.) The total turbulent kernel is added to the Brownian and gravitational kernels to obtain a total agglomeration kernel β_T which is then integrated over sections for use in Equation (2-36):

$$\beta_T = \beta_B + \beta_{grav} + \beta_{TT} \qquad (2\text{-}40)$$

Examination of the relations for the agglomeration kernels in the proportionalities given above shows that the effects of gravitational collision efficiency, aerosol shape factors, and turbulence are coupled together in a highly nonlinear fashion. The dependence upon the various parameters differs among the different agglomeration mechanisms, and the net effects are strongly size-dependent. Hence, it is possible to give only a few generalizations.

All the agglomeration processes are enhanced by large values of the agglomeration shape factor γ, with the effect being largest for turbulent shear agglomeration and smallest for Brownian agglomeration. Large values of the dynamic shape factor reduce all the kernels (calculational coefficients) except the turbulent shear kernel, which is

unaffected. Hence, large values of the shape factors enhance the relative importance of turbulence, especially for the turbulent shear effect. Reference [18] includes sensitivity studies examining the implications of uncertainties in these shape factors as well as in the turbulent energy dissipation density ε.

2.4.2.2 Deposition, Settling, and Fallout

Aerosols can directly deposit onto heat structure and water pool surfaces through five processes calculated within MAEROS. All heat structure surfaces are automatically designated as deposition surfaces for aerosols using information from the HS package, unless made inactive through user input. The parameters obtained from the HS package are:

(1). Geometric orientation
(2). Surface area in the atmosphere
(3). Surface heat flux
(4). Mass transfer coefficient
(5). Water condensation mass flux

Each surface of a MELCOR heat structure must be designated as a ceiling, a floor, or a wall, since MAEROS only calculates deposition kernels for these orientations. The default treatment is:

> The upper surface of a rectangular heat structure with an angle of inclination less than 45 degrees is considered to be a floor, and the lower surface a ceiling. The heat structure orientation parameter ALPHA on HS Input Record HS_EOD determines both the inclination and whether the “left” surface is the upper or the lower surface.
>
> Both surfaces of a rectangular heat structure with an angle of inclination greater than 45 degrees, and both surfaces of vertical cylinders and spheres are treated as walls.
>
> The inner (left) surface of a bottom-half hemisphere is treated as a floor and the outer (right) surface as a ceiling. For a top-half hemisphere, the treatment is reversed.

The user can override these default orientations or deactivate a surface for aerosol deposition through the RN1_DS input records. However, if the surface of a structure is deactivated for the purposes of deposition, it is also removed from consideration in the calculation of condensation and evaporation of fission product vapors, as discussed in Section 2.5. (Note that the orientation of a structure does not otherwise affect the rate of condensation or evaporation.)

If a control volume contains a water pool, the pool surface is treated as a floor for the purposes of deposition. The area of the water pool is extracted from the CVH database.

Aerosols can also *settle* from one control volume to another through *flowthrough areas* (i.e., the gravitational settling and Brownian diffusion kernels in MAEROS described below are applied to flowthrough areas in addition to HS and pool surfaces). Such areas ordinarily correspond to open flow paths between the control volumes, through which aerosols and radionuclide vapors are also advected. The appropriate flow areas, path elevations, etc., are specified in the RN1_SET input records. Aerosols are not transported through these areas if the flow path is blocked by a water pool.

Finally, aerosols can agglomerate and become larger than the user-specified maximum diameter. These aerosols are assumed to immediately deposit onto water pools or horizontal heat structure surfaces or to settle from one control volume to another through *flowthrough areas* defined as part of RN input. The term *fallout* in MELCOR is used exclusively for this immediate deposition or settling of aerosols larger than the maximum user-specified diameter. All control volumes must have at least one upward-facing deposition surface (floor) or flowthrough area defined to receive fallout aerosols generated by this mechanism. During MELGEN a check is made for the existence of at least one such area; if none is present, an error message is generated and no restart file is written.

The MAEROS deposition kernel for each type of surface is made up of four contributions: gravitational deposition, Brownian diffusion to surfaces, thermophoresis, diffusiophoresis, and turbulent depositon. Of these natural depletion processes, gravitational deposition is often the dominant mechanism for large control volumes such as those typically used to simulate the containment, although phoretic effects may be significant in some cases (e.g., diffusiophoresis during water condensation). Particle diffusion is generally considered to be a relatively unimportant deposition process. The contribution of each of these processes to the deposition kernel for each type of heat structure surface and for pools and flowthrough areas in MELCOR is summarized below:

Surface	**Deposition Kernel**[1]			
	grav	**BD**	**therm**	**diffus**
Heat Structure				
Floor	+	+	+	+
Wall	0	+	+	+
Ceiling	-	+	+	+
Pool	+	+	+[2]	+[2]
Flowthrough Area	+	+	0	0

[1] The symbols +, 0, and - mean a positive contribution, no contribution, and a negative contribution, respectively. Of course, the total deposition kernel for any surface can not be less than zero.

[2] Included in the general formulation but currently zeroed out internally.

The velocities calculated for each of these deposition processes are defined below.

Gravitational Deposition

Gravitational deposition is effective only for upward-facing surfaces (i.e., floors and water pools) and flowthroughs to lower control volumes; for downward-facing surfaces (i.e., ceilings), this mechanism works to oppose other deposition processes. The gravitational deposition velocity is given by

$$v_{grav} = \frac{d_p^2 \rho_p g C_m}{18 \mu \chi}$$

where

v_{grav} = the downward terminal velocity (m/s)

d_p = the particle diameter (m)

ρ_p = the particle density (kg/m^3)

g = acceleration of gravity = 9.8 m/s^2

C_m = the particle mobility, or Cunningham slip correction factor, which reduces the Stokes drag force to account for noncontinuum effects

The particle mobility, or Cunningham slip correction factor, in the equation above is expressed as

$$C_m = 1 + \frac{2\lambda}{d_p}\left[F_{slip} + 0.4 \exp\left(-1.1 d_p / 2\lambda\right)\right] \qquad (2\text{-}41)$$

where

λ = mean free path of air at 298 K (~ $0.069 \bullet 10^{-6}$m)

F_{slip} = slip factor specified on Input Record RN1_MS00 (default value of 1.257)

μ = viscosity of air at 298 K [~1.8×10^{-5}(N-s/m^2)]

χ = dynamic shape factor

This model assumes that the aerosol particle Reynolds number Re, based on particle diameter and net deposition velocity, is much less than 1. This physically means that inertial effects of the flow may be neglected. This Reynolds number is not to be confused with the bulk mass flow (air, steam, aerosol particles) Reynolds number based on the dimensions and velocities calculated by the CVH package, which is typically much greater than 1.

Brownian Diffusion

Deposition can also result from diffusion of aerosols in a concentration gradient from a higher to a lower concentration region. The diffusive deposition velocity is given by

$$v_{diff} = \frac{\sigma T C_m}{3\pi \mu \chi d_p \Delta} \qquad (2\text{-}42)$$

where

v_{diff} = diffusion deposition velocity (m/s)

σ = Boltzmann constant = $1.38 \bullet 10^{-23}$ (J/s-m^{2}K^4)

T = atmosphere temperature (K)

μ = viscosity (N • s/m^2)

χ = dynamic shape factor

Δ = user-specified diffusion boundary layer thickness specified on input record RN1_MS01 (default value of 10^{-5} m)

The assumption is that there is no gas velocity perpendicular to the deposition surface. This impaction mechanism is most effective for larger aerosol particle sizes.

Thermophoresis

This aerosol deposition mechanism results from the force exerted on aerosol particles by temperature gradients in the bulk gas. The thermophoretic deposition velocity v_{therm} is given by

$$v_{therm} = \frac{3 \mu C_m \left(c_t \, Kn + k_{gas}/k_p\right)}{2 \chi \rho_{gas} T \left(1 + 3 F_{slip} Kn\right)\left(1 + 2 c_t \, Kn + k_{gas}/k_p\right)} \nabla T \qquad (2\text{-}43)$$

where

Kn = $2\lambda / d_p$ (Knudsen number)

k_{gas}/k_p = ratio of thermal conductivity of gas over that for aerosol particle k_p, and is user-specified (on Input Record RN1_MS01)

∇T = structure surface temperature gradient (K/m)

ρ_{gas} = gas density (kg/m^3)

T = wall temperature (K)

F_{slip} = slip factor

c_t = constant associated with the thermal accommodation coefficients (specified on Input Record RN1_MS01 with default value of 2.25)

The coefficient of ∇T in Equation (2-43) is calculated for each of the four aerosol coefficient sets at minimum/maximum temperature and pressure and stored as described in Section 2.4.2.3. The actual temperature gradient at each heat structure surface, calculated from the heat flux q'' obtained from the HS package as

$$\nabla T = -q''/k_{air} \qquad (2\text{-}44)$$

is used with an interpolated coefficient (see Section 2.4.2.3) to calculate the actual diffusion velocity. The thermal conductivity of air, k_{air}, is evaluated at the surface temperature of the heat structure using the properties of air for consistency with the evaluation of the aerosol coefficients with air properties.

Diffusiophoresis

When water condenses on (evaporates from) a structure surface, composition gradients exist in the adjacent gas that affect aerosol deposition on the surface. Two related mechanisms produce these gradients. First, a net molar flux of gas toward (away from) the condensing (evaporating) surface exists, and this net flux, commonly called the Stefan flow [20], tends to move aerosol particles with it. Second, differences in the momentum transferred by molecular impacts on opposite sides of the particle tend to drive the particle in the direction of decreasing concentration of the heavier constituent. By some definitions, only this second component constitutes diffusiophoresis; however, in this discussion the term “diffusiophoresis” is used to represent the net result of both effects and the equations given include both effects. Note that when the noncondensible gas is heavier than steam, as in air-steam mixtures, the differential molecular impact effect opposes the Stefan flow (which dominates the net result); the effects are in the same direction if the noncondensible gas is lighter than steam.

The treatment in MELCOR is valid for particle sizes large compared with molecular mean free paths, a condition that generally applies for accident analyses. A diffusiophoretic deposition velocity (including the Stefan flow) $v_{diffusio}$ is calculated from

$$v_{diffusio} = \left(\frac{\sqrt{M_s}}{X_s \sqrt{M_s} + X_{NC} \sqrt{M_{NC}}} \right) \left(\frac{W_{cond}}{\rho_b} \right) \text{ if } W_{cond} \geq 0 \text{ (condensation)} \qquad (2\text{-}45)$$

$$v_{diffusio} = W_{cond} / \rho_s \qquad \text{if } W_{cond} < 0 \text{ (evaporation)} \qquad (2\text{-}46)$$

where

M_s = molecular weight of water (kg/mole)

M_{NC} = molecular weight of noncondensible gases (air

W_{cond} = condensation mass flux to the surface (kg/s-m^2)

ρ_b = density of bulk gas (kg/m^3)

ρ_s = saturation density of water vapor (kg/m^3)

X_s = mole fraction of water vapor in the bulk gas

X_{NC} = mole fraction of noncondensible gases in the bulk gas

The condensation mass flux is obtained from the HS package. Note that the differential molecular impact effect is ignored in MELCOR for evaporation ($W_{cond} < 0$). The velocity calculated is toward the surface for condensation and away from the surface for evaporation.

Turbulent Deposition

Turbulent deposition may be important for high Re flow in pipes and in bends. Turbulent deposition is only available for heat structure surfaces (no pool surfaces) and is only calculated when specified by the user. Several optional models are available to the user for specifying turbulent deposition on the surface of a heat structure. These options include both deposition on straight pipes as well as on bends.

Turbulent deposition is greatly dependent on the inertia of aerosol particles and Is therefore characterized over three ranges of particle sizes, inertia moderated regime, eddy diffusion impaction regime, and turbulent particle diffusion regime. For very small particles (turbulent particle diffusion regime) Brownian motion is important to transport particles across the viscous sub layer. For the eddy diffusion-impaction regime (larger particles) deposition is dominated by eddy diffusion where particles are accelerated to the wall due to turbulent eddies in the core and buffer layer and coast across the viscous sub layer. For the inertia moderated regime, very large particles are subject to reduced acceleration by the turbulent core but little or no acceleration to small eddies in the buffer near the wall.

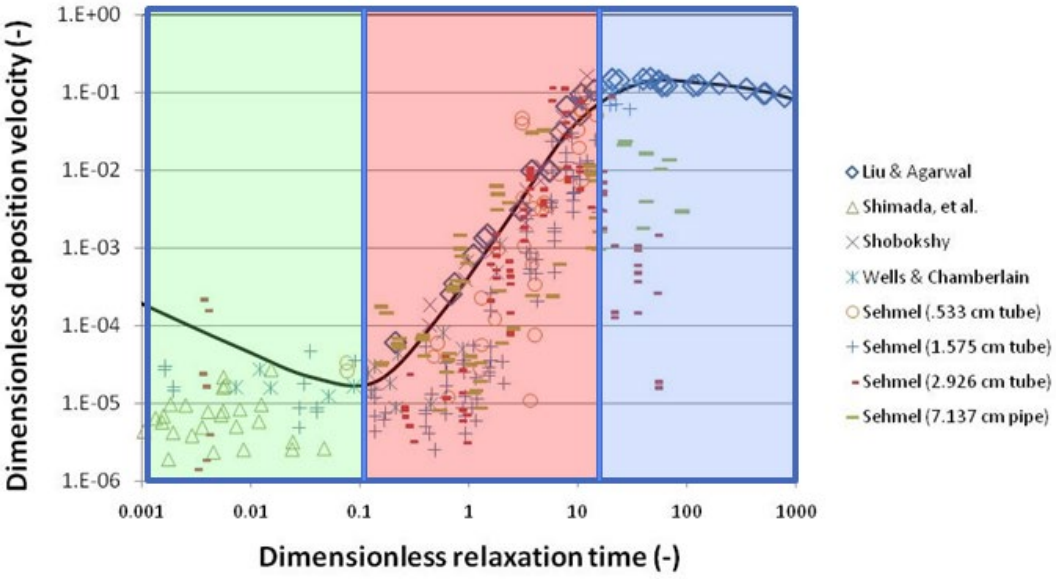


Figure 2.2 Correlation of deposition velocity as a function of relaxation time with experimental data for three regimes: (green) turbulent particle diffusion, (red) eddy diffusion impaction, (blue) Inertia moderated regime

2.4.2.2.1.1 Turbulent Deposition in straight pipes

Wood's model for pipes

Wood developed a semi-empirical for predicting turbulent deposition on surfaces. His model characterizes deposition over the three deposition regimes that are characteristic of particle size. In the turbulent particle diffusion regime Brownian diffusion is important and deposition occurs by a combination of Brownian and eddy diffusion. Davies [21] proposed the following equation for the deposition velocity in this regime:

$$u_{t,s} = \frac{Sc^{-2/3}\tilde{v}}{14.5\left\{\frac{1}{6}1n\left[\frac{(1+\varphi)2}{1-\varphi+\varphi^2}\right]+\frac{1}{\sqrt{3}}atan\left[\frac{2\varphi-1}{\sqrt{3}}\right]+\frac{\pi}{\sqrt{3}}\right\}} \quad (2\text{-}47)$$

Where

$u_{t,s}$ = turbulent deposition velocity for submicron particles(m/s)

$\tilde{v}$ = friction velocity (m/s), defined by the following expression:

$$\tilde{v} = U\sqrt{\frac{f}{2}} \quad (2\text{-}48)$$

f = Fanning friction factor (dimensionless)

φ = $SC^{1/3}/2.9$

Wood found that for particles that are order of the mean free path or greater, this equation could be approximated by:

$$V_d^* = \frac{u_{t,s}}{\tilde{v}} = \frac{3\sqrt{3}}{29\pi} Sc^{-2/3} \quad (2\text{-}49)$$

Where the deposition velocity is non-dimensionalized by the friction velocity. In terms of the dimensionless relaxation time, τ^*, this can be written:

$$V_d^* = \frac{3\sqrt{3}}{29\pi\tau_*^{1/3}} Sc^{-2/3}\ \tau_*^{1/3} \quad (2\text{-}50)$$

As particle size increases impaction increases and a second term is added to this equation:

$$V_d^* = \frac{3\sqrt{3}}{29\pi\tau_*^{1/3}} Sc^{-2/3}\ \tau_*^{1/3} + K\tau_*^2 \quad (2\text{-}51)$$

The coefficient, K, is derived by solving a diffusion equation written in the form of a turbulent version of Fick's law, i.e.,

$$N = \left(D_p + \varepsilon\right)\frac{dc}{dy} \quad (2\text{-}52)$$

where

N = particle flux ($\#/m^2$-s)

D_p = particle diffusion coefficient (m^2/s)

ϵ = particle turbulent eddy-diffusivity (m^2/s)

c = particle concentration ($\#/m^3$)

y = distance from surface (m)

Wood's model for smooth pipes

For smooth pipes, Wood [22] proposed the following approximation for the deposition velocity:

$$V_d^* = \frac{3\sqrt{3}}{29\pi\tau_*^{1/3}} Sc^{-2/3}\ \tau_*^{1/3} + 0.00045\tau_*^2 \quad (2\text{-}53)$$

For large particles, τ^* >10, particle inertia becomes important in the inertia moderated regime and the deposition velocity becomes constant, though dependent on the Reynold's number through the friction factor:

$$V_d^* = \sqrt{\frac{f}{2}} \quad 10 \le \tau_* < 270 \quad (2\text{-}54)$$

$$V_d^* = \frac{2.6}{\sqrt{\tau_*}}\left(1 - \frac{50}{\tau_*}\right) \quad \tau_* \ge 270$$

For rough pipes, this equation is a little more complicated but was formulated for MELCOR by Merril:

$$V_d^* = \frac{1}{(I_s + I_B)} \quad for\ \tau_* \le 10 \quad (2\text{-}55)$$

$$V_d^* = 0.69\sqrt{\frac{2}{\pi}}erfc\left(\frac{1}{\sqrt{2}}\left[1 + \frac{b_+}{s_+}\right]\right) \quad for\ \tau_* > 20$$

Where IS and IB result from integration of the non-dimensional diffusion equation over the buffer layer and sub layer respectively, b+ is the non-dimensional roughness, and s+ is the non-dimensional perpendicular stopping distance. This model was originally implemented into MELCOR 1.8.0 by Merrill [23] for a branch version of the code for safety analysis of the International Thermonuclear Experimental Reactor (ITER)

Victoria Deposition Model

The Victoria model [24] also predicts three regimes for turbulent deposition as was observed for the Wood models. Similar to the Wood model, deposition in the turbulent particle diffusion regime, follows that of Davies, Equation (2-47). Though the approximation in Equation (2-48) is not used for the Victoria model, this does not lead to significant differences in results. This term is then added to a term derived by Sehmel [25] for the particle impaction regime:

$$u_{t,s} = 1.47 * 10^{-16} \left(\frac{\rho_a}{1000}\right)^{1.01} (\frac{2 * 10^4 r_a}{D_H})^{2.1} Re^{3.02} \tilde{v} \quad (2\text{-}56)$$

To obtain the following equation for the non-dimensional deposition velocity:

$$V_d^* = \frac{Sc^{-\frac{2}{3}}}{14.5\left\{\frac{1}{6} 1n\left[\frac{(1+\varphi)2}{1-\varphi+\varphi^2}\right] + \frac{1}{\sqrt{3}} atan\left[\frac{2\varphi-1}{\sqrt{3}}\right] + \frac{\pi}{\sqrt{3}}\right\}} + \frac{}{1.47 * 10^{-16} \left(\frac{\rho_a}{1000}\right)^{1.01} (\frac{2 * 10^4 r_a}{D_H})^{2.1} Re^{3.02}} \quad (2\text{-}57)$$

Note that the correlation reported in Equation (2-56) was based on a least squares curve fit to a restricted data set based on experiments for which surfaces were treated, often with petroleum jelly, to simulate a perfect particle sink, by eliminating or drastically reducing particle bounce. Sehmel recommended use of another correlation, fit over a more general data set, for untreated surfaces:

$$u_{t,s} = 1.0 * 10^{-16} \left(\frac{\rho_a}{1000}\right)^{1.83} (\frac{2 * 10^4 r_a}{D_H})^{2.99} Re^{3.08} \tilde{v} \quad (2\text{-}58)$$

This equation was not used by the Victoria code and not implemented into MELCOR, though would be more representative of nonideal surfaces. This correlation is reported here for completeness.

It should also be pointed out that the Victoria user manual indicates Sehmel's equation is used for supermicron particles and Davies model is used for submicron particles. This is an error in the documentation. Doing so leads to a discontinuity in the deposition velocity. Examination of the Victoria source code indicates that the sum of these two terms is actually used.

A maximum is placed on the non-dimensional deposition velocity so that it does not exceed a value of 0.1. This leads to the constant deposition velocity characteristic of the inertia moderated regime. This is also undocumented in the VICTORIA manual.

2.4.2.2.1.2 Turbulent impaction in pipe bends

INL Bend Model Merrill [23]

To calculate the inertial deposition of aerosols in pipe bends, we start with the centrifugal force acting on the particle as the fluid turns a pipe bend. This force is given by

$$F_C = \frac{\pi}{6}(\rho_p - \rho_f) d_p^3 \frac{u_f^2}{r_b} \approx m_p \frac{u_f^2}{r_b} \quad (2\text{-}59)$$

where

d_p = particle diameter (m)
u_f = fluid velocity (m/s)
r_b = bend radius of pipe (m)
ρ_p = particle density (kg/m^3)
ρ_f = fluid density (kg/m^3)
m_p = particle mass (kg)
Θ_b = bend turning angle (radians)
S = the particle radial drift (m)
B = the particle mobility

The terminal velocity in the radial direction that a particle obtains as a result of this force is given by

$$u_{p\div} = BF_c \tag{2-60}$$

where "B" is the particle mobility defined as

$$B = \frac{1}{(3\pi\mu_g d_p)} \tag{2-61}$$

Where μ_g, is the carrier gas viscosity. The time that it takes for a particle to travel around a bend is given by

$$t_b = \frac{r_b \Theta_b}{u_f} \tag{2-62}$$

where Θ_b is the pipe turning angle in radians. Consequently, the radial distance a particle drifts in this turn is the product of bend travel time and the particle radial velocity, which becomes

$$S = \Theta_b B m_p u_f \tag{2-63}$$

when Equation (2-63) is used to express the centrifugal force. By assuming a well mixed particle concentration in the pipe (c_o), the fraction of particles that collide with the wall in the bend is approximately the radial drift distance divided by the pipe diameter (i.e. s/D). The particle flux (#/m^2-s) for inertial deposition based on this collided fraction, when averaged over the pipe surface area, can be expressed as

$$\Gamma_i = \frac{s}{D} \frac{c_o u_f A_c}{A_s} \quad (2\text{-}64)$$

Where

D = pipe diameter (m)

A_C = pipe cross-sectional area (m^2)

A_s = pipe surface area (m^2)

The deposition velocity associated with this particle flux is as follows

$$V_i = \frac{r_i}{c_o} \quad (2\text{-}65)$$

Pui Bend Model

The model used in VICTORIA for deposition in 90° pipe bends under turbulent conditions (i.e., Re ≥ 2300) is based on the experimental and theoretical work of Pui et al. [26]. Their experiments covered a range of Reynolds numbers from 10^2 to 10^4. They found that an exponential relationship between Stokes number and deposition efficiency correlated well with their data. This relationship is

$$\eta_b = 1 - 10^{-0.963St} \quad (2\text{-}66)$$

Where

η_b = deposition efficiency due to flow irregularity (dimensionless)

And the particle Stokes number is given by:

$$St = \frac{C_c \rho_p d_p^2 U_{ave}}{9 \mu D_h} \quad (2\text{-}67)$$

Deposition efficiency is defined as the fraction of aerosol particles of a specific size that deposit. More specifically for Equation (2-66), the deposition efficiency represents the fraction of aerosol particles that deposit near the pipe bend because of inertial effects induced by curvature of the fluid streamlines. Deposition efficiency is converted to deposition velocity in Victoria by the following definition:

$$u_b = \eta_b \frac{U}{L} \frac{V_B}{A} \quad (2\text{-}68)$$

Where

u_b = deposition velocity for flow through a bend

V_B = volume of bulk gas subregion (m^3),

A = surface area for aerosol deposition (m^2)

McFarland Bend Model

McFarland's model [27] is purely empirical and is based on fitting an equation to data obtained from physical experiments and Lagrangian simulations:

$$\eta_b = 1 - 0.01\, exp\left(\frac{4.61 + a\theta St}{1 + b\theta St + c\theta St^2 + d\theta^2 St}\right) \quad (2\text{-}69)$$

where

$$a = -0.9526 - 0.0568\delta \quad (2\text{-}70)$$

$$b = \frac{-0.297 - 0.0174\delta}{1 - 0.07\delta + 0.0171\delta^2} \quad (2\text{-}71)$$

$$c = -0.306 + \frac{1.895}{\sqrt{\delta}} - \frac{2.0}{\delta} \quad (2\text{-}72)$$

$$d = \frac{0.131 - 0.0132\delta + 0.000383\delta^2}{1 - 0.129\delta + 0.0136\delta^2} \quad (2\text{-}73)$$

$$\delta = \frac{2R_{bend}}{h} \quad (2\text{-}74)$$

where

R_{bend} = radius of the bend in the flow path (m)

Calculation of aerosol removal

MELCOR calculates these five velocities, representing deposition by gravity, diffusion, thermophoresis, diffusiophoresis, and turbulent deposition for each surface. The sum gives the aerosol removal rate term $\overline{\Re}_{\ell,k}$ (kg/m^3·s) in Equation (2-36) in the form

$$\overline{\Re}_{\ell,k} = \sum_{j=1}^{N_{str}} K_{j,\ell} Q_{\ell,k} \quad (2\text{-}75)$$

where

N_{str} = total number of heat structure surfaces and/or pool surfaces for aerosol deposition in the control volume

$K_{j,\ell}$ = deposition rate for the heat structure *j* for aerosol section ℓ (s^{-1})

$Q_{\ell,k}$ = aerosol density for section ℓ of component *k* (kg/m^3)

$K_{j,\ell}$ in Equation (2-75) is defined as

$$K_{j,\ell} = \frac{A_j}{V}\left(v_{grav} + v_{diff} + v_{therm} + v_{diffusio} + v_{turbulent}\right) \quad (2\text{-}76)$$

where

A_j = area of heat structure surface *j* (m^2)

V = control volume atmosphere volume (m^3)

The total component mass that deposits on all surfaces from each section is calculated by MAEROS. The fraction $Fr_{j,\ell}$ of the mass in each section that deposits on surface *j* in the control volume is given by the simple expression

$$Fr_{j,\ell} = \frac{A_j K_{j,\ell}}{\sum_{j}^{N_{sur}} A_j K_{j,\ell}} \quad (2\text{-}77)$$

For fallout aerosols the procedure is similar except that the areas are summed for the floor heat structures, pool, and flowthrough areas; no kernels are involved since any kernel would be common to all surfaces involved. The total fallout mass calculated by agglomeration in MAEROS is then distributed over the floor heat structures, pools, and passes through flowthrough areas proportional to the area of each as follows:

$$Fr_i = A_i \Big/ \sum_{i}^{N_{sur}} A_i \quad (2\text{-}78)$$

where

N_{sur} = total number of surfaces and flowthrough areas.

If part or all of a water film drains from a surface of a heat structure to the pool in the associated control volume, any fission products deposited on that surface or in the water film are normally relocated with the water, in proportion to the fraction of the film that is drained. However, the user may change this for any class by resetting the corresponding value in sensitivity coefficient array 7136 to the fraction of the class assumed to relocate with the film (which includes dissolved and suspended aerosols).

When a phase (pool or atmosphere) in a control volume ceases to exist, the aerosols it contains must be relocated. If the pool in a volume completely evaporates, any aerosols in the pool are distributed between the floor heat structures and the flowthrough areas according to Equation (2-78). If the atmosphere in a control volume that is almost completely filled with water completely condenses, all the suspended aerosol mass is added to the aerosol mass in the pool because it is assumed that the pool then completely fills the control volume.

2.4.2.3 Numerical Implementation

In stand-alone MAEROS, the full aerosol dynamics equations are integrated using a conventional Runge-Kutta integration routine [28]. Because the integration is stopped and restarted only at times when an edit is desired, this approach is both accurate and efficient. However, in MELCOR the integration must be stopped at the end of each system timestep and restarted at the beginning of the next to account for the continuous coupling with other MELCOR models, most of which must be exercised outside the MAEROS framework. These include aerosol release from fuel in the Core package, aerosol generation during core-concrete interactions by the MELCOR implementation of VANESA, fog condensation or evaporation calculated by CVH package thermodynamics, simultaneous condensation or evaporation of fission product vapors on heat structure and aerosol particle surfaces, and advection of aerosols between control volumes as controlled by CVH flow rates. Because of this, the Runge-Kutta solver can be very inefficient (the startup costs become excessive) and, for very short steps, there is little or no increase in accuracy over an explicit (forward Euler) integration.

Therefore, in MELCOR appropriate rates of change are evaluated at the beginning of each system timestep and, if an explicit step produces only small changes in the sectional densities, the distribution is updated using this explicit Euler step. Otherwise, the Runge-Kutta solver is used to advance the equations. The criteria for “small change in the sectional densities” and the error tolerances for the Runge-Kutta solution are controlled by the sensitivity coefficients in array 7000 (see Appendix A). If the Runge-Kutta solver does not converge within the requested tolerances, the RN package reduces the timestep to one-half the current value and write a message to the output and diagnostic files informing the user.

Whether the new aerosol distribution is calculated by an explicit step or by the Runge-Kutta solver, a check is performed to ensure that component masses are conserved within a suitable tolerance (given by a sensitivity coefficient in array 7000; see Appendix A). If this check fails, the RN package reduces the timestep to one-half the current value and write a message to the output and diagnostic file informing the user.

The calculation of the MAEROS coefficients is somewhat costly; a full calculation for 20 sections requires about 10 s processing time on a CRAY 1S computer. Therefore, the coefficients are calculated on the first call to the aerosol model for use throughout the entire problem. Input records describing these coefficients are written to a file

automatically and may be read in from this file on a subsequent restart if called for on the RN1_ACOEF record, but this practice is not recommended because of the possibility of user file handling errors. Sensitivity coefficient array 7001 contains error tolerances for numerical integration of the MAEROS coefficients.

Using a constant set of coefficients imposes some modeling constraints however, because various parameters embedded in the coefficients, such as material properties for the CVH atmosphere, are also effectively held fixed despite the fact that they should vary with changing conditions during the problem. Several of the terms in Equation (2-36) also contain driving forces. The coefficients of these forces are calculated and stored.

The following constraints pertain to the current coefficient set:

a	The aerosol material density is assumed to be the same for all components (specified by the user on Input Record RN1_ASP).
b	The aerosol shape, as modeled by the dynamic and agglomeration shape factors (specified by the user on Input Record RN1_MS00), is independent of aerosol composition.
c	The medium in which the aerosol processes are assumed to occur has fixed properties, taken as those for air.
d	The degree of turbulent agglomeration is fixed throughout the problem, specified by the user on Input Record RN1_MS00.
e	Other parameters that control deposition rates do not depend on particle composition. For example, the ratio of the thermal conductivity of air to that of the aerosol material is fixed.

The pressure and temperature of the atmosphere are embedded in these coefficients and are fixed for a single set of coefficients. However, the aerosol module actually calculates four sets of coefficients at points given by combinations of two temperatures (T_{min} and T_{max}) and two pressures (P_{min} and P_{max}), all of which may be specified by the user. The effects of changing thermal-hydraulic conditions during the problem are approximated by interpolating between these sets of coefficients. The T_{min}, T_{max}, P_{min}, and P_{max} parameters are chosen to bound the temperatures and pressures expected in the calculation, and are specified on user Input Record RN1_PT.

The interpolated sectional coefficients CF_i for agglomeration or deposition mechanism i are given by

$$CF_i = (1-F_T)\left[(1-F_p)AC_{11,i} + F_p AC_{12,i}\right] + F_T\left[(1-F_p)AC_{21,i} + F_p AC_{22,i}\right] \quad (2\text{-}79)$$

where

$AC_{11,i}$ = the aerosol coefficient for mechanism i for the lower atmospheric temperature (T_{min}) and pressure (P_{min})

$AC_{12,i}$ = the aerosol coefficient for mechanism i for the lower atmospheric temperature (T_{min}) and higher pressure (P_{max})

$AC_{21,i}$ = the aerosol coefficient for mechanism i for the higher atmospheric temperature (T_{max}) and lower pressure (P_{min})

$AC_{22,i}$ = the aerosol coefficient for mechanism i for the higher atmospheric temperature (T_{max}) and pressure (P_{max})

and F_T and F_p in Equation (2-80) are defined as

$$F_T = \left(\frac{T_{gas} - T_{\min}}{T_{\max} - T_{\min}}\right) \quad (2\text{-}80)$$

and

$$F_p = \left(\frac{P_{gas} - P_{\min}}{P_{\max} - P_{\min}}\right) \quad (2\text{-}81)$$

where

T_{gas} = cell temperature (K), and

P_{gas} = cell pressure (Pa).

At the expense of larger sets of coefficients, some of the constraints above could be removed by interpolating to accommodate other changing parameters or by separating the coefficients so that a relevant parameter is not embedded, but this is not currently allowed through user input.

2.4.3 Sources

In stand-alone MAEROS, sources of aerosols are included in the differential equation solution at a constant source rate over that timestep. In MELCOR, however, only user-defined sources are treated in this way; sources generated by models in other packages

are currently added as a single increment because of the explicit coupling of these packages. Since masses that are added to the aerosol scheme could be from the previous timestep or the present timestep, depending on the calling sequence of the various packages, all masses to be added from other models are lumped together and added to the aerosol size distribution at the start of the timestep.

Sources of aerosols are calculated in-vessel by the fuel-cladding gap release model and the CORSOR release models, as described in Section 2.3.1. Aerosols generated by these models are put into the smallest aerosol section, consistent with the production of small particles by gas-to-particle conversion. Sources of aerosols are also calculated ex-vessel by the VANESA model, as described in Section 2.3.2. The size distribution for these aerosols is assumed to be log-normal, with median diameter and standard deviation given by VANESA.

A number of time-dependent aerosol sources (specified on record RN1_DIM) can also be specified for a control volume by the user (see the RN1_AS input record series). The aerosols can be put in either the control volume pool or atmosphere, with the time rate of the source specified by a tabular function. The mass added is determined by multiplying the mass addition rate (an input constant times the value of the tabular function at the midpoint of the current timestep) by the timestep, or

$$M_{added} = \left[\frac{dM}{dt}\right] \Delta t = [C \times TF(t + \Delta t/2)] \Delta t \qquad (2\text{-}82)$$

where *C* is the mass addition constant *XM* on the RN1_AS input records, *TF* is the tabular function value, and *t* and Δt are the time and timestep, respectively. The size distribution of the source can be uniform, log-normal with respect to log diameter, or user specified, and is constant with time.

2.4.4 Resuspension

When activated, the resuspension model determines whether deposited aerosols are released from a given heat structure surface. For wet surfaces, the resuspension model is automatically turned off, meaning that there is no resuspension from these surfaces. Wet surfaces include pools and surfaces with film present. For dry surfaces, the particles remain attached to a heat structure surface until the gas flow past the surface is sufficient to aerosolize (i.e. resuspend) the deposit. Then all particles larger than a critical diameter are resuspended.

The user has three options for specifying the critical diameter as either, (1) in the input file, (2) have MELCOR use a default model, or (3) use a control function to specify the critical diameter. Regardless of how the critical diameter is determined, MELCOR stores the cumulative particle size distribution of deposited aerosol by section and resuspends all particles in a section for which the lower section boundary particle diameter is above

the critical diameter. Thus, the resuspended aerosol does not form agglomerates, but is the same particle size distribution of the aerosol that deposited on the surface.

Details of the default resuspension model in MELCOR and comparison to experimental data are fully discussed in SAND2015-6119 [29]. The default MELCOR resuspension model computes the critical diameter for resuspension inside a pipe as

$$D_{crit} = \frac{4 \times 10^{-5}}{\pi \tau_{wall}} \tag{2-83}$$

where τ_w is the wall shear stress, which must be given in units of N/m^2, and the critical diameter is in units of meters.

The wall shear stress τ_w can be expressed as

$$\tau_w = \frac{1}{2} f \rho U^2 \tag{2-84}$$

where

f	=	friction factor,
ρ	=	gas density (kg/m^3), and
U	=	gas velocity along the surface (m/s).

In the above equation, the friction factor is calculated using the Blasius formula [31]

$$f = \frac{0.0791}{Re^{0.25}} \tag{2-85}$$

where Re is the flow Reynold's number

$$Re = \frac{\rho D U}{\mu} \tag{2-86}$$

D is the hydraulic diameter (m), and μ is the gas viscosity (Pa•S). The velocity used in the calculation is the gas velocity flowing through the control volume.

2.5 Condensation/Evaporation

Fission products and water can condense onto or evaporate from aerosols, heat structure surfaces, and water pools. Aerosol water is identified with "fog" in the CVH package. The change in fog mass is determined by thermodynamics calculated within the CVH package and is distributed over aerosol sections by the RN package as described below in

Section 2.5.1. Water condensation and evaporation for heat structure and water pool surfaces are treated solely in the HS and CVH packages, respectively. The calculation of fission product vapor condensation and evaporation in the RN package is described in Section 2.5.2.

2.5.1 Water

The stand-alone version of MAEROS includes terms, given in Equation(2-36), for particle growth resulting from condensation of water onto (and shrinkage from evaporation of water from) aerosols. In MELCOR, these terms are not included with the MAEROS numerical solution for agglomeration and deposition. The reason is that inclusion of these terms makes the MAEROS equations “stiff” and therefore computationally difficult to solve, because the characteristic time for mass transfer is small compared to other characteristic times in the problem.

There are two approaches available in MELCOR to deal with condensation and evaporation of aerosol water. The original model, which neglects hygroscopic, surface tension, and molecular free path effects, is described in this section. The user has the option to specify (as part of RN package input) the use of a more detailed model that includes these effects, as described in Section 2.10. The original model is used by default.

In addition to neglect of hygroscopic and surface tension effects, the original MELCOR model assumes that both the temperature difference between gas and aerosols and the characteristic time for mass transfer to and from aerosols may also be neglected. Under these assumptions, the atmosphere can never become significantly supersaturated, and can be significantly subsaturated only if there is no water available to evaporate from the aerosols. In short, the system of atmosphere plus aerosol water must be in thermodynamic equilibrium.

This makes the aerosol assumptions consistent with the equation of state as described in the Control Volume Thermodynamics (CVT) Package Reference Manual and avoids the need to estimate the disequilibrium between liquid and vapor within a basically equilibrium formulation of thermodynamics or to reconcile calculations including rate effects in the RN package with calculations based on equilibrium thermodynamics within the CVT package. It also allows the water on aerosols to be identified with “fog” in the CVT package.

This reduces the task of the RN package to one of distributing the total change in fog mass, as calculated by equilibrium thermodynamics in the CVT package, among the aerosol sections. In general, this is done with changes in sectional water masses proportional to the appropriate relative rates, which are all proportional to the same super- or sub-saturation driving force, and the actual driving force need not be calculated. However, in a few cases (e.g., a sudden decompression in a volume with little or no initial aerosol content) the condensation rate necessary to maintain equilibrium may exceed

that possible on existing aerosols. In such cases, a very rough estimate of the limiting condensation rate is made (as described below), and the excess water is assumed to form new aerosols in the smallest aerosol section by spontaneous nucleation.

The MAEROS equations do not account for the distribution of *composition* of particles within a single section. This major simplification of the general equations resulted from approximating all material densities as equal, rendering the agglomeration and deposition coefficients independent of composition. Thus, the evolution of particle composition and size distribution is independent of composition for these two processes. The composition distribution can be important in cases of water condensation or evaporation, where a change in water mass can carry a wet aerosol particle from one size section to another. A full treatment would require both the tracking of a more general size-and-composition distribution, and the inclusion of models to account for the differing rates of condensation of water on particles of differing composition.

In MELCOR, two assumptions are permitted for condensation/evaporation of water. The first is equivalent to assuming that all particles within a section have the same composition and allows changes in water mass to freely carry particles of other materials from one size section to another. If water condenses on and then evaporates from a dry aerosol, the final distribution calculated using this treatment does not match the initial one—even in the absence of agglomeration or deposition—and may contain particles smaller than any initially present. The alternative assumption is that condensation and evaporation of water are ineffective in moving other materials from section to section. This is sometimes described as “allowing water to condense only onto water.” The errors in this treatment are different from—but no less serious than—those in the first treatment. The two options, while not necessarily representing limiting cases, allow a user to investigate the potential importance of the effects modeled.

Condensation within a section is evaluated explicitly. The total change in water mass is taken as proportional to the sum over sections of the ${}^1\overline{G}$ term in Equation (2-36) for water, using start-of-step aerosol masses (the ${}^2\overline{G}$ growth terms cancel when summed over all sections, while the ${}^3\overline{G}$ terms are infinitesimal contributors in the differential limit and are ignored). Since the new total water mass on aerosols is equal to the new fog mass calculated by the CVH package, the normalization constant, A, can therefore be determined from the equation

$$\Delta m_w = A\ \Delta t\ \sum_{\ell} {}^1\overline{G}_{\ell,w} Q_\ell \qquad (2\text{-}87)$$

where Δm_w is the total mass of water that must be condensed, as required by the CVH package.

The rate of growth of an individual aerosol particle as a result of condensation is given by the Mason equation [9] as

$$\frac{dm}{dt} = \rho_1 4\,\pi\, r\left(r\,\frac{dr}{dt}\right) = \frac{4\,\pi\, r\,(S-1)}{a+b} \tag{2-88}$$

where *a* and *b* are heat flux and vapor diffusion terms, respectively,

$$a = \frac{M_w\, i_{fg}^2}{k_v\, R\; T^2} \tag{2-89}$$

$$b = \frac{R\;T}{P_{sat}\; D\; M_w} \tag{2-90}$$

and where

m	=	particle mass
ρ_1	=	particle density
S	=	ambient saturation ratio
r	=	mean aerosol particle radius of section *i*
M_w	=	molecular weight of water
i_{fg}	=	latent heat of water
k_v	=	vapor thermal conductivity
R	=	gas constant
T	=	ambient temperature
P_{sat}	=	saturation pressure at T
D	=	diffusivity of water vapor in air

Equations (2-87) through (2-90) can be combined to relate the normalization constant *A* and the ${}^1\overline{G}_{\ell,w}$ term to the Mason equation:

$$\frac{dm_w}{dt} = \sum_\ell N_\ell \left\langle \frac{dm}{dt} \right\rangle = \sum_\ell Q_\ell \left\langle \frac{1}{m}\,\frac{dm}{dt} \right\rangle = \sum_\ell Q_\ell \left\langle \frac{4\pi r}{m} \right\rangle \frac{(S-1)}{a+b} = \sum_\ell Q_\ell\, {}^1\overline{G}_{\ell,w}\, A \tag{2-91}$$

where N_ℓ is the number of particles in section ℓ and the angle brackets denote an appropriate sectional average. Therefore, the MAEROS coefficient ${}^1\overline{G}_{\ell,w}$ can be

evaluated as an appropriate sectional average of $4\pi\ r/m$ and A can be taken as the term $(S - 1)/(a + b)$, which is independent of size. Equations (2-88) and (2-91) are consistent if an effective value of the saturation ratio S, which varies through the timestep, is chosen appropriately. A limiting rate on condensation can be estimated from Equation (2-91), using an upper bound on the saturation ratio based on the assumption that all vapor destined to condense exists in the vapor phase at the start of the step. That is,

$$S_{\max} = 1 + \Delta m_w \ / \ m_{v,sat} \tag{2-92}$$

where Δm_w is again the mass of water to be condensed and $m_{v,sat}$ is the mass of water vapor at saturation in the atmosphere. If the required condensation exceeds this limiting rate, A in Equation (2-87) is set to the limiting value, $(S_{max} - 1)/(a + b)$, and the excess water is simply put into the smallest aerosol section, consistent with the assumption that excess water that cannot condense on existing aerosol, structures, or pools condenses by homogeneous nucleation, forming small fog droplets.

Transfer from section to section by growth of aerosols is evaluated implicitly; that is, the ${}^2\overline{G}_{1,w}$ terms are evaluated using end-of-step masses. For condensation, aerosols can only grow, and by definition there can be no growth into the smallest section. This allows the new masses to be evaluated in a single pass from the smallest section to the largest by forward substitution,

$$Q_{1,k}^n = \frac{Q_{1,k}^{o+}}{1 + A\,\Delta t\ {}^2\overline{G}_{1,w}} \tag{2-93}$$

$$Q_{\ell,k}^n = \frac{Q_{\ell,k}^{o+} + A\ \Delta t\ {}^2\overline{G}_{\ell-1,w} Q_{\ell-1,k}^n}{1 + A\,\Delta t\ {}^2\overline{G}_{\ell,w}} \tag{2-94}$$

where $Q_{\ell,k}^{o+}$ is the start-of-step mass for all classes but water, in which case it includes the explicitly calculated condensation. Note that, from a strictly numerical standpoint, no negative masses can be predicted by this equation if there were none at the start of the step.

The treatment of evaporation is very similar to that for condensation. Evaporation within a section is calculated explicitly, and the total is normalized to the change in water mass required by the CVH package, but no rate limit is considered. If one or more explicitly calculated water masses would be negative, they are set to zero and the remaining (positive) masses renormalized to the correct total.

As in condensation, the section-to-section transfers are evaluated implicitly in a single pass, this time from the largest to the smallest. Experience has shown that one further modification is necessary. If the limit and renormalize procedure just mentioned is used,

the value of A used for section-to-section transfers out of each section must be made to agree with the effective value of A used for evaporation from that section. This is easily done by defining

$$A_s \Delta t = \frac{Q^{o+}_{\ell,w} - Q^{o}_{\ell,w}}{{}^1G_{\ell,w} Q_\ell} \tag{2-95}$$

where A_s is simply the normalization constant A in cases where no dryout occurred.

2.5.2 Fission Product Vapors

The condensation and evaporation of fission product vapors to and from heat structures, pool surfaces, and aerosols is evaluated by the same equations as in the TRAP-MELT2 code [8]. The fission product vapor masses in the control volume atmosphere and condensed on the aerosol and heat structure surfaces are determined by rate equations based on the surface areas, mass transfer coefficients, atmosphere concentration, and the saturation concentrations corresponding to the temperatures of the surfaces:

$$\frac{dM_a}{dt} + \sum_i \frac{dM_i}{dt} = 0 \tag{2-96}$$

$$\frac{dM_i}{dt} = A_i k_i \left(C_a - C_i^s \right) \tag{2-97}$$

where

C_a = M_a / V = concentration of vapor in atmosphere

C_i^s = saturation concentration of vapor in atmosphere at temperature of surface i

M_i = condensed mass of vapor on surface i

V = volume of atmosphere

A_i = area of surface i

k_i = mass transfer coefficient for surface i

Subscript i denotes any heat structure surface, pool surface, or aerosol section.

These differential equations can be solved as in TRAP-MELT2 to yield the following algebraic equations:

$$C_a = M_a / V = \frac{\beta}{\alpha} - \left(\frac{\beta}{\alpha} - C_{a0} \right) e^{-\alpha \Delta t} \tag{2-98}$$

$$M_i = M_{i0} + A_i k_i \left(\frac{\beta}{\alpha} - C_i^s \right) \Delta t - A_i k_i \left(\frac{\beta}{\alpha} - C_{a0} \right) \left(\frac{1 - e^{-\alpha \Delta t}}{\alpha} \right) \tag{2-99}$$

where

$$\alpha = \sum_i A_i k_i / V \tag{2-100}$$

$$\frac{\beta}{\alpha} = \frac{\sum_i A_i k_i C_i^s}{\sum_i A_i k_i} \tag{2-101}$$

Subscript 0 denotes the value at start of the timestep, Δt .

Total sectional areas A_p for aerosols are calculated from the average particle in each section, as derived in Appendix C:

$$A_p = 12\pi \left(\frac{3}{4\pi\rho} \right)^{2/3} \frac{M}{ln(m_2/m_1)} \left({m_1}^{-1/3} - {m_2}^{-1/3} \right) \tag{2-102}$$

The mass transfer coefficient k_p for aerosols is based on zero slip flow, or Sherwood number = 2.0.

All HS package heat structures are automatically included for condensation and evaporation of fission product vapors unless made inactive through user input on RN1_DS records. The area of the heat structure in the atmosphere A_w is used to define the net area for fission product vapor interactions. This area is the total heat structure area times the fraction of the heat structure in the atmosphere as determined by the HS package.

Although fission products may condense on pool surfaces, evaporation of fission products residing in control volume pools is not permitted. The fission product vapor location within a phase in a control volume (pool or atmosphere) may change when one phase is no longer present. Any vapor mass associated with a disappearing phase is added to the remaining phase in that control volume.

The mass transfer coefficient for condensation of fission product vapors onto heat structure surfaces, k_w, is calculated based on the mass transfer coefficient, k_{HS}, for water condensation onto a heat structure surface calculated by the HS package, which uses the steam-air diffusivity, $D_{st,a}$:

$$k_w = k_{HS}\, D_{k,g} / D_{st,a} \tag{2-103}$$

The vapor diffusivity for the fission product vapors in the bulk gas, $D_{k,g}$, is calculated from the following equation as presented in Welty, Wicks, and Wilson [30]:

$$D_{k,g} = \frac{1 - y_k}{\sum_n \left(y_n / D_{k,n} \right)} \tag{2-104}$$

where

y_k = mole fraction of trace vapor k

y_n = mole fraction of bulk gas n

$D_{k,n}$ = binary diffusivity of vapor k in gas n

The binary diffusivities are evaluated from the following expression from Bird, Stewart, and Lightfoot [31]:

$$D_{A,B} = 0.0018583 \; \frac{\left[T^3 \left(M_A^{-1} + M_B^{-1} \right) \right]^{1/2}}{P \sigma_{AB}^2 \Omega_{D,AB}} \tag{2-105}$$

with

$D_{A,B}$ = binary diffusivity in cm^2/s

T = temperature in K

P = pressure in atmospheres

M_i = molecular weight in kg/kg-mole

σ_{AB} = collision diameter in Angstroms = $0.5 \left(\sigma_A + \sigma_B \right)$

$\Omega_{D,AB}$ = collision integral = function of kT/ε (see Table B-2 of Reference [31])

The actual calculation of $D_{A,B}$ is performed by a model in the Material Properties (MP) package, using data for the collision integral contained in the MP database. Values for the Lennard-Jones potential parameters σ and ε/k for the bulk gases are obtained from the MP database, while values for some of the fission product vapors, obtained from Reference [31], are stored in RN sensitivity coefficient array C7111 (see Appendix A). Actual values are used for Xe and I_2; other classes are defaulted to values for air due to a lack of information. (The values for the bulk gases are the same ones used for calculation of viscosity in the absence of tabular data; they may be changed through MP input if desired.)

In addition to being used to determine the amount of each material class present as aerosol and as fission product vapor, the vapor pressure is used in the model for condensation and evaporation to determine the saturation concentrations, C_i^s, calculated from the perfect gas law,

$$C_i^s = \frac{P(T_i)M_w}{R\,T_i} \tag{2-106}$$

The expression for the vapor pressure is

$$\log_{10}(P) = -A/T + B + C\ \log_{10}(T) \tag{2-107}$$

with P and T in units of mm of Hg and K, respectively. The coefficients A, B, and C for each class are stored in sensitivity coefficient array 7110 for different temperature ranges (see Appendix A). Classes for which there are no data are assumed to have a default vapor pressure curve characteristic of a nonvolatile ceramic (zero vapor pressure below 3000 K and the vapor pressure of UO_2 above 3000 K); non-default vapor pressure coefficients are defined for classes 2 (Cs), 3 (Ba), 4 (I2), 5 (Te), 6 (Ru), 7 (Mo), 8 (Ce), 9 (La), 10 (UO_2), 11 (Cd), 12 (Ag), 13 (B_2O_3), 16 (normally CsI), and 17 (normally CsM), and class 1 (Xe) is always a vapor. (See the RN Package Users' Guide for details on defining temperature ranges and forcing classes to always be an aerosol or always a vapor.)

For temperatures above a maximum temperature value, T_{max}, the correlation is extrapolated. However, direct use of the correlation outside its range of applicability can return a pressure that decreases with increasing temperature, because C is negative and $C\ \log_{10}(T)$ can dominate $-A/T$. Therefore, the extrapolation uses

$$\log_{10}(P) = -A'/T + B' \tag{2-108}$$

The coefficients A' and B' are derived from the last range coefficient values A, B, and C by demanding that P and dP/dT be continuous at the matching temperature T_{max}. This requires

$$A' = A + C\ \log_{10}(e)T_{\max} \tag{2-109}$$

$$B' = B + C\ \left[\log_{10}(e) + \log_{10}(T_{\max})\right] \tag{2-110}$$

2.6 Decay Heat Distribution

All decay heat released by radionuclides in a control volume pool is assumed to be absorbed by that pool. None of this decay heat is added directly to any heat structure surface or to the atmosphere of the control volume.

The decay heat released by radionuclides in the control volume atmosphere and from those deposited on the various heat structure surfaces can be apportioned according to user specifications among the volume atmosphere, the surfaces of heat structures in that volume, and the pool surface (if a pool is present). Fractions may also be specified as going to the atmosphere and surfaces of other volumes to simulate decay radiation transmitted through flow paths. Defaults are provided, as discussed below.

Approximately one half of decay heat is generated as gamma radiation and one half as beta radiation. Because typical gaseous atmospheres are nearly transparent to typical gammas and fairly opaque to typical betas, deposition of decay heat in a volume atmosphere results primarily from absorption of beta radiation. (The split and the characteristic energies are not explicitly modeled by MELCOR.) These observations and solid angle considerations led to the default splits suggested by Reference [3]:

Decay Heat from Radionuclides in the Atmosphere	
Atmosphere of current CV	50%
Surfaces of current CV	50%
Atmosphere of other CVs	0%
Surfaces of other CVs	0%
Decay Heat from Radionuclides on Heat Structure Surfaces	
Current Heat Structure	50%
Atmosphere of current CV	25%
Other surfaces of current CV	25%
Atmosphere of other CVs	0%
Surfaces of other CVs	0%

All fractions are independent of the RN class. Those for airborne radionuclides can be changed on a volume-by-volume basis using the RN1_DHV and RN1_DHVS input record series. Those for radionuclides on surfaces can be modified similarly, on a surface-by-surface basis, using the RN1_DHVS input record series.

Decay heat from airborne or deposited radionuclides that is absorbed by surfaces in the same control volume is allocated among the surfaces in proportion to their areas. (Note that for deposited radionuclides the bearing surface is not included.) The areas considered are the portions of heat structure surfaces exposed to the atmosphere, and the surface of the pool (if a pool is present). If there are no such surfaces, the fraction of decay heat allocated to the surfaces of a control volume is deposited instead in the atmosphere of that control volume.

The fractions specified as going to the local control volume atmosphere (by default or user input) are interpreted as the values appropriate for complete absorption of beta radiation. They must be reduced for small volumes or low densities, where the thickness of the atmosphere is insufficient to permit complete absorption of beta rays. This reduction is by a factor

$$\min\left(\rho_A\, D_{CV}/R_\beta, 1.0\right) \qquad (2\text{-}111)$$

where ρ_A is the atmosphere density, D_{CV} is the characteristic dimension for absorption in the control volume, and R_β is the range of a typical beta particle (given in sensitivity coefficient array 7002, with a default value of 1.2 kg/m^2; see Appendix A). D_{CV} has a default value given by the minimum of the cube root of the volume and the square root of the flow area from the CVH database (to be reasonable for both tanks and pipes). It can be modified using the RN1_DHL input record series.

Any reduction in deposition to the local atmosphere is compensated by proportionate increase in energy distributed to other surfaces in the volume and to the atmosphere and surfaces of other control volumes. (The calculation is bypassed if the sum of these other split coefficients is zero.)

2.7 ESF Models

Models are currently available for the removal of radionuclides by pool scrubbing, filter trapping, and spray scrubbing. These models are described in the following subsections. The normal RN deposition and condensation models described in Sections 2.4 and 2.5 are applied to heat structures used to model ice condensers; see the HS Package Reference Manual for a detailed description of methods used to model ice condensers, including a surface area enhancement factor for radionuclide deposition.

2.7.1 Pool Scrubbing

The pool scrubbing model in the RN package is based on the SPARC-90 code [10]. (The thermal-hydraulic aspects of pool scrubbing are modeled in the CVH package.) Aerosols and iodine vapor are removed by pool scrubbing; the model also treats organic iodine vapor (CH_3I) but currently it is not included in the MELCOR RN class structure. Decontamination is calculated for those flow paths activated on the FL_JSW input record (see the FL Package Users' Guide) and for gases evolved from core-concrete interactions in cavities activated on the CAV_U input record (see the CAV Package Users' Guide). By default, the model treats these cases by using the horizontal vent scrubbing option from SPARC-90 along with the flow area provided by the FL package (FLARA on input record FL_GEO) or the flow area calculated by the CAV package. However, the user may override the default venting treatment by providing appropriate input on the RN2_PLS records. For consistency with the CVH package, pool scrubbing is only calculated if the submerged depth of the flow path is greater than the zero-efficiency

bubble rise height given in CVH sensitivity coefficient array 4405. The gases evolved from the core-concrete interactions calculated by VANESA are supplemented by an inferred steam flux generated by boiling at the cavity/pool interface. This flux is evaluated by dividing the cavity/pool interfacial heat flux calculated by CORCON by the latent heat of vaporization for water in the pool.

The decontamination factor (DF) is defined as the ratio of the radionuclide mass entering the pool to that leaving and has a value greater than or equal to unity. However, when the iodine concentration in the pool divided by the equilibrium partition coefficient (discussed in Appendix F) exceeds the concentration of iodine vapor in the gas entering the pool, then iodine vapor scrubbing cannot occur and the corresponding decontamination factor must be equal to unity. (Furthermore, MELCOR is not structured to calculate iodine stripping from the pool under these conditions, so iodine removal from the pool is not considered.) If the iodine concentration in the bubbles is significant (i.e., exceeds a threshold value implemented in sensitivity coefficient array 7159 with a default value of 10^{-6} moles/cm^3), a message is issued once per calculation by the scrubbing routine to inform the user of this condition.

The gas flow through the pool is described in two overlapping regions. In the vent exit region, the injected gas forms large, unstable globules. The initial size of the globule depends on the vent type and the noncondensible gas flow rate. As the globules rise they begin to break up into swarms of smaller bubbles. It is assumed that break-up is complete by the time the globule rises a distance equal to twelve times its initial diameter. In the swarm rise region, bubbles continually coalesce and redisperse during their erratic ascent. On average, however, it is assumed that they can be represented by oblate spheroids of a constant, stable size with the flatness given by a correlation depending on bubble size. The rise velocity of individual bubbles in the swarm relative to the liquid is given by a correlation depending on bubble size, also, and remains constant since the size remains constant. The swarm rise velocity represents the volumetric average velocity on a cross section of the swarm. Bubbles in the center rise faster than swarm periphery bubbles, and the swarm rise velocity increases as the swarm ascends because the volumetric flow rate of the swarm increases as the gas expands under a decreasing static head. In the SPARC-90 model, however, the swarm rise velocity is assumed to remain constant with a value given by the average of the value at the vent exit depth and the value at the pool surface. The bubbles in the swarm multiply (i.e., the number density increases) as the expanding bubbles split to preserve their stable size. The viscous shear of the liquid in relative motion past the bubble causes the bubble surface and interior to move in a top-to-bottom rotation.

It is assumed that the inlet gas comes into thermal equilibrium with the pool almost instantaneously in the vent exit region. When this results in steam condensation in the inlet gas, aerosol particles and iodine vapors are removed in proportion to the reduction in the volumetric flow rate. Particle capture also occurs when the injection velocity is large because inertia forces the particles into the front boundary of the rapidly decelerating

globules. For multihole vents with small orifices, centrifugal, diffusional and gravitational deposition are evaluated during gas injection because they are significant at the large velocities achieved. Details of globule formation and vent exit region scrubbing are given in Appendix D.

Scrubbing in the swarm rise region is evaluated by numerically marching through the region in several discrete spatial steps. At the beginning of each step, the fraction of the inlet gas that is still contained in the initial globule is determined. The remainder is assumed to be contained in bubbles. During each step the thermal hydraulic conditions within the bubbles are updated and used to evaluate the incremental removal of particles and iodine vapors during the step. The particle removal mechanisms modeled in the bubble include centrifugal and diffusional deposition and gravitational sedimentation. These mechanisms generate a flux of particles toward the bubble surface, where they are removed by absorption into the pool. The particle flux may be hindered by a flux of water vapor into the bubble, if evaporation is occurring at the bubble surface. Conversely, condensation onto the particles within the bubble because of supersaturation from bubble expansion enhances particle removal. The vapor removal mechanism is diffusion, which also may be hindered if there is an evaporative flux into the bubble. The removal factor for each particle size and iodine vapors during the step is given by:

$$DF_{SR,i} = \frac{1}{f_{gl} + \left(1 - f_{gl}\right) / DF_{BB,i}} \qquad (2\text{-}112)$$

where

$DF_{BB,i}$ = removal factor inside the bubble

f_{gl} = fraction of inlet gas still in the inital globule

The cumulative removal factors for each particle size and iodine vapors in the swarm rise region are given by the product of the incremental removal factors at each step. Details of transient bubble behavior and particle scrubbing in the bubbles are given in Appendix E. Details of iodine scrubbing in the bubbles are given in Appendix F.

The overall removal factor for each particle size and iodine vapors in the vent exit and swarm rise regions is given by

$$DF_{OV,i} = DF_{EC} \bullet DF_{II,i} \bullet DF_{ER,i} \bullet DF_{SR,i} \qquad (2\text{-}113)$$

where

DF_{EC} = *DF* from steam condensation in the vent exit region

$DF_{II,i}$ = *DF* from inertial impaction (of particles only) in the vent exit region

$DF_{ER,i}$ = *DF* from centrifugal, diffusional and gravitational capture (of particles only) in the vent exit region

$DF_{SR,i}$ = cumulative *DF* in swarm rise region

The overall removal factor for all particle sizes is obtained by dividing the sum of the inlet mass rates over all sizes by the sum of the outlet mass rates (the inlet rates for each size divided by the overall removal factor for that size) over all sizes

$$DF_{OV,part} = \frac{\sum_{i=1}^{NBINS} \dot{m}_{part,i}}{\sum_{i=1}^{NBINS} \left(\frac{\dot{m}_{part,i}}{DF_{OV,i}} \right)} \qquad (2\text{-}114)$$

2.7.2 Filters

The MELCOR RN package contains a simple filter model. When aerosols and vapors are transported through flow paths with the bulk fluid flow of pool and/or atmosphere calculated by the CVH package, some fraction of the transported RN materials may be removed by the action of filters in the flow path. A single filter can remove either aerosols or fission product vapors, but not both. However, a flow path can contain more than one filter. The efficiency of each filter is defined by decontamination factors, specified by user input. By default, a single decontamination factor is applied to all RN classes *except* water, for which the default *DF* is 1.0. Additional user input may be used to modify the *DF* on a class-by-class basis, *including the water class*. The parameters for the filter characteristics are specified on the RN2_FLT input record series.

A maximum loading may be specified for each filter; when this loading is reached, no further RN materials are removed (i.e., the *DF* is set to unity).

The effect of filter loading on the flow resistance of the associated flow path may be modeled through user input. This requires construction of a control function to link the laminar loss coefficient for the flow path (SLAM, input on segment record FL_SEG; see the FL Package Users' Guide) to the filter loading. The filter loading may be obtained from one or more of the RN2-AMFLT or RN2-VMFLT control function arguments described in Section 5 of the RN Package Users' Guide.

The decay heat energy from radionuclides deposited on filters is given to the downstream control volume according to the vapor flow direction.

2.7.3 Sprays

The MELCOR Containment Sprays (SPR) package, which calculates the thermal-hydraulic behavior associated with spray systems, is coupled to the RadioNuclide package for the calculation of aerosol washout and atmosphere decontamination by the sprays.

The SPR Package Reference Manual describes the thermal-hydraulic modeling of the spray systems. To summarize here, the spray droplets are assumed to be spherical and isothermal and to fall through containment at their terminal velocity without a horizontal velocity component. Droplet heatup and cooldown in a steam environment are modeled using a correlation for forced convection heat transfer coefficients. Similarly, evaporation and condensation are modeled using a correlation for mass transfer coefficients. A standard integrator is used to integrate these transfer rates over the fall height of the spray droplet to obtain the final droplet mass and temperature. By comparing the droplet mass and temperature at the bottom of the compartment to the inlet conditions, the heat and mass transfer to a given droplet is computed. Total heat and mass transfer rates are calculated by multiplying the rates for one droplet by the total number of droplets of that size and summing over all droplet sizes.

The SPR-RN interface may produce nonphysical results if the SPR package is required to make multiple passes (numerically) through the same control volume on a given timestep. Therefore, the user is strongly encouraged to avoid this situation by limiting the spray activity to a single drop size in each spray train. The user must also ensure that only one spray train passes through each control volume. These restrictions are necessary only when the SPR and RN packages are used at the same time.

The particulate removal by sprays is a mechanistic treatment of removal processes, closely coupled to the thermal-hydraulic behavior calculated by the spray package. The user is cautioned to use a single drop size and a single spray train per volume because of the method by which the RN removal calculation is “piggybacked” onto the Spray Package thermal-hydraulic calculations. Specifically, the thermal-hydraulic stepwise integration over the spray train height is made first, then the RN removal processes are calculated by a simple trapezoidal integration over the step, using the appropriate end-of-interval values. Because each droplet size is integrated over the full height of fall separately, there exists the possibility of competing radionuclide removal by differing drop sizes and competing removal by different spray trains.

The particulate removal from sprays is modeled as a first-order rate process,

$$\frac{dM_k}{dt} = -\lambda_{k,i} M_k \tag{2-115}$$

where

M_k = mass of class k

$\lambda_{k,i}$ = rate constant for class k, droplet size i

The actual physical removal processes for vapors and aerosols are different and therefore different rate constants, λ, are associated with each process.

Vapor removal by adsorption is calculated using a stagnant film model for the adsorption efficiency. The vapor removal is calculated as an injection spray removal rate; no recirculation of spray liquid is considered. The expression for the rate constant is [32, 33, 34]:

$$\lambda_{k,i} = \frac{F_i E_{k,i} H}{V} \tag{2-116}$$

where

F_i = volumetric flow rate for droplets of size i

$E_{k,i}$ = adsorption efficiency for vapor class k

H = partition coefficient for partition of the vapor between spray water and gas

V = volume of control volume

The vapor absorption efficiency is given by the expression [35]

$$E_{k,i} = 1 - \exp\left[- \frac{6\ k_g\ t_e}{2 r_i \left(H + k_g / k_\ell\right)} \right] \tag{2-117}$$

where k_g, the gas boundary layer mass transfer coefficient, is calculated using the Ranz and Marshall approximation [34] to the Frossling equation [34],

$$k_g = \frac{D_{k,gas}}{2 r_i} \left(2.0 + 0.060\ \mathrm{Re}^{1/2}\, Sc^{1/3}\right) \tag{2-118}$$

and k_ℓ, the liquid boundary layer mass transfer coefficient, is calculated using Griffith's approximation for diffusion in a rigid drop [34],

$$k_\ell = \frac{\pi^2\ D_{k,\mathrm{H_2O}}}{3 r_i} \tag{2-119}$$

In these equations,

r_i = drop radius

t_e = drop exposure time

$D_{k,gas}$ = diffusivity of vapor *k* through bulk gas

D_{k,H_2O} = diffusion constant for vapor *k* in liquid water

Re = Reynolds number, $2\rho_g v_d r_i / \mu_g$

Sc = Schmidt number, $\mu_g / \rho_g D_{k,gas}$

v_d = drop velocity

Under LWR accident conditions, iodine may exist as a vapor over relatively long time periods in containment pressure/temperature conditions. Other materials have low vapor pressures at accident conditions that preclude their extended existence as vapors; that is, they condense to aerosol forms quickly. The RN input record series RN1_IOD01 allows the user to specify a limit on iodine adsorption by spray droplets using a partition coefficient. The partition coefficient for iodine, defined as the equilibrium ratio of the iodine density in the liquid to its density in the gas,

$$H = \rho_{\ell,eq} / \rho_{g,eq} \tag{2-120}$$

is specified by the user for sprays containing different additives, with various recommended values ranging from 500 to 100,000 [36] listed in the RN Package Users' Guide.

Aerosol removal is calculated primarily by inertial impaction and interception; diffusiophoresis and diffusion effects are also included. No droplet interactions are considered. Impaction and interception are the primary removal mechanisms as long as droplet radii are in the 10 – 100 micron size range. From 1 – 10 microns diffusiophoresis becomes an important contributor; diffusion only becomes important for droplets with radii < 0.1 micron. The expression for the rate constant is [36]

$$\lambda_{k,i} = \frac{3F_i h E_{i,j}}{4Vr_i} \tag{2-121}$$

where F_i, V, and r_i are as defined before, h is the fall height of the drops, and $E_{i,j}$ is the efficiency of collection of aerosol particles in size section *j* by drops of size *i*.

For viscous flow around a sphere, the collection efficiency for interception (denoted by subscript *In*) is given by the expression [37]

$$\varepsilon_{In,vis} = (1+I)^2 \left[1 - \frac{3}{2(1+I)} + \frac{1}{2(1+I)^3} \right] \quad (2\text{-}122)$$

where $I = r_p / r_d$ and r_p and r_d are the radii of the particle and the drop, respectively.

For potential flow around a sphere, the collection efficiency for interception is given by the expression [37]

$$\varepsilon_{In,Pot} = (1 + I)^2 - (1 + I)^1 \quad (2\text{-}123)$$

For potential flow around a sphere, the collection efficiency for inertial impaction (denoted by subscript *Im*) is given by the expression [38]

$$\varepsilon_{Im,Pot} = \left[\frac{Stk}{Stk + 0.5} \right]^2 \quad (2\text{-}124)$$

for $Stk \geq 0.2$, is zero for $Stk \leq 0.0834$, and is given by linear interpolation for $0.0834 < Stk < 0.2$. For viscous flow around a sphere, the collection efficiency for inertial impaction is given by the expression [37]

$$\varepsilon_{Im,Vis} = \left[1 + \frac{0.75 \log_e (2\ Stk)}{Stk - 1.214} \right]^{-2} \quad (2\text{-}125)$$

for $Stk > 1.214$, and is zero otherwise. *Stk* is the Stokes number,

$$Stk = \frac{2\, r_p^2 \rho_p (v_d - v_p)}{9 \mu r_d} \quad (2\text{-}126)$$

where v_d and v_p are the terminal settling velocities of the drop and particle, respectively, and μ is the bulk gas viscosity. An interpolation scheme from Reference [38] is used to combine the potential and viscous efficiencies for both interception and inertial impaction:

$$\varepsilon_x = \frac{\varepsilon_{x,Vis} + \varepsilon_{x,Pot}(Re/60)}{1 + (Re/60)} \quad (2\text{-}127)$$

where Re is the drop Reynolds number and subscript *x* is either *In* (interception) or *Im* (inertial impaction).

The collection efficiency due to diffusion is given by the expression [33]

$$\varepsilon_{diff} = 3.02\, \mathrm{Re}^{1/6}\, Pe^{-2/3} + 1.14\, (\mathrm{Re}/Pe)^{1/3} I + 0.57\, \mathrm{Re}^{1/3}\, I^2 \quad (2\text{-}128)$$

where *Pe* is the Peclet number, $2r_d(v_d - v_p)/D$.

The collection efficiency due to diffusiophoresis is given by the expression

$$\varepsilon_{diffusio} = \frac{4}{3}\frac{r_d}{F\ h}\left[\frac{M_s^{1/2}}{X_s\ M_s^{1/2} + X_g M_g^{1/2}}\right]\ \frac{W_s}{cM_s} \qquad (2\text{-}129)$$

where W_s is the mass condensation rate of steam onto drops, M is molecular weight, X is mole fraction, c is the molar concentration of bulk gases, and subscripts s and g refer to steam and noncondensible bulk gases, respectively.

Finally, the collection efficiencies for different processes are combined using the following expression

$$E_{i,j} = 1 - \prod_k \left(1 - \varepsilon_{ijk}\right) \qquad (2\text{-}130)$$

where subscript k refers to the collection process.

2.8 Fission Product Chemistry

Chemistry effects can be simulated in MELCOR through the use of *class reactions* and *class transfers*. The class reaction process uses a first-order reaction equation with forward and reverse paths. The class transfer process, which can change the material class or location of a radionuclide mass, can be used to simulate fast chemical reactions. With these two processes, phenomena including adsorption, chemisorption, and chemical reactions can be simulated.

Note: Only fission product vapors are considered in the chemistry models.

2.8.1 Class Reactions

The reaction process model in MELCOR is a first-order reversible reaction for a class going from state C in the gas-phase to state C_1 on a surface, or

$$\frac{dM_c}{dt} = -\left(\frac{k_m\ A\ /V}{k_m A\ /V + k_f}\right)\left(k_f M_C - k_r M_{C1}\right) \qquad (2\text{-}131)$$

where

k_m = mass transfer rate constant for the process, based on the mass transfer coefficient calculated by the HS Package, (m/s)

k_f = forward reaction rate constant from user input, (s^{-1})

k_r = reverse reaction rate constant from user input, (s^{-1})

A / V = surface-to-volume ratio, where the surface area is that for the reaction and the volume is that of the control volume (m^{-1}).

The mass transfer rate constant is calculated in the same manner as the vapor condensation/evaporation diffusivity given in Section 2.5.

The solution technique is the same as for vapor condensation/evaporation under the assumption that the mass of $C1$ does not change during the timestep. This assumption avoids solving a differential equation and allows the use of the same algebraic solution given in Section 2.5.

Alternatively, if the user specifies the use of a deposition velocity instead of the forward and reverse reaction rate constants,

$$\frac{dM_C}{dt} = -V_d(A/V)C = -\frac{dM_{C1}}{dt} \qquad (2\text{-}132)$$

where V_d is the user input reaction deposition velocity in m/sec.

The reaction only occurs in user-specified control volumes and depends on the availability of the various classes as determined by the user input reaction stoichiometry. The first “from” class in the reaction must be in the vapor phase, while all the other specified classes must be deposited on the surface when the reaction occurs. Surfaces that can undergo reactions include heat structures, the pool surface, and aerosol surfaces as specified by the user. A flag to specify whether the reaction still occurs when a water film is present is also available. At the present time, water mass should not be used in the class reaction model.

In addition to the masses, reaction energy can also be specified for both the forward and reverse directions. The energy is in terms of the mass of the first “from” reacting class. This energy is added to the atmosphere in the case of reaction with aerosols, to the pool for a pool reaction, and to the heat structure if a surface reaction occurs.

2.8.2 Class Transfers

Mass transfers between classes may be accomplished by the transfer mechanisms. The user may change the class and location of aerosols and/or vapors in an arbitrary fashion. Therefore, this feature must be carefully used.

A stoichiometric reaction is specified, and the permitted control volumes and “from” and “to” states are given. The permitted states are aerosols or condensed vapors on a given surface of a heat structure, or aerosols or vapors in either the atmosphere or pool. A flag

to determine if the transfer proceeds with a water film present is also available. Water should not be used in the class transfer model.

The mass transfer rate is given by the user as is the energy transfer information. The masses are changed as follows:

$$M_{from,\,t+\Delta t} = M_{from,\,t} - \frac{dM}{dt}\Delta t \qquad (2\text{-}133)$$

$$M_{to,\,t+\Delta t} = M_{to,\,t} + \frac{dM}{dt}\Delta t \qquad (2\text{-}134)$$

where dM/dt is the user-specified mass transfer rate. Thus, with this option, aerosols of Class A in the pool may be, for example, changed into condensed vapors of Class B on a heat structure. This model is used for fast reactions with the "from" and "to" state generally the same.

2.8.3 Example

As an example of both class reactions and class transfers, consider the adsorption of CsI on a surface with a known deposition velocity that is then transformed immediately to CsOH plus HI when adsorbed water is present. After the transformation, the revaporization of CsOH is delayed until the surface temperature reaches T_1 while the HI revaporization is simply mass transfer limited. In this case, CsI, CsOH, and HI are separate material classes, and the reaction diagram can be written as

$$\begin{aligned} CsI(g) &\rightarrow CsI(ad) \\ &\quad\downarrow H2O \\ CsOH(g) &\leftrightarrow CsOH(ad) \\ &\quad + \text{(depends on } T_1\text{)} \\ HI(g) &\leftrightarrow HI(s) \end{aligned}$$

where (g), (ad), and (s) are gaseous, adsorbed, and solid states, respectively.

This reaction can be simulated by the RN package by the following sequential class reactions and transfers:

CsI(g) → CsI(ad)	rate constant for adsorption is supplied through input
CsI(ad) → CsOH(ad) + HI(s)	instantaneous and complete transfer between classes when water is present. **Note that the water mass is not included in the model; water mass is not explicitly conserved.**
CsOH(g) → CsOH(ad)	rate constant for adsorption supplied or condensation limited
CsOH(ad) → CsOH(g)	reaction with zero rate constant below T_1
	positive value or instantaneous above T_1
HI(s) ↔ HI(g)	controlled by condensation/evaporation

2.9 Chemisorption on Surfaces

The chemisorption model is implemented as a set of chemisorption rate equations as in [39]. The relevant radionuclide classes that are chemisorbed are removed from the vapor mass arrays and stored in chemisorption arrays. The chemisorption arrays correspond to six chemisorption classes. In accounting for radionuclide mass and decay power, the chemisorption classes are mapped back to the corresponding radionuclide class, so chemisorption output edits are ordered by the radionuclide class rather than by chemisorption class. Chemisorption shows up in the output edits as an additional column in the radionuclide mass edits.

2.9.1 Implementation

There are six chemisorption classes corresponding to the first six chemisorption relations in Table 1 of [34], reproduced below as Table 2.5. This table gives the chemisorption transport coefficients for chemisorption of several vapor species on different metal surfaces. The coefficients are those used to calculate the chemisorption mass transport coefficient, Eq.(2-136). These coefficients are accessible to the user via sensitivity coefficient C7160.

Table 2.5 Chemisorption Transport Coefficients

Species j	Surface i	A_{ij} (m/s)	E_{ij} (J/kg)	Reference
CsOH	Stainless Steel	0.139	5.96e7	Vine[40]
CsOH	Inconel	0.035	5.95e7	*
CsI	Stainless Steel	2.0e-7	0.0	Sallach[41]
CsI	Inconel	2.0e-6**	0.0	Sallach[41]
HI	Stainless Steel	5.5e-7	2.49e7	Williams[42]
I_2	Stainless Steel	9.0e-10	3.39e7	Williams[42]
Te	Stainless Steel	0.0	-	Sallach[43]
Te	Inconel	0.0	-	Sallach[43]

* Estimated from Sabathier[44] and Elrick[45] data.

** Cesium retained, Iodine released

There is a mapping array that establishes the correspondence between the chemisorption classes and the radionuclide classes. The default mapping is shown in Table 2.6. The radionuclide classes mapped are CsOH (2), I_2 (4), and CsI (16). There is no HI radionuclide class, and hence chemisorption class 5, HI, is mapped to radionuclide class 4, I_2.

Table 2.6 Chemisorption Class to RN Class Default Mapping

CA Class	Chemisorption Reaction	Radionuclide Class
1	CsOH on SS	CsOH (2)
2	CsOH on Inconel	
3	CsI on SS	CsI (16)
4	CsI on Inconel	
5	HI on SS	I_2 (4)
6	I_2 on SS	I_2 (4)

There also is an array that establishes the type of surface material for the chemisorption class; at present, this only contains mapping for stainless steel and Zircaloy, although this could be extended by adding more materials to the database or by implementing a method of mapping between user-defined materials and the chemisorption classes.

The chemisorption rate equation is

$$\frac{dM_{ij}}{dt} = A_i k_{ij} C_j \quad (2\text{-}135)$$

M_{ij} = mass of species *j* chemisorbed on surface *i* (kg)

A_i = area of surface *i* (m^2)

k_{ij} = chemisorption coefficient of species *j* on surface *i* (m/s)

C_j = concentration of species *j* in atmosphere (kg/m^3)

The mass chemisorption coefficient k_{ij} is temperature dependent and is given as

$$k_{ij} = a_{ij} e^{-E_{ij}/RT_i} \quad (2\text{-}136)$$

where

a_{ij} = chemisorption coefficient for species *j* on surface type *i* (m/s)

E_{ij} = activation energy for species *j* on surface type *i* (J/kg)

T_i = temperature of surface *i* (K)

R = universal gas constant (8314 J/kg-K)

As implemented, a finite-difference equation of the form

$$M_{ij}^n = M_{ij}^0 + \Delta t\ A_i k_i(T_i) C_j \quad (2\text{-}137)$$

is used to advance the chemisorption equations in time. These equations are applied sequentially in each control volume, for each surface, for each chemisorption class. After all equations are applied in a given volume, the total chemisorbed for each vapor radionuclide class is checked to ensure that the total is not greater than the total vapor mass; the chemisorbed masses for the current timestep are reduced by the ratio of vapor mass to chemisorbed mass if this occurs. The chemisorbed masses are then subtracted from the corresponding radionuclide vapor mass to complete the timestep.

2.9.2 Comparison to Exact Solution

An exact solution to the chemisorption equations can be found over a timestep for comparison to the numerical solution given above. Briefly, noting that the change in mass of a given species chemisorbed on a given surface is the same as the negative change in the species in the vapor phase, that the vapor concentration is the vapor mass divided by the component volume, and that the sum of the changes over all surfaces is the total change in vapor mass, the chemisorption equations can be summed and written in terms of the vapor species mass as

$$\frac{dM_j}{dt} = -\frac{M_j}{V}\sum_i k_{ij} A_i \qquad (2\text{-}138)$$

Defining an effective chemisorption rate for species *j* as

$$(kA)_j = \frac{1}{V}\sum_i k_{ij} A_i \qquad (2\text{-}139)$$

The solution to the above equation is

$$M_j(t) = M_j^0 e^{-(kA)_j} \qquad (2\text{-}140)$$

where

M_j^0 = vapor mass at time zero (kg).

If we apply the exact equation over a timestep and expand in a Taylor series about the beginning of the timestep,we have

$$M_j = M_j^0\left[1 - \Delta t(k\,A)_j + \ ...\right] \qquad (2\text{-}141)$$

after dropping higher powers of the timestep. This can be compared to the result of applying the finite-difference equations, which can be written as the sum also:

$$M_j = M_j^0\left(1 - \Delta t\frac{1}{V}\sum_i k_{ij} A_j\right) = M_j^0\left[1 - \Delta t(kA)_j\right] \qquad (2\text{-}142)$$

Comparison of the two above equations shows that the finite-difference result is the same as the Taylor series expansion of the exact solution carried out through linear terms of the timestep. At this point, it might be asked, why not use the exact solution? This is not done because this is an exact solution for the change in the vapor mass, not the change in chemisorbed mass for each surface. The change in chemisorbed mass for each surface in the control volume cannot be backed out of the vapor solution.

An exact solution for each surface could be formed, given the assumption that the vapor mass remains constant over the timestep; these could then be summed, leading to an equation for vapor mass involving the sum of the exponents, rather that the exponential of the sum. When expanded in a Taylor series, this results in the same equation as the above equation.

The above expansion in Taylor series gives a criterion for the accuracy of the solution:

$$\frac{k_{ij} A_i}{V} \Delta t << 1 \qquad (2\text{-}143)$$

The chemisorption coefficients are much less than 1, barring user input error (the largest coefficient, CsOH on stainless steel, is about 0.01 at 2500 K). The ratio A_j/V is less than 1 provided $V > 1$ m^3; for typical MELCOR timesteps of 1 to 5 s, the lower limit on V for the above inequality to hold is about 1 cc, so it appears that the above is true except for very small volumes.

2.9.3 Implementation Restrictions

As implemented, there is no provision for revaporization of chemisorbed species. Chemisorbed species thus stay on the absorbing surface. The first six chemisorption equations listed in the design report, Table 1, are implemented as the default classes in the model, because the deposition coefficients for tellurium, rows 7 and 8 in Table 1, are zero. Also, the model is set up to use the materials in the MELCOR material properties database as surface materials. As presently coded, surfaces consisting of user-defined materials cannot be made active for chemisorption because there is no method to relate them to the chemisorption classes. Also, the database does not contain Inconel, which means that only chemisorption of CsI, CsOH, HI, and I_2 on stainless steel and Zircaloy can occur. The coding framework is set up to use Inconel if it is added to the database in the future.

As noted in [48], there was no trace of iodine on the surface when CsI was chemisorbed on stainless steel. This means that, realistically, the iodine mass from the CsI should be transferred to the HI or I_2 class when chemisorbing. In the present model, the iodine from the chemisorbed CsI is transferred to the RN iodine class (4), so that it can be released on the next timestep if the surface is hot enough. Because the CsI chemisorbed class is mapped to the CsI vapor class, and this class is treated separately in MELCOR from the Cs and I_2 element classes, the chemisorbed Cs is transferred to the corresponding chemisorbed CsOH class (there currently are two each, for stainless steel and Inconel surfaces, see Table 2.6). This has two consequences: the CsI chemisorbed class is always zero, with the Cs showing up in the CsOH class, and the CsOH class must be active if the CsI class is active (this is the default).

2.10 Hygroscopic Aerosols

Aerosol particles that are soluble in water exhibit hygroscopic properties such that they can absorb moisture from an atmosphere with relative humidity less than 100%. This effect leads to a growth of the particle size as water vapor condenses onto the soluble particle. An important consequence of this growth in size (and mass) is an increase in the gravitational settling rate, and the subsequent depletion of airborne fission product aerosols.

The hygroscopic model in MELCOR is based on the Mason equation describing the diffusion of water vapor molecules to the surface of an aerosol particle, and the conduction of the latent heat of vaporization away from the particle and to the bulk atmosphere. The model presented here includes the solubility (hygroscopic) effect. In addition, the Kelvin effect, (surface tension) as well as noncontinuum (free molecule) effects, both of which are important for very small particle sizes, are considered.

In MELCOR 1.8.5, some improvements to the earlier MELCOR 1.8.4 implementation of the hygroscopic effect were included. Principally these included an updated and generalized method for calculating the chemical activity of the soluble particle, and a means of calculating a mean hygroscopic effect that considers the fact that not all aerosol materials are soluble and that multi-component aerosols can be comprised of varying proportions of soluble and non-soluble materials.

2.10.1 The Mason Equation for Particle Growth

The Mason equation [46] describes the rate of condensation or evaporation of water on an aerosol particle of radius *r* as:

$$\frac{dr}{dt} = \frac{1}{r}\frac{(S - S_r)}{a + b} \tag{2-144}$$

where,

$$S_r = A_r \cdot exp\left(\frac{2M_w\sigma}{RT_\infty\rho_w r}\right). \tag{2-145}$$

In the Mason equation, S is the atmosphere saturation ratio, or relative humidity and S_r is the effective saturation ratio at the particle surface. *(Note to reader: a subscript "r" in the subsequent text indicates that the quantity is size or radius dependent.)* The term (S -S_r) is the driving potential for condensation or evaporation. If the difference is positive, condensation occurs and if the difference is negative, evaporation takes place. The S_r term is a function of the chemical activity of the solution, A_r, which varies with the concentration of the solute (dissolved solid) within the solvent (water). The exponential term represents the Kelvin effect that resists condensation for small particles due to surface tension effects.

In Equation (2-144), the terms *a* and *b* determine the time constant for the particle growth rate and are defined as:

$$a = \left(\frac{\Delta h_f^2 M_w \rho_w}{RT_\infty k_a^*}\right) \tag{2-146}$$

$$b = \left(\frac{R T_\infty \rho_w}{D_v^* M_v p_{sat}(T_\infty)} \right). \qquad (2\text{-}147)$$

The term *a* accounts for the thermal conduction of the latent heat associated with condensation from the particle to the atmosphere, and the term *b* accounts for the diffusion of water vapor from the atmosphere to the particle surface. The other terms are defined in the following list of variables.

D^*_v	=	effective vapor diffusion coefficient
k^*_a	=	effective thermal conductivity of atmosphere
M_w	=	molecular weight of water
T_∞	=	bulk atmosphere temperature
$P_{sat}(T_\infty)$	=	saturation pressure of bulk atmosphere gas
R	=	gas constant
r	=	particle radius
S	=	atmosphere saturation ratio (RH/100)
S_r	=	saturation ratio at particle surface
ρ_w	=	density of water
Δh_f	=	heat of vaporization of water
σ	=	water surface tension

The activity, A_r, is a function of the concentration of the solute and is the dominant term in the driving potential for condensation or evaporation, S_r. In the MELCOR 1.8.4 [47] implementation, the activity was estimated using the van't Hoff formula as follows:

$$A_r = \frac{1}{1 + \sum_i \frac{I_i n_i}{n_w}} \qquad (2\text{-}148)$$

where n_i is the moles of solute *i*, n_w is moles of water, and I_i is the van't Hoff ionization constant for solute *i*. An important limitation in the 1.8.4 model was the fact that the sum in Equation (2-148) in effect was "simplified" by assuming that all aerosols were soluble and all had the same ionization factor. Hence, the effective form for calculating activity in MELCOR 1.8.4 was:

$$A_r = \frac{1}{1 + \frac{l_s n_s}{n_w}} \qquad (2\text{-}149)$$

where the subscript "s" refers to soluble aerosol (and all aerosols were considered soluble). In the present MELCOR 1.8.5 model, a generalized and more contemporary form for the activity is used as follows:

$$A_r = \exp\left[-\sum_i \frac{\nu_i n_i}{n_w} \right] \qquad (2\text{-}150)$$

where,

n_s, n_i = moles of dissolved solute in wet particle (may be less than total)

n_w = moles of water on wet particle

ν_i = ionization factor for solute molecule (usually 2).

Note that Equation (2-148) constitutes a linear approximation of Equation (2-150) for dilute solutions. The van't Hoff factors provided some correction to the linearization for concentrated solutions. In Equation (2-150) the term ν_i represents the number of ions formed when the solute becomes dissolved. This value is normally two. The form for activity in Equation (2-150) is more commonly encountered in chemistry texts describing the solute effect and is similar to that used in the CONTAIN [48] model for hygroscopic growth. Additionally, the present activity form estimates a *net* activity that is a mole weighted average of all aerosol materials within a given size range - soluble and insoluble. Finally, the value of n_i in the present model is limited by the saturation solubility of the aerosol component *i*. The importance of the revised activity formula is as follows. If the aerosol materials are insoluble or of low solubility, the aerosol exhibits low hygroscopic behavior; if the proportion of soluble materials in the aerosol composition is large, then a proportionally larger hygroscopic effect results. This replaces the "all or nothing" treatment that was present in the MELCOR 1.8.4 model.

The activity term, A_r, is a function of the wet particle radius since, as the particle grows by condensation of water, the concentration of the solute decreases. When the soluble particle is virtually dry, any water on the drop acquires a concentration of dissolved solute that is limited to the maximum solubility of the solute (that is, the solution is saturated with solute). At this point the chemical activity is at its lowest value, and as a result, the driving potential, S-S_r, is at its highest value. Until sufficient water is acquired to completely dissolve the aerosol solid material, the activity remains at this minimum value. However, after this point the concentration of the solute begins to drop below the saturation value, resulting in an increase in the activity. When infinitely dilute, the activity approaches 1. In general, the value of S_r is dominated by the activity. As the particle acquires more

water, the value of S_r increases thereby increasing the atmospheric humidity necessary to drive further condensation.

2.10.2 Transition Regime Corrections to the Mason Equation

The particle growth rate Equations (2-144) to (2-147), make use of *effective* values for the air thermal conductivity and the diffusion coefficient for water vapor molecules in moist air. These effective values approach the nominal conductivity and diffusion coefficient values when the aerosol particle radius is large in comparison to the mean free path of the water vapor molecules. However, when the aerosol particle radius is on the order of the vapor molecule mean free path, these factors introduce correction terms to the otherwise continuum regime Mason model. Based on the derivation presented in Prupracher and Klett [46], the effective values of thermal conductivity and diffusion coefficient are determined by:

$$k_a^* = \frac{k_a}{\left(\frac{r}{r+\Delta_T}\right)+\left(\frac{k_a}{r\,\alpha_T\,\rho_a\,c_{p,a}}\left(\frac{2\pi\,M_a}{R\,T_a}\right)^{1/2}\right)} \qquad (2\text{-}151)$$

$$D_v^* = \frac{D_V}{\left(\frac{r}{r+\Delta_V}\right)+\frac{D_V}{r\,\alpha_C}\left(\frac{2\pi M_a}{RT_a}\right)^{1/2}} \qquad (2\text{-}152)$$

where

α_c = 0.036, the condensation coefficient,

α_T = 0.7, thermal accommodation coefficient,

$c_{p,a}$ = atmosphere constant pressure specific heat,

λ = vapor molecular mean free path,

Δ_V = vapor jump distance $\approx 1.32\lambda$,

Δ_T = thermal jump distance $\approx \lambda$, and

M_a = atmosphere molecular weight.

2.10.3 MELCOR Solution to the Mason Equation

In the MELCOR implementation of the Mason expression, Equation (2-144) is rewritten as

$$\frac{dr^2}{dt} = \frac{2 \cdot \left(S - S_r\right)}{a + b} \tag{2-153}$$

which, when expressed in an implicit backwards difference form becomes:

$$\left(r_{new}^2 \; - \; r_{old}^2 \right) \; = \; \frac{2 \cdot \left(S - S_r\right)}{a + b} \cdot \left(t_{new} - t_{old}\right) \tag{2-154}$$

where all of the right-hand side terms are evaluated at the end of timestep conditions. A zero-finder routine is used to solve for the new value of r^2 that results in a value for S_r satisfying Equation (2-154). This numerical method is fast and stable and the small amount of "undershooting" that results from the backward difference is inconsequential in that the characteristic time associated with Equation (2-153) attaining steady state is short in comparison to a typical MELCOR time step.

2.10.4 User Suggestions Concerning Use of the Hygroscopic Model

MELCOR uses Equation (2-154) to predict the growth of a representative particle in each of the size sections, and from this determines a section to section mass transfer that reflects this growth. MELCOR uses MAEROS to perform other aerosol dynamics calculations, including agglomeration and deposition. Understanding the overall model requires understanding a little about MAEROS. The user is encouraged to review the sections in this manual pertaining to the MAEROS model. For convenience, the following review is given. MELCOR of course uses a ***Class*** grouping to represent fission product species that have common physical/chemical characteristics such as volatility. MELCOR's aerosol mechanics model (MAEROS) recognizes and operates on the aerosol portions of these fission product (radioactive and non-radioactive) classes. For the purposes of performing more economic aerosol mechanics calculations, MELCOR allows the user to define aerosol ***Components***, which are groupings of one or more fission product ***Classes***. These components can have distinct size distributions. The size distributions are characterized by the amount of aerosol mass within a range of aerosol particle sizes. These size ranges are referred to as ***Sections***, or sometimes as size bins. MAEROS homogenizes the section populations of aerosol *classes* that are members of the same aerosol *component*, even if the user sources in the classes with different size distributions.

The hygroscopic growth model operates on the section populations without any consideration of the component definition. That is to say that all particles of all RN classes within a given *section* (regardless of their component assignment) are used to determine

the mean activity using Equation (2-150), which in turn is used to determine particle growth by condensation. As a result, all aerosol mass associated with all radionuclide *classes* that reside in a given size *section* are transferred to larger (or smaller) size sections proportionally by the hygroscopic growth routines.

This means that non-hygroscopic particles residing in a size section that is dominated by hygroscopic particles are moved to different size sections along with their hygroscopic companions, and conversely, hygroscopic particles residing in a size section that is dominated by non-hygroscopic particles are retained in the section to the extent determined by their non-hygroscopic companions. However, if hygroscopic and non-hygroscopic particles reside in different size sections (which can only be represented by MAEROS if they are assigned to different aerosol components), the two particle populations behave independently. The hygroscopic particles grow or shrink, depending on the relative humidity, while the non-hygroscopic ones remain the same size (after losing any water that they may have contained). This makes it important that water and non-water aerosol components be assigned to different aerosol classes. The MELCOR 1.8.5 code release has a default configuration of two aerosol components, one for water class aerosol that subsume fog droplets formed as a result of thermodynamic conditions in the atmosphere into the smallest size section, and one for non-water class aerosols. Three aerosol components are recommended if it is desired to track hygroscopic and non-hygroscopic aerosols that have different size distributions.

2.11 Flashing Jet Impaction Model

2.11.1 Introduction

The flashing jet impaction model for MELCOR was developed to describe the impingement of a flashing water jet on a plate. The model is based on the one described in [49], although it has been somewhat simplified to fit the MELCOR code architecture. The model (see Figure 2.3) describes a flashing water jet that enters a volume at lower pressure than the inlet pressure, flashes, and then expands according to the regular jet expansion formula until it is deflected by a plate. Water droplets larger than a cutoff diameter hit the plate and are removed from the jet.

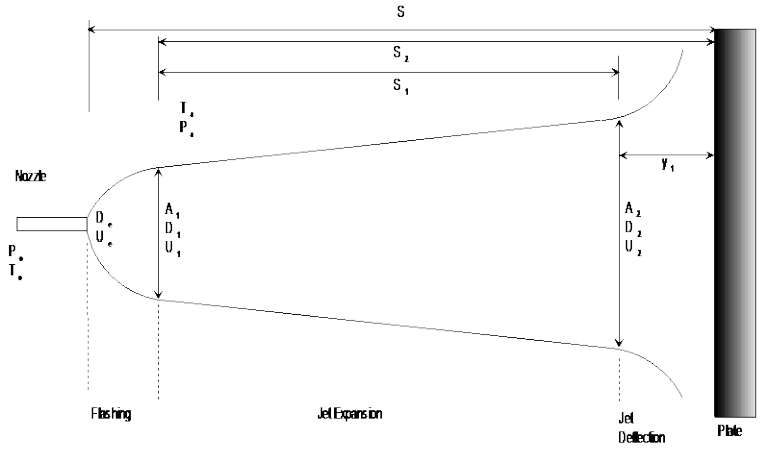


Figure 2.3 Flashing jet impaction model

As adapted for MELCOR, the incoming water jet is either a flowpath entering a target control volume (see Figure 2.4) or a source to the control volume giving the appropriate mass flow, temperature, pressure, etc. parameters. The water droplet size distribution is determined by the flashing model (see FL Package Reference Manual) and sourced into the RN aerosol package. The target volume, which is intended to allow better simulation of atmosphere conditions around the jet nozzle, connects to the main volume through another flowpath. The main volume contains one or more heat structures that simulate the plate(s) on which the jet impacts. The water droplets removed by impaction become part of the water film on the heat structure and can drain according to the existing film model in MELCOR.

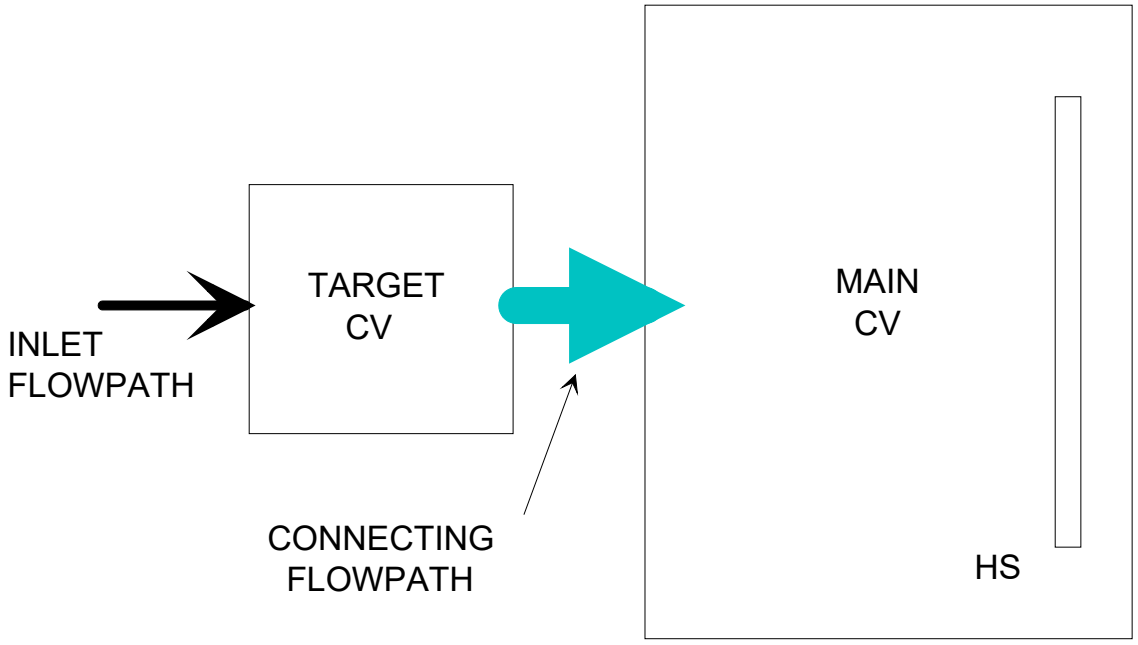


Figure 2.4 Jet Impaction as set up in MELCOR

2.11.2 Model Description

The model description as given in [49] starts with water entering a volume from a jet nozzle at a given source pressure, mass flow and saturation conditions. The entrance velocity U_e can then be obtained as

$$U_e = \frac{G v_e}{A_e} \tag{2-155}$$

where G is the mass flow, v_e is the entrance specific volume of water, and A_e is the jet entrance area. The jet entrance pressure P_e, which is lower than the source pressure after accounting for friction and dynamic head, is taken in MELCOR to be the same as the source pressure P_o. The jet then flashes to a larger size, typically 10-20x the entrance diameter, and attains equilibrium with the lower pressure in the volume. This occurs within 4-8 jet diameters from the jet nozzle. Assuming adiabatic expansion, the quality and hence the mixture specific volume can be determined from

$$h_e = (1 - x) h_\ell + x h_m \tag{2-156}$$

and

$$v_m = (1 - x) v_\ell + x v_m \tag{2-157}$$

where h is the water specific enthalpy, x is the quality after flashing, and the subscripts e, ℓ, v, and m indicate entrance, liquid, vapor, and mixture respectively. The entrance pressure and ambient pressure are used to obtain the jet velocity at pressure equilibrium from the momentum equation

$$U_1 = U_e + (P_e - P_a)\frac{A_e}{G} \tag{2-158}$$

where U_1 is the velocity at equilibrium and P_a is the ambient pressure in the target volume. The jet area at equilibrium can now be obtained as

$$A_1 = \frac{G v_m}{U_1} \tag{2-159}$$

where A_1 is the area of the jet at equilibrium and v_m is the mixture specific volume. Assuming a circular jet, the jet diameter D_1 can then be calculated from A_1.

The jet velocity and diameter at the deflection distance can be obtained using regular equations for jet expansion. The jet velocity, assumed to vary only in the axial direction, is then calculated using the momentum balance equation

$$\rho_m A_1 U_1^2 = \rho_m A_{2m} U_2^2 + \rho_a (A_2 - A_{2m}) U_2^2 \tag{2-160}$$

which accounts for the entrained gas from the ambient atmosphere. In this equation, ρ_m is the jet mixture density, A_{2m} is the jet area at the deflection point of the steam-water mixture only, A_2 is the jet area including the entrained atmosphere, ρ_a is the atmosphere ambient density, and U_2 is the jet velocity at the deflection point. By substituting in for U_1 and U_2 we can get a quadratic equation involving only the areas and the densities, which can be solved for the flow area of the water-steam mixture:

$$A_{2m} = A_1 \left\{ -\left(-1 + \frac{\rho_a}{\rho_m} \right) + \left[\left(-1 + \frac{\rho_a}{\rho_m} \right)^2 + 4 \left(\frac{A_2}{A_1} \right) \frac{\rho_a}{\rho_m} \right]^{1/2} \right\} / 2 \tag{2-161}$$

We also need A_2, which can be obtained assuming the jet expands linearly at an angle of 10° [50]:

$$D_2 = D_1 \left(1 + \frac{2S_2}{D_1} tan(10^\circ) \right) \tag{2-162}$$

We can now solve for A_2, A_{2m}, and finally U_2.

The distance the jet deflects from the plate is given by the correlation [51]

$$y_1/D_1 = 1.2 \quad for \quad S_1/D_1 < 6.8, \tag{2-163}$$

$$y_1/D_1 = 0.153(1 + S_1/D_1) \quad for \quad S_1/D_1 > 6.8 \tag{2-164}$$

where y_1 is the deflection distance from the plate and S_1 is the jet-plate separation. If the deflection distance is greater than the jet-to-plate separation S_1, then the velocity and diameter at the deflection are assumed to be the same as those at equilibrium.

The correlation given in [49] for a jet of general shape is used to relate the cutoff Stokes number for the droplets and the cutoff water droplet diameter:

$$St_{m,50\%} = \rho_p d_{p,50\%}^2 UC_u/(9\mu D_e) = 0.268 \tag{2-165}$$

where ρ_p is the aerosol droplet density, $d_{p,50\%}$ is the droplet cutoff diameter, U is the droplet (jet) velocity, C_u is the Cunningham slip factor, and μ is the atmosphere viscosity.

U_2 is used for U in Equation (2-165) to determine the cutoff droplet size from the cutoff Stokes number. Droplets with a size greater than or equal to the cutoff diameter are assumed to impact the plate and are removed from the jet (in MELCOR, removed from the volume's atmosphere aerosol size distribution).

2.12 Iodine Pool Model

2.12.1 Introduction

The potential release of radioactive iodine as a result of a core damage accident in a nuclear power plant has long been a principal concern of reactor safety and consequence analyses. Iodine in particular is a concern because of its major contribution to the radiological hazard to the environment. A specific model devoted to the chemistry of iodine in reactor containments under accident conditions is needed in the MELCOR computer code because of the unique chemical properties of iodine and the severe consequence attributed to the release of the radioactive isotopes of iodine to the environment. Possible release of iodine has always played a significant role in the regulation of nuclear reactors. In early assessments of iodine consequences, it was assumed that iodine would be released to the reactor containment as a gaseous species. About one quarter of the initial core inventory was assumed to remain in the containment atmosphere, available for release to the environment. However, research over the last 15 years has shown it to be more likely that most of the iodine is released to the containment atmosphere as aerosol particles, principally CsI. The Revised Accident Source Term (NUREG-1465) [52] assumes that at least 95% of the iodine reaching the containment is in aerosol form. Iodine within the containment atmosphere is able to pass through containment leak paths to the environment, thereby resulting in a dose to the public with ensuing consequences. Reduction of releases therefore requires control of atmospheric iodine concentration. This can be accomplished by causing the iodine to remain confined in aqueous forms in pools and sumps. Advanced reactor designs may

incorporate chemical systems to keep the atmospheric concentration of iodine low by trapping iodine in aqueous forms and hence limit risk. An important use for MELCOR is to assess the adequacy of these designs and identify processes and mechanisms that may defeat the intent of these systems.

Light water reactor containment temperatures can be expected to condense any residual cesium iodide vapors and form aerosols. These containments also include substantial quantities of water that can trap aerosol particles during severe accidents. For example, the condensation of steam formed during the core degradation processes takes place to a large extent within the containment. Trapping of most radionuclides in water effectively removes these radionuclides from further consideration in the analysis of the public consequences of reactor accidents by removing them from the containment atmosphere. However, radioactive iodine may not remain trapped in water because of its relatively dynamic chemical behavior. Engineered safety systems, such as sprays and suppression pools, are still effective mechanisms for scrubbing particulate iodine from the system and trapping it in the aqueous phase. However, there are important processes that can regenerate gaseous forms of iodine that release into the containment atmosphere from the water, thus becoming available for release to the environment for long times after the accident initiation.

The chemical and radiolytic oxidation of iodine in the pool can lead to the formation of a variety of chemical forms of iodine, such as elemental iodine and volatile organic iodides. The formation of volatile iodine in the pool is followed by a “partitioning” of the iodine between the pool and the atmosphere. This partitioning is important primarily in the longer term phases of the accident after the natural and engineered safety features have removed the other radioactive aerosols released during the accident. The formation of volatile forms of iodine in solution is dependent not only upon the dose rate to the aqueous phase but also on temperature, the hydrogen ion concentration (conventionally expressed as pH), and the total iodine concentration. It has been shown experimentally that large fractions of the iodine released from the reactor core can be expected to reside within the containment atmosphere in a volatile form when pH is not controlled to an alkalinity level greater than 7 [52]. It has also been observed that irradiation induced release of acids from the wall surface coatings, cable insulation, and the containment air lowers the pH [53]. However, the combination of high pH and high irradiation has not been thoroughly tested. In addition, the effect of other materials on the pool chemistry is not well established. Consequently, any model must be adaptable to the results of ongoing research. This fact is considered in the design of the MELCOR model, and provision is made to accommodate new information as it becomes available.

2.12.2 Features of Iodine Pool Model

The iodine pool model addresses these concerns. It embodies the current state of knowledge in a form that can be easily modified as current research yields new results. It uses the known chemistry to predict what factors affect the iodine concentration in the

atmosphere, while allowing for additional chemical reactions. In the containment atmosphere, where gas-phase behavior is important, there are submodels relating the radiolysis of the air and cable insulation to the generation of nitric acid and hydrochloric acid, respectively. On the structural surfaces, provision is made to account for the type of surface, thus allowing the extension to treat the effects of different paints and other surface coatings on iodine behavior. In the water pool, where liquid phase behavior is important, the model determines the pH based upon the user controlled boric acid and phosphate buffering, the effects of cesium hydroxide, cesium iodide and control rod silver released by the accident scenario chosen, and the effects of the acids introduced from the containment atmosphere due to radiolysis. The aqueous pool chemistry model then determines the speciation of iodine, particularly the important elemental, molecular, and organic forms, over the full range of pH. Thus, chemical systems that control pool pH can be examined as well as pools and films on surfaces that have no pH controls. With this combination of features, the iodine pool model allows for the ability to conduct sensitivity studies and incorporate new effects found in the course of ongoing research.

2.12.3 Criteria for Application of the Model

A MELCOR calculation typically involves several volumes with differing properties. When the model is invoked, it is applied everywhere. The full model is used only in volumes with a pool, atmosphere, and iodine. Acid generation by radiolysis is calculated in volumes with only an atmosphere, as these acids can be transported by MELCOR to other connected volumes. However, the aqueous chemistry model was designed for volumes where the pressure is less than 10 atm and the liquid temperature is less than 423 K, corresponding to conditions in a commercial reactor containment. If these limits are exceeded, the pool model may become invalid. In such cases, the aqueous chemistry model is not used.

The effects of partitioning of iodine between the aqueous and gaseous phases are typically only important in the late term phases of an accident (after about 10 hours for the NUREG-1465 severe accident [52]). By this time, most of the iodine in a MELCOR calculation has been transported to volumes where the pool model is valid. At earlier times, radionuclide behavior is dominated by other phenomena. Thus, the limitations on the applicability of the aqueous chemistry model should have little impact on the ability to calculate the important phenomena in reactor accident sequences.

2.12.4 Detailed Description of the Model

The model involves four areas of modeling, as shown schematically in Figure 2.5. The area labeled as one (1) indicates the transport of iodine species among the walls, the bulk gas, and the pool. This part of the model interacts directly with the MELCOR intra-cell mass transport coefficient (TRAP-MELT like) solution and contributes to determining the structural surface concentration of the chemically and physically bound iodine species by using kinetic reactions to determine a transport rate. The change in pool depth from timestep to timestep changes in heat structure surface area, and transfer of iodine between pools and films is

handled by existing MELCOR coding. The area labeled as two (2) is the containment atmosphere part of the model. It determines the radiolytic formation of acids and the gas-phase destruction and formation of iodine species. Species of iodine added to the cell atmosphere come from the pool, the structural surfaces, and adjacent cells (e.g., the reactor coolant system break location). The area labeled as three (3) is concerned with the hydrogen ion concentration (i.e., pH), and accounts for the effects of the acids and bases introduced into the pool as well as the removal of iodine due to silver. The pH solution is typically dominated by the effect from the initial buffering of the pool. Thus, the model does not currently account for the hydrolysis of the other materials that may be in the pool, for example, cadmium, sludge, iron, and uranium. The area labeled as four (4) is the aqueous iodine chemistry model where the iodine, hydrogen, oxygen, carbon, iron, and electron balance equations are solved.

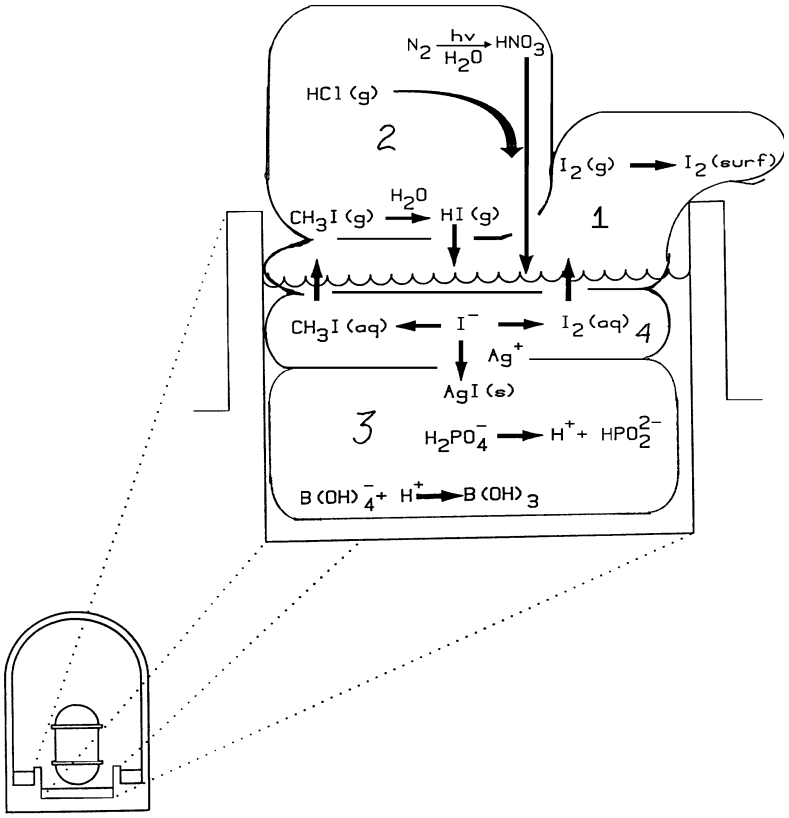


Figure 2.5 Schematic Representation of the Iodine Transformations Considered

The formulation chosen uses a dilute solution approximation that allows the effect of water radiolysis and radiolytic reactions to be explicitly included and should allow the results of current experimental studies to be compared. The approach adopted by Weber et al. [54] is modified here to include a more comprehensive set of chemical reactions and to

explicitly include dose rate effects for the radiolysis while retaining the quasi-steady approximations for the dynamic equations.

2.12.5 Interaction with MELCOR

In MELCOR, intra-cell transport processes, for example condensation and aerosol deposition in a volume, are followed by inter-cell transport of material, for example silver and iodine moving from the reactor coolant system to the containment. The iodine chemistry model can be thought of in terms of intra-cell transport. The iodine model processes affect the distribution of iodine among the pool, the atmosphere, and the heat structures in various control volumes. Thus, for a PWR, after a mix of water and radionuclides has been removed from the containment by deposition or through the action of the sprays and placed into the sump, this model allows MELCOR to distribute the iodine among the sump, the containment open volume, and the walls. Similarly, in a BWR, after the radionuclides have been placed in the wetwell, this model allows MELCOR to distribute the iodine among the suppression pool, the vapor space above it, and the wetwell walls.

Figure 2.6 shows the relationship between the iodine models and the balance of the MELCOR code. Volumes 1 and 2 are typical MELCOR hydrodynamic control volumes where a variety of processes take place. As shown for volume 1, these include scrubbing of aerosols from the atmosphere by sprays, deposition of aerosols onto structural surfaces with water films draining into the pool, and interface transport between pool and atmosphere. MELCOR also accounts for the transport of material between volumes. Not all MELCOR processes are shown; for example, the heat transfer processes are not indicated. None of these MELCOR processes are affected by adding the iodine model. The iodine model performs aqueous chemistry calculations within existing pool regions of MELCOR control volumes. That is, based upon a species distribution and the radiation environment it determines the local pH and the quantity of elemental and organic iodine available at the pool-atmosphere interface. The model also performs vapor chemistry calculations within the existing atmosphere regions of MELCOR control volumes. That is, based upon a species distribution and the radiation environment; it determines the radiolytic formation of acids and destruction of iodine. These submodels are shown as the two add-on boxes above and below volume 1 in Figure 2.6. The model determines the transport and partitioning of the iodine species between the pool and atmosphere regions, allowing MELCOR to determine the late phase concentration of iodine in the atmosphere.

MELCOR determines the flux of important species into and out of all volumes within the inter-volume transport calculation. For the purposes of this model, important transported species include: the original thirteen (13) MELCOR radionuclide classes, used to determine the distribution of radiation sources in the control volume (xenon, cesium hydroxide, barium, elemental iodine, tellurium, ruthenium, molybdenum, cerium, lanthanum, uranium dioxide, cadmium, tin, and boron classes); four (4) species to control

the hydrogen ion concentration in the pool (boric acid, cesium iodide, and phosphate are new; cesium hydroxide can be represented by existing class 2); four (4) in the atmosphere (methyl iodine, hydrochloric acid, and nitric acid are new; iodine is represented as class 4); two (2) deposited species (non-volatile form of iodine and methyl iodine to allow for surface chemistry); one (1) pool species that acts as a sink for iodine (silver, represented as existing class 12); and three (3) water pool chemistry species (silver iodine and methyl iodine are new; aqueous iodine is represented as class 4). There are many more species included in the aqueous pool chemistry model, including the two main species, elemental and molecular iodine; however, due to the equilibrium nature of the chemistry model, these species do not all need to be transported—the model is initialized at the beginning of a timestep by a small subset of the species, principally iodine, and creates the speciation for the conditions existing in the pool during that timestep. Obviously, these new species are not all radionuclides and do not need to be examined in all code modules, i.e., they do not all need to be full RN classes. Table 2.7 shows some of the main and secondary species available from the model.

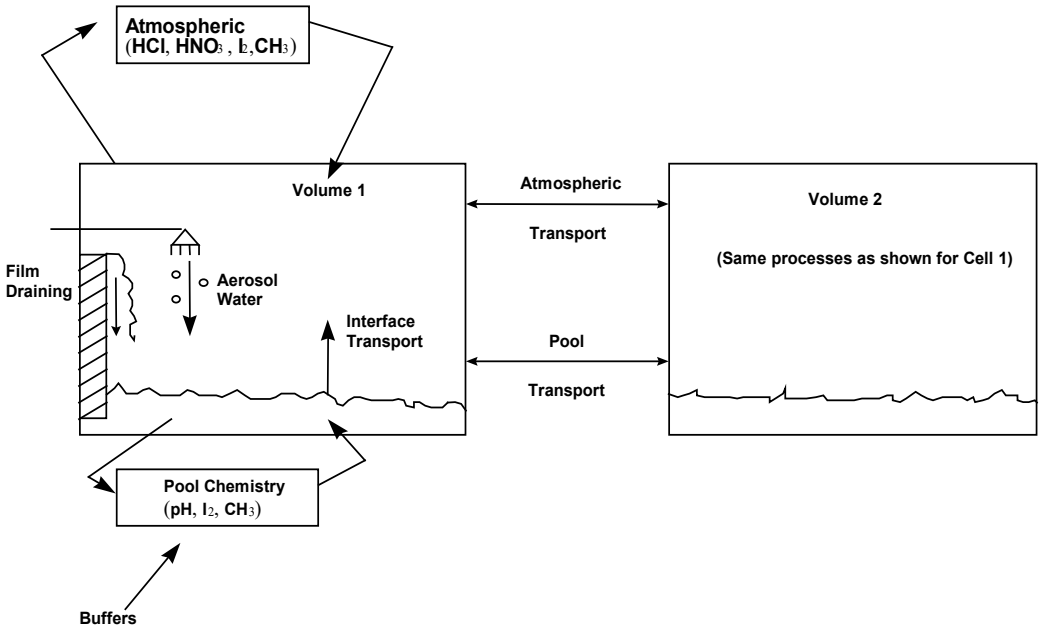


Figure 2.6 Interface between MELCOR and the Iodine Pool Model

Table 2.7 Representative Species in Iodine Pool Model

Species	RN Class (Y/N)?	Species	RN Class (Y/N)?
I2	Y (4)	HOI-	N
I-	Y (4 or 16)	H2O2	N
I3-	N	O2-	N
IO-	N	HO2-	N
IO3-	N	CH3I	Y
I2OH	N		

Classes 14 (water) and 15 (concrete) are included in the original RN list—even though they are not “radionuclides”—because they form aerosols. Many current calculations include cesium iodide as a user-defined 16th class. CsI has been changed as part of the iodine pool model update to be a default RN class.

Transport of air and water, also used by the iodine pool model, is done by the MELCOR hydrodynamics module CVH. To use the pool model, it is necessary that the atmosphere components hydrogen, methane, and carbon dioxide be initialized in MELCOR input, as well as the usual atmosphere constituents (nitrogen, oxygen, and steam).

MELCOR determines the liquid, vapor, and heat structure surface temperatures and vapor pressure within the volume, within the energy transport calculation. With this information, the iodine models determine the intra-cell transport coefficients for the iodine species, that is, those coefficients determining the transport of elemental and organic iodine between the pool and the vapor space and between the vapor space and the heat structures. The model also determines the change from volatile to non-volatile iodine species on the surfaces, the change from one iodine species to another in the pool (including silver iodide), and the homogeneous destruction of iodine species in the atmosphere.

2.12.6 Order of Calculation of Model

The order of calculation in a control volume for the model is shown in Figure 2.7. This figure shows that the main functions of the model are carried out in a simple consecutive order, starting with the check for atmosphere volume in the upper left corner (Block 1) and continuing to the output block in the lower right corner (Block 14). Starting with the check for atmosphere volume, Block 1 in Figure 2.7, the calculation proceeds as follows:

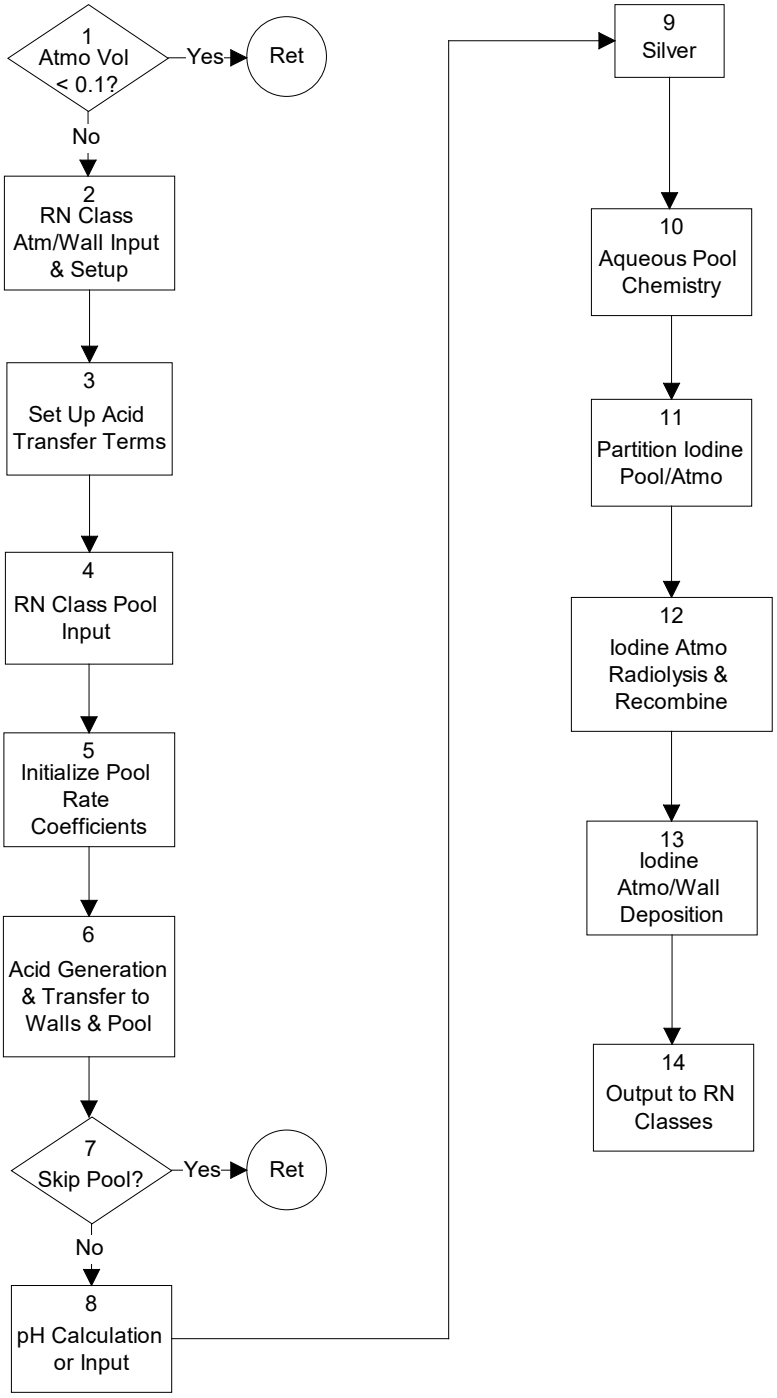


Figure 2.7 Calculation Flow of MELCOR Iodine Pool Model

(1) The atmosphere volume in the control volume is checked against a limit with a default of 0.1 m^3. If this test is not satisfied, the rest of the model is skipped for this control volume.

(2) RN class input and atmosphere/pool setup: the atmosphere and pool driver species are initialized at the beginning of the timestep from the MELCOR RN classes. This is only done for the atmosphere and walls at this point in the calculation. In the atmosphere, these are iodine (class 4), methyl iodine (class 17), hydrochloric and nitric acids (classes 18 and 19), and nitrogen, steam, oxygen, hydrogen, carbon dioxide, and methane (hydrodynamic materials). Atmosphere, pool, and wall areas and volumes are set up. Wall species for physically and chemically bound iodine and methyl iodide, and deposited nitric and hydrochloric acid on wet walls, are initialized from extended MELCOR chemisorption classes.

(3) The terms in the ordinary differential equations describing mass transport of hydrochloric and nitric acid from the atmosphere to the walls are set up, as are the radiolysis generation terms.

(4) The pool species are initialized from the MELCOR RN classes. These are iodine (class 4 and 16, CsI), the buffers boric acid (class 20) and phosphate (class 22), hydrochloric and nitric acid (classes 18 and 19), cations (CsI, class 16), silver (class 12), and iron (class 7). Although silver iodide is also transported as an RN class, the pool silver iodide does not need to be initialized, as silver acts only as a sink for iodine, not a source, and hence silver iodide (once formed) plays no further role in the pool chemistry.

(5) The rate coefficients for the pool chemistry calculation are initialized. These are used later in the aqueous chemistry routine.

(6) The calculation of mass transport for hydrochloric and nitric acid is done. This includes the radiolysis generation rates, transport between the atmosphere and wall surfaces, and transport between atmosphere and pool. This last step is necessary to have the updated pool acid concentrations available for the pH calculation.

(7) The conditions for using the full iodine pool model are checked against limits here. These include the presence of iodine, atmospheric pressure less than 1 MPa, pool present, and pool temperature less than 425 K. If these conditions are not satisfied, then the rest of the pool calculation is skipped. There is a user input flag that overrides the iodine criterion, allowing pool hydrolysis calculations to be done.

(8) The pH calculation is performed based on the relative molar concentrations of acids and bases in the pool. Alternatively, the pH can be directly entered in user input via tabular or control functions, or an external data file.

(9) A fraction of the silver present (set to 10^{-6}) is assumed chemically active and can remove some of the iodine in the pool as silver iodide, acting as a sink.

(10) The aqueous pool chemistry solver is called. This is a quasi-equilibrium solver and assumes steady-state conditions. The iodine from the MELCOR RN classes 4 (I_2) and 16 CsI is treated as an initial inventory of I. The aqueous chemistry model performs the speciation of the iodine each timestep, based on the pH and radiolysis in the pool (see Table 2.7 for major and secondary species available for output in MELCOR).

(11) The molar concentrations of iodine and methyl iodide in the pool are used to determine a pool surface concentration. This is used together with atmosphere conditions to partition the iodine and methyl iodide between the pool and atmosphere, giving new concentrations in the pool and atmosphere.

(12) The atmospheric iodine and methyl iodide concentrations are further modified by atmospheric radiolysis and thermal and concentration-dependent destruction rates to form free iodine; the final concentrations are determined by a recombination step using equilibrium coefficients.

(13) The atmosphere iodine and methyl iodide concentrations are used together with the wall concentrations to determine mass transport between the atmosphere and dry wall surfaces. Radiolysis at painted walls is included.

(14) Results of the pool model calculation are output. The relevant RN classes are updated, (see block 2 and 4 descriptions), including the silver iodine class. The silver iodide class is necessary to maintain mass conservation. On output, available cations (Cs) is combined with available I^- in the pool to form the new CsI (class 16) mass, and uncombined Cs or I^- are added to the CsOH (class 2) or I_2 (class 4) masses, respectively. The main iodine species, I_2 and I^-, are otherwise output as class 4. Other secondary species are also added into class 4 to maintain mass conservation. Updated wall concentrations are also output. The pH of the pool is available as a MELCOR plot variable. The masses and concentrations of the RN classes for the pool and atmosphere (transported species) are likewise available via control functions. A list of main and secondary species available via control function is shown in Table 2.7.

2.12.7 Submodels in the Iodine Pool Model

There are seven main submodels in the iodine pool model. These are detailed below, starting with the acid generation and transport models.

2.12.7.1 Acid Generation and Transfer to Walls and Pool

Formation of nitric acid in the atmosphere by radiolysis is calculated using the rate

$$\dot{S}_{HNO_3} = 5.45 \times 10^{-7} M_{N_2} \dot{D}_{atm} \quad \text{(2-166)}$$

where

$\dot{S}_{HNO_3}$ = formation rate of nitric acid by radiolysis (kg-mole/s)

M_{N_2} = mass of nitrogen in the atmosphere (kg)

$\dot{D}_{atm}$ = atmosphere dose rate (MRad/hr)

and the constant has the appropriate units

Formation of hydrochloric acid is assumed to occur via radiolysis of plastic wire insulation in a control volume and go into the atmosphere instantly. The rate is given as

$$\dot{S}_{HCL} = 2.88 \times 10^{-7} M_{cable} \dot{D}_{cable} \quad (2\text{-}167)$$

where

$\dot{S}_{HCL}$ = rate of formation of HCl by radiolysis of wire cable insulation (kg-mole/s)

M_{cable} = mass of cable insulation in control volume (kg)

$\dot{D}_{cable}$ = cable dose rate (MRad/hr)

and the constant has the appropriate units.

Nitric and hydrochloric acids in the atmosphere can be deposited in the water films on wet walls via a non-reversible mass transport equation of the form

$$\frac{dC_{w,acid,n}}{dt} = k_{w,acid,n} C_{atm,acid} \quad (2\text{-}168)$$

where

$C_{w,acid,n}$ = moles of acid on wall surface *n* (kg-mole)

$k_{w,acid,n}$ = transport coefficient from atmosphere to wall *n* for acid (m/s)

$C_{atm,acid}$ = atmospheric moles of acid (kg-mole)

and the subscript *acid* refers either to nitric or hydrochloric acid. A similar equation is used for transport from the atmosphere to the pool. The new amount of acid in the atmosphere is determined by summing up the transport to all the wet walls in a control volume and the pool (if present) to get

$$\frac{dC_{atm,acid}}{dt} = -C_{atm,acid}\frac{1}{V_{atm}}\left(\sum_n k_{w,acid,n}A_{w,n} + k_{p,acid}A_{pool}\right) + \dot{S}_{acid} \quad (2\text{-}169)$$

where

$k_{w,acid,n}$	=	transport coefficient from atmosphere to wall surface *n* for acid (m/s)
$A_{w,n}$	=	wall *n* surface area (m^2)
$k_{p,acid}$	=	transport coefficient from atmosphere to pool for acid (m/s)
A_{pool}	=	pool-atmosphere surface area (m^2)
V_{atm}	=	atmospheric volume (m^3)
$\dot{S}_{acid}$	=	formation rate of acid (kg-mole/s).

This can be solved analytically over the timestep as

$$C_{atm,acid}(t) = C_{atm,acid}(t_0)\,exp\left(-k_{eff,acid}t\right) + \frac{\dot{S}_{acid}}{V_{atm}k_{eff,acid}}\left(1 - exp\left(-k_{eff,acid}t\right)\right) \quad (2\text{-}170)$$

where $k_{eff,acid}$ is defined by

$$k_{eff,acid} = \frac{1}{V_{atm}}\left(\sum_n k_{w,acid,n}A_n + k_{p,acid}A_{pool}\right) \quad (2\text{-}171)$$

The change in amount of wall acid can be expressed in terms of the change in atmospheric acid as

$$C_{w,acid,n}(t) = C_{w,acid,n}(t_0) + \frac{k_{w,acid,n}}{k_{eff,acid}}\left(\left[C_{atm,acid}(t) - C_{atm,acid}(t_0)\right] + \dot{S}_{acid}(t - t_0)\right) \quad (2\text{-}172)$$

and a similar equation applies for the change in pool acid,

$$C_{p,acid}(t) = C_{p,acid}(t_0) + \frac{k_{p,acid}}{k_{eff,acid}}\left(\left[C_{atm,acid}(t) - C_{atm,acid}(t_0)\right] + \dot{S}_{acid}(t - t_0)\right) \quad (2\text{-}173)$$

Acids deposited in wall films are transported to the pool or other surfaces using the MELCOR film transport model.

2.12.7.2 Pool pH Calculation

The pool pH is determined either from an acid-base balance or set via user input. The pH calculation is done by first performing a charge balance of the acids and bases to estimate pH, and then performing an iteration over the species and charge balance to get the final pH.

The first step is to estimate the hydrogen ion concentration (or pH) from a charge balance on the phosphate (Na_3PO_4), cation (Cs), nitric and hydrochloric acid concentrations, as

$$\Delta Z = 3x_{NaP} + x_{Cs} - x_{HNO_3} - x_{HCl} \qquad (2\text{-}174)$$

where (kmole=10^3 mole)

ΔZ = charge balance (kmole/m^3)

x_{NaP} = phosphate concentration (Na_3PO_4), (kmole/m^3)

x_{Cs} = cation concentration (cesium), (kmole/m^3)

x_{HNO_3} = nitric acid concentration (kmole/m^3)

x_{HCl} = hydrochloric acid concentration (kmole/m^3)

If ΔZ is greater than 0, then the pH is estimated as

$$x_{OH^-} = \min\left(0.0001, \Delta Z\right)$$

$$x_{H^+} = \frac{K_{eq,H_2O}}{x_{OH^-}}$$

$$pH = -\log_{10}\left(x_{H^+}\right).$$

If ΔZ equals 0, then

$$x_{H^+} = 10^{-7}$$

$$pH = 7$$

If ΔZ is less than 0, then

$$x_{H^+} = \min\left(0.0001, \left|\Delta Z\right|\right)$$

$$pH = -\log_{10}\left(x_{H^+}\right)$$

The activities are then initialized. Activities are calculated using Davies modification of the Debye-Huckel equation [55],

$$\log_{10}\gamma(i) = -A\,Z(i)^2\left[\frac{\sqrt{I}}{1+\sqrt{I}} - b\ I\right] \qquad (2\text{-}175)$$

where

$\gamma(i)$ = activity coefficient for ion *i*

$Z(i)$ = absolute value of the charge on ion *i*

b = empirical constant = 0.2

I = ionic strength, defined as

$$I = \tfrac{1}{2}\sum_{all\ ions} C(j)Z(j)^2$$

$C(j)$ = concentration of the j^{th} ion in solution

A = $1.825\times10^6\,\rho_w^{1/2}/(\varepsilon\,T)^{3/2}$

ρ_w = density of water (g/cm^3)

ε = dielectric constant of water.

The initial strength is estimated using the initial buffer species along with the OH^- and H^+ concentrations

$$I = 0.5\left(3x_{NaP} + x_{Cs} + x_{HNO_3} + x_{HCl} + x_{OH^-} + x_{H^+}\right)$$

The equations to be solved are:

(1) the phosphate mole balance:

$$M(P) = [H_3PO_4] + [H_2PO_4^-] + [HPO_4^{2-}] + [PO_4^{3-}],$$

where $M(P)$ is the total kmoles of phosphate per m^3 of water and is user input,

(2) the borate mole balance:

$$M(B) = [B(OH)_4^-] + [B(OH)_3]$$

where M(B) is the input kmoles of borate per m^3 of water,

(3) the CO_2 hydrolysis balance:

$$M(C) = \left[HCO_3^-\right] + \left[CO_3^{2-}\right]$$

where *M(C)* is the moles of dissolved CO_2 per m^3 of water, and

(4) the charge balance:

$$\left[NO_3^-\right] + \left[Cl^-\right] + \left[OH^-\right] + \left[H_2PO_4^-\right] + 2\left[HPO_4^{2-}\right] + 3\left[PO_4^{3-}\right] + \left[B(OH)_4^-\right] + \left[HCO_3^-\right] + 2\left[CO_3^{2-}\right] = \left[H^+\right] + \left[A^+\right] + 3M(P)$$

where $[A^+]$ is the kmoles of alkali per m^3 added to adjust for boric acid and the kmoles/m^3 of phosphate added is assumed to be in the form Na_3PO_4. Also needed is the ionization constant for water, written in the form:

$$K_w = \left[H^+\right]\gamma(1)\left[OH^-\right]\gamma(1)$$

where the activity coefficients have been included. The ionization constant is determined from the formula

$$\log_{10} K_{w,p} = 2\left(7.2 + 2.5\rho_w\right)\log_{10} \rho_w - \frac{3108}{T} - 3.55$$

where

$K_{w,p}$ = ionization constant of water in units of (moles/kg)^2

ρ_w = density of water in g/cm^3.

To get the ionization constant in units of (kmole/m^3)^2, multiply by the density of water squared:

$$K_w = K_{w,p}\rho_w^2.$$

The concentrations of the derived species are determined from equilibrium constants:

$$k_4 = \frac{\left[H^+\right]\left[B(OH)_4^-\right]}{\left[B(OH)_3\right]}\gamma(1)^2$$

$$\log_{10} k_4 = \frac{1573}{T} + 28.8397 + 0.11748T - 13.2258\log_{10} T + \log_{10} K_w$$

$$k_5 = \frac{\left[HPO_4^{2-}\right]}{\left[H^+\right]\left[PO_4^{3-}\right]}\frac{\gamma(2)}{\gamma(1)\gamma(3)}$$

$$\log_{10} k_5 = \frac{-675}{T} + 1.793 + \log_{10} K_w$$

$$k_6 = \frac{\left[OH^-\right]\left[H_2PO_4^-\right]}{\left[HPO_4^{2-}\right]}\frac{\gamma(1)^2}{\gamma(2)}$$

$$\log_{10} k_6 = \frac{-17156.9}{T} - 37.7345 \ln T + 0.0322082\, T + \frac{8.97579 \times 10^5}{T^2} + 246.045$$

$$k_7 = \frac{\left[OH^-\right]\left[H_3PO_4\right]}{\left[H_2PO_4^-\right]}\gamma(n)$$

$$\log_{10} k_7 = \frac{-17655.8}{T} - 39.4277 \ln T + 0.0325405\, T + \frac{810134}{T^2} + 253.198$$

$$k_2 = \frac{\left[H^+\right]\left[HCO_3^-\right]}{\left[CO_{2aq}\right]}\gamma(1)^2$$

$$\log_{10} k_2 = \frac{2518}{T} - 0.7566 + \log_{10} K_w$$

$$k_3 = \frac{\left[H^+\right]\left[CO_3^{2-}\right]}{\left[HCO_3^-\right]}\gamma(2)$$

$$\log_{10} k_3 = \frac{2142}{T} - 3.523 + \log_{10} K_w$$

The iteration proceeds by

(1) setting the activities, and mass ratios of the acid and CO_2 total masses to the principal species,

(2) get new species concentrations from the ratios and mole balances,

(3) recalculate the strength and activities including all species in the charge balance, and

(4) use the charge balance to calculate the pH. This process is repeated until the pH converges to within 0.0001. The iteration is accelerated by using the gradient of the change in pH after the first 5 iterations.

2.12.7.3 Silver-Iodine Model

Silver in the pool can act to trap iodine. This is modeled in the iodine pool model by assuming a fixed fraction of the silver present in the pool (default set at 1 x 10^{-6}) is available to react with iodine, forming AgI sludge. The iodine thus reacted is assumed trapped and does not participate in the pool aqueous chemistry. The silver is assumed to be provided by RN class 12, and AgI is given its own RN class.

2.12.7.4 Iodine Aqueous Pool Chemistry

The aqueous iodine chemistry model is a semi-mechanistic model based primarily on the INSPECT equation set [56,57] plus work by Powers [58]. The model includes the effects of radiolysis, take-up of iodine by silver, metal ions (represented by iron), and acid-base buffers. Equations are included for organic iodine, represented as methyl iodine.

Chemical Reaction Equations

The chemical equations in the set are of the general form

$$A + B \Leftrightarrow C + D \tag{2-176}$$

with forward reaction rate k_f and reverse rate k_r. These are used to set up chemical reaction kinetic equations for each chemical species in the set. Using the above equation as an example, the reaction rate equation for species *C* would include terms from this equation plus perhaps a source from radiolysis:

$$\frac{d[C]}{dt} = k_f[A][B] - k_r[C][D] + \dot{S} \tag{2-177}$$

where the brackets [ ] indicate concentration of the species and the $\dot{S}$ is the source of *C* from radiolysis. The set of chemical reaction kinetics equations form a coupled set of nonlinear ordinary differential equations, which are solved using a standard stiff differential equation solver [59] to get the pool speciation. Initial conditions are set up by assuming some species, termed driver species, are given and constant over a calculational timestep. There are five driver species in the current equation set. These are aqueous O_2, H_2, CH_4, OH^-, and H^+. Some driver species are set by assuming equilibrium with the atmosphere via a Henry's law relationship; these are aqueous O_2, H_2, and CH_4. The OH^- and H^+ are set by determining the pool pH.

The initial total iodine concentration is specified at the beginning of the timestep as species I, and the iron ion concentration is specified as Fe^{3+}. Other species in the pool,

such as silver, nitric and hydrochloric acids, and phosphate and borate buffers, do not actually participate in the calculation of speciation other than to set the initial pH and iodine level (silver, by removing some iodine). Pool pH is determined either from an acid-base balance or is read in directly via user input.

The current chemical equation set consists of 276 equations as given in Table 2.8 through Table 2.12 and includes 39 species.

Table 2.8 Kinetic Equations for Water Radiolysis

Number	Reaction	Rate Constant*
M1	$OH^{O} + H_2(aq) \rightarrow H^{O} + H_2O$	4.2 x 10^7
M2	$OH^{O} + H_2O_2 \rightarrow HO_2^{O} + H_2O$	2.7 x 10^7
M3	$OH^{O} + O_2^{-} \rightarrow O_2(aq) + OH^{-}$	8 x 10^9
M4	$H^{O} + O_2(aq) \rightarrow HO_2^{O}$	1.2 x 10^{10}
M5	$H^{O} + O_2^{-} \rightarrow HO_2^{-}$	2 x 10^{10}
M6	$e^{-} + O_2(aq) \rightarrow O_2^{-}$	1.9 x 10^{10}
M7	$e^{-} + H_2O_2 \rightarrow OH^{O} + OH^{-}$	1.2 x 10^{10}
M8	$e^{-} + O_2^{-} + H_2O \rightarrow HO_2^{-} + OH^{-}$	1.3 x 10^{10}
M9	$e^{-} + H^{+} \rightarrow H^{O}$	2.3 x 10^{10}
M10	$e^{-} + H_2O \rightarrow H^{O} + OH^{-}$	19
M11	$e^{-} + HO_2^{-} \rightarrow O^{-} + OH^{-}$	3.5 x 10^9
M12	$OH^{O} + HO_2^{O} \rightarrow O_2(aq) + H_2O$	6 x 10^9
M13	$2\ OH^{O} \rightarrow H_2O_2$	5.5 x 10^9
M14	$H^{O} + HO_2^{O} \rightarrow H_2O_2$	2 x 10^{10}
M15	$H^{O} + H_2O_2 \rightarrow OH^{O} + H_2O$	5 x 10^7
M16	$OH^{-} + H^{O} \rightarrow e^{-} + H_2O$	2.5 x 10^7
M17	$HO_2^{O} + O_2^{-} \rightarrow O_2(aq) + HO_2^{-}$	9.7 x 10^7
M18	$2\ HO_2^{O} \rightarrow H_2O_2 + O_2(aq)$	2.35 x 10^6
M19	$H^{+} + O_2^{-} \rightarrow HO_2^{O}$	5 x 10^7
M20	$HO_2^{O} \rightarrow H^{+} + O_2^{-}$	(7.93 x 10^5)

Number	Reaction	Rate Constant*
M21	$H^+ + HO_2^- \rightarrow H_2O_2$	2 x 10^{10}
M22	$H_2O_2 \rightarrow H^+ + HO_2^-$	(0.0413)
M23	$OH^O + OH^- \rightarrow H_2O + O^-$	1.3 x 10^{10}
M24	$O^- + H_2O \rightarrow OH^O + OH^-$	(1.47 x 10^8)
M25	$HO_2^O + OH^- \rightarrow O_2^- + H_2O$	(1 x 10^9)
M26	$O_2^- + H_2O \rightarrow HO_2^O + OH^-$	(0.639)
M27	$H^O + OH^O \rightarrow H_2O$	7 x 10^9
M28	$2\,H^O \rightarrow H_2(aq)$	5 x 10^9
M29	$e^- + H^O + H_2O \rightarrow OH^- + H_2(aq)$	2.4 x 10^{10}
M30	$2e^- + 2H_2O \rightarrow 2OH^- + H_2(aq)$	5.5 x 10^9
M31	$e^- + OH^O \rightarrow OH^-$	3 x 10^{10}
M32	$O^- + O_2(aq) \rightarrow O_3^-$	3.5 x 10^9
M33	$H_2(aq) + O^- \rightarrow H^O + OH^-$	1 x 10^8
M34	$H_2O_2 + O^- \rightarrow H_2O + O_2^-$	5 x 10^8
M35	$OH^O + HO_2^- \rightarrow HO_2^O + OH^-$	7.5 x 10^9
M36	$HO_2^- + O^- \rightarrow OH^- + O_2^-$	8 x 10^8
M37	$O_3^- + H_2O_2 \rightarrow O_2^- + O_2(aq) + H_2O$	1.6 x 10^6
M38	$O_3^- + HO_2^- \rightarrow O_2^- + O_2(aq) + OH^-$	8.9 x 10^5
M40	$O_3^- + H_2 \rightarrow H^O + O_2(aq) + OH^-$	2.5 x 10^5
M102	$H_2O_2 \rightarrow 2\,OH^O$	2.33 x 10^{-7} (6.4x10^5 exp (-8540/T))

*Rate constants are in units m^3/kmole-s and s^{-1}. Most rate constants were taken from [56]. Rate constants within parentheses were estimated as part of this work.

Table 2.9 Reactions of Iodine

Number	Reaction	Rate Constant*
M53	$I^O + e^- \rightarrow I^-$	2.4 x 10^{10}
M54	$I_2(aq) + e^- \rightarrow I_2^-$	5.1 x 10^{10}
M55	$IO_3^- + e^- \rightarrow IO_3^{2-}$	7.8 x 10^9
M56	$I_2^- + e^- \rightarrow 2I^-$	1.3 x 10^{10}
M57	$I_3^- + e^- \rightarrow I^- + I_2^-$	3.5 x 10^{10}
M59	$IO_3^O + e^- \rightarrow IO_3^-$	13.5 x 10^{10}
M60	$IO_2^- + e^- \rightarrow IO^- + O^-$	11 x 10^{10}
M61	$IO^- + e^- \rightarrow I^- + O^-$	22.9 x 10^{10}
M62	$IO^O + e^- \rightarrow IO^-$	21 x 10^{10}
M63	$HOI^- + e^- \rightarrow I^- + OH^-$	1.9 x 10^{10}
M65	$I^O + H^O \rightarrow H^+ + I^-$	2.7 x 10^{10}
M66	$I_2(aq) + H^O \rightarrow I_2^- + H^+$	3.5 x 10^{10}
M67	$I_2^- + H^O \rightarrow 2I^- + H^+$	1.8 x 10^7
M68	$I_3^- + H^O \rightarrow I^- + I_2^- + H^+$	8 x 10^9
M69	$I^- + H^O \rightarrow HI^-$	5.3 x 10^6
M70	$HOI^- + H^O \rightarrow I^- + H_2O$	4.4 x 10^{10}
M72	$HOI^O + H^O \rightarrow I^O + H_2O$	1 x 10^9
M73	$IO_2^- + O_2^- + H_2O \rightarrow IO^O + O_2(aq) + 2OH^-$	1 x 10^7

1 Rate from Wren

2 Rate from Karasawa

Number	Reaction	Rate Constant*
M74	$I_2(aq) + O_2^- \rightarrow I_2^- + O_2(aq)$	21 x 10^9
M75	$HOI^O + O_2^- \rightarrow I^O + O_2(aq) + OH^-$	1 x 10^6
M76	$I_3^- + O_2^- \rightarrow I_2^- + I^- + O_2(aq)$	22.5 x 10^8
M77	$I_2^- + O_2^- \rightarrow 2I^- + O_2(aq)$	27.5 x 10^9
M78	$IO^O + O_2^- \rightarrow IO^- + O_2(aq)$	8 x 10^7
M79	$IO_3^- + O_2^- \rightarrow IO_3^{2-} + O_2(aq)$	8 x 10^9
M80	$I_2^- + HO_2^O \rightarrow I_2(aq) + HO_2^-$	1 x 10^{10}
M81	$HOI^O + HO_2^O \rightarrow I^O + O_2(aq) + H_2O$	1 x 10^5
M82	$IO_2^- + HO_2^O \rightarrow IO^O + O_2(aq) + OH^-$	1 x 10^7
M83	$I_2(aq) + HO_2^O \rightarrow I_2^- + O_2(aq) + H^+$	1.8 x 10^7
M84	$I^O + OH^O \rightarrow HOI^O$	1.6 x 10^{10}
M85	$HOI^O + OH^O \rightarrow IO^O + H_2O$	7 x 10^9
M86	$IO^O + OH^O \rightarrow HIO_2^O$	11 x 10^{10}
M87	$IO_2^O + OH^O \rightarrow IO_3^- + H^+$	11 x 10^{10}
M88	$I^- + OH^O \rightarrow HOI^-$	1.8 x 10^{10}
M89	$I_2(aq) + OH^O \rightarrow HOI^O + I^O$	1.1 x 10^{10}
M90	$IO_3^- + OH^O \rightarrow IO_3^O + OH^-$	21 x 10^6
M91	$HOI^- + OH^O \rightarrow HOI^O + OH^-$	2.7 x 10^{10}
M92	$I_2^- + OH^O \rightarrow I_2(aq) + OH^-$	3.8 x 10^{10}
M93	$I_3^- + OH^O \rightarrow I_2(aq) + I^O + OH^-$	2 x 10^{10}
M94	$I^- + O^- + H_2O \rightarrow I^O + 2OH^-$	24.7 x 10^7

Number	Reaction	Rate Constant*
M95	$IO^- + O^- + H_2O \rightarrow IO^O + 2OH^-$	21.1 x 10^8
M96	$IO_3^- + O^- + H_2O \rightarrow IO_3^O + 2OH^-$	15.23 x 10^6
M97	$I^- + H_2O_2 \rightarrow IO^- + H_2O$	10.014
M100	$IO_3^O + H_2O_2 \rightarrow IO_3^- + HO_2^O + H^+$	1 x 10^9
M101	$I^O + H_2O_2 \rightarrow I^- + HO_2^O + H^+$	3000
M103	$HIO_2^O \rightarrow IO_2^- + H^+$	21 x 10^{10}
M104	$I^O + I^- \rightarrow I_2^-$	21.1 x 10^{10}
M105	$HOI^- \rightarrow I^- + OH^O$	2.25 x 10^6
M106	$I^- + HOI^- \rightarrow I_2^- + OH^-$	2.5 x 10^4
M107	$HOI^- \rightarrow I^O + OH^-$	1.2 x 10^8
M108	$2I_2^- \rightarrow I_3^- + I^-$	4.5 x 10^9
M109	$I_2^- + O_2(aq) + H^+ \rightarrow HO_2^O + I_2(aq)$	6 x 10^5
M110	$I_2^- + I^O \rightarrow I_3^-$	4.5 x 10^9
M111	$2I^O \rightarrow I_2(aq)$	1 x 10^{10}
M112	$I_2^- + HOI^O \rightarrow IO^O + 2I^- + H^+$	1 x 10^5
M113	$IO_3^{2-} + H_2O \rightarrow HIO_3^- + OH^-$	1 x 10^8
M114	$IO_3^O + I^- \rightarrow IO_2^- + IO^O$	11 x 10^6
M115	$IO_2^O + I^- \rightarrow I_2(aq) + O_2^-$	1 x 10^{10}
M116	$HI^- + H_2O \rightarrow I^O + H_2(aq) + OH^-$	1000
M117	$HI^- + H^+ \rightarrow I^O + H_2(aq)$	11 x 10^{10}
M118	$I_2^- + HOI^- \rightarrow I_3^- + OH^-$	1.8 x 10^{10}
M119	$HOI^- + I^O \rightarrow I_2(aq) + OH^-$	2.3 x 10^{10}

Number	Reaction	Rate Constant*
M120	$2HOI^- \rightarrow I_2(aq) + 2OH^-$	2 x 10^{10}
M121	$HOI^O + e^- \rightarrow HOI^-$	2 x 10^{10}
M122	$HOI^O + O_2^- \rightarrow HOI^- + O_2(aq)$	11 x 10^9
M123	$2IO^O \rightarrow I_2O_2$	1.5 x 10^9
M124	$I_2O_2 + H_2O \rightarrow IO_2^- + HOI^O + H^+$	11000
M125	$IO^O + IO_2^- \rightarrow IO_2^O + IO^-$	11 x 10^{10}
M126	$IO_2^O + H_2O \rightarrow HIO_3^- + H^+$	11000
M127	$2HIO_3^- \rightarrow IO_3^- + IO_2^- + H_2O$	25.2 x 10^9
M128	$IO_3^O + HIO_3^- \rightarrow 2IO_3^- + H^+$	11 x 10^{10}
M129	$IO_3^O + IO_3^{2-} \rightarrow 2IO_3^-$	11 x 10^{10}
M130	$IO_2^- + H_2O_2 \rightarrow IO^- + O_2(aq) + H_2O$	1 x 10^8
M131	$I_2(aq) + H_2O \rightarrow I_2OH^- + H^+$	*
M132	$I_2OH^- + H^+ \rightarrow I_2(aq) + H_2O$	1 x10^{10}
M133	$I_2(aq) + OH^- \rightarrow I_2OH^-$	1 x 10^{10}
M134	$I_2OH^- \rightarrow I_2(aq) + OH^-$	13 x 10^5
M135	$I_2OH^- \rightarrow I^- + HOI^O$	1963
M136	$I^- + HOI^O \rightarrow I_2OH^-$	1 x 10^6
M137	$IO^- + I_2OH^- \rightarrow IO_2^- + 2I^- + H^+$	6
M138	$IO_2^- + I_2OH^- \rightarrow IO_3^- + 2I^- + H^+$	26
M139	$2HOI^O \rightarrow IO_2^- + I^- + 2H^+$	16.7
M141	$HOI^O + IO_2^- \rightarrow IO_3^- + I^- + H^+$	1 x 10^7

Number	Reaction	Rate Constant*
M144	$HOI^{O} + OH^{-} \rightarrow IO^{-} + H_2O$	1 x 10^{9}
M145	$IO^{-} + H_2O \rightarrow HOI^{O} + OH^{-}$	2750
M146	$2I^{-} + 1/2O_2(aq) + 2H^{+} \rightarrow I_2(aq) + H_2O$	347
M147	$I_2(aq) + H_2O \rightarrow 2I^{-} + 1/2O_2(aq) + 2H^{+}$	1 x 10^{-10}
M148	$I_2(aq) + I^{-} \rightarrow I_3^{-}$	
M149	$I_3^{-} \rightarrow I_2(aq) + I^{-}$	*
M150	$I^{-} + IO_2^{-} + 2H^{+} \rightarrow 2HOI^{0}$	10^{12} x R139
M151	$IO_3^{-} + 2I^{-} + 2H^{+} \rightarrow H_2I_3O_3^{-}$	6.72 x 10^{8}
M152	$H_2I_3O_3^{-} + 3I^{-} + 4H^{+} \rightarrow 3I_2(aq) + 3H_2O$	1 x 10^{10}
M153	$I_2^{-} \rightarrow I^{0} + I^{-}$	1.1 x 10^{5}
M154	$HOOI + I^{-} \rightarrow I_2(aq) + HO_2^{-}$	4.5 x 10^{5}
M155	$I_2(aq) + HO_2^{-} \rightarrow HOOI + I^{-}$	R154 / 0.04119
M156	$HOI^{0} + HO_2^{0} \rightarrow HOOI + OH^{0}$	2.1 x 10^{9}
M157	$HOOI + OH^{-} \rightarrow I^{-} + O_2 + H_2O$	2 x 10^{9}
M158	$HOI^{0} + H_2O_2 \rightarrow HOOI + H_2O$	37
M159	$HOOI \rightarrow I^{-} + O_2 + H^{+}$	0.2

Number	Reaction	Rate Constant*
Footnotes to table: *INSPECT selects a rate constant based on the equilibrium constant and the rate constant for the back reaction. **See Powers [58]. $\frac{d[I_2(aq)]}{dt} = -\frac{1}{2}\frac{d[I^-]}{dt} = -\frac{1}{2}\frac{d[H^+]}{dt} = -2\frac{d[O_2(aq)]}{dt} = 34[I^-]^2[H^+][O_2(aq)] + \frac{7.14\times10^4[I^-][B(OH)_3][H^+][O_2(aq)]}{(1+1.47\times10^8[H^+])}$ ***See Powers [58]. $\frac{d[IO_3^-]}{dt} = -\frac{d[I^-]}{dt} = -\frac{2}{3}\frac{d[O_2(aq)]}{dt} = 3.2\times10^{-5}[I^-][O_2(aq)]$		

Table 2.10 Reactions of Ferrous and Ferric Ions

Number	Reaction	Rate Constant
M259	$Fe^{2+} + O_2(aq) \rightarrow Fe^{3+} + O_2^-$	$\frac{d[Fe^{2+}]}{dt} = k[[Fe^{2+}][O_2(aq)][OH^-]^2$ $k = 3.10^{21}\exp(-3557/T)$
M260	$Fe^{3+} + H_2O_2 \rightarrow Fe^{2+} + HO_2^O + H^+$	2 x 10^{-3}/(1 + x/[H^+])
M261	$Fe^{3+} + HO_2^- \rightarrow Fe^{2+} + HO_2^O$	1.1 x 10^{24} exp(-14090/T)/(1 + x/[H^+])
M262	$Fe^{3+} + H_2^O \rightarrow Fe^{2+} + O_2(aq) + H^+$	3 x 10^5
M263	$Fe^{3+} + O_2^- \rightarrow Fe^{2+} + O_2(aq)$	5040 exp(3294/T)
M264	$Fe^{3+} + e^- \rightarrow Fe^{2+}$	2.3 x 10^{10}
M265	$Fe^{3+} + H^O \rightarrow Fe^{2+} + H^+$	9.6 x 10^7
M266	$Fe^{2+} + H^O + H^+ \rightarrow Fe^{3+} + H_2$	7.5 x 10^6
M267	$Fe^{3+} + OH^O + H_2O \rightarrow Fe^{2+} + H_2O_2 + H^+$	1.5/(1 + y/[H^+]) Ref. 184

Number	Reaction	Rate Constant
M268	$Fe^{2+} + O^- + H_2O \rightarrow Fe^{3+} + 2OH^-$	3.8 x 10^9
M269	$Fe^{2+} + OH^O \rightarrow Fe^{3+} + OH^-$	3 x 10^8
M270	$Fe^{2+} + H_2O_2 \rightarrow Fe^{3+} + OH^- + OH^O$	77
M271	$Fe^{2+} + HO_2^O \rightarrow Fe^{3+} + HO_2^-$	3 x 10^7
M272	$Fe^{2+} + HO_2^- \rightarrow Fe^{3+} + OH^- + O^-$	770
M273	$Fe^{2+} + O_2^- \rightarrow Fe^{3+} + 2O^-$	7.2 x 10^6
M274	$Fe^{3+} + H_2O_2 \rightarrow Fe^{2+} + O_2^- + 2H^+$	2 x 10^{-3} x /([H^+] + x)
M275	$Fe^{3+} + HO_2^- \rightarrow Fe^{2+} + O_2^- + H^+$	1.1 x 10^{24} exp(-14090/T) x /([H^+] + x)
M276	$Fe^{3+} + OH^O + H_2O \rightarrow Fe^{2+} + HO_2^- + 2H^+$	1.5 y/([H^+] + y) Ref. 184
x = equilibrium constant for $HO_2^O \rightarrow H^+ + O_2^- = 10^{-1431/T}$ y = equilibrium constant for $H_2O_2 \rightarrow H^+ + HO_2^- = 10^{-3484/T}$		

Table 2.11 Organic Reactions

Number	Reaction	Rate Constant
M377	$CH_4(aq) + OH^O \rightarrow CH_3 + H_2O$	1.21 x 10^8
M378	$CH_4(aq) + I_2(aq) \rightarrow CH_3I + I^- + H^+$	4
M379	$CH_4(aq) + IO^- \rightarrow CH_3I + I^- + OH^-$	1 x 10^8
M380	$CH_4(aq) + HOI^O \rightarrow CH_3I + H_2O$	1 x 10^8
M381	$2\,CH_3 \rightarrow$ products	1.24 x 10^9
M382	$CH_3 + OH^O \rightarrow CH_3OH$	1 x 10^8
M383	$CH_3 + H_2O_2 \rightarrow CH_3OH + OH^O$	3.5 x 10^7
M384	$CH_3 + O_2(aq) \rightarrow$ products	4.9 x 10^9

Number	Reaction	Rate Constant
M385	$CH_3 + HOI^O \rightarrow CH_3I(aq) + OH^O$	1 x 10^9
M386	$CH_3 + e^- + H_2O \rightarrow CH_4 + OH^-$	1 x 10^{10}
M387	$CH_3 + H^O \rightarrow CH_4(aq)$	1 x10^{10}
M388	$CH_3 + I^O \rightarrow CH_3I(aq)$	1 x 10^{10}
M389	$CH_3 + IO_3^O \rightarrow CH_2O + HIO_2^O$	1 x 10^8
M390	$CH_3 + IO_2^O \rightarrow CH_2O + HOI^O$	1 x 10^8
M391	$CH_3 + IO^O \rightarrow CH_3OH + HOI^O$	1 x 10^8
M392	$CH_3 + HOI^O \rightarrow I^O + CH_3OH$	1 x 10^8
M393	$CH_3 + I_2(aq) \rightarrow CH_3I(aq) + I^O$	6 x 10^9
M394	$CH_3 + HIO_3^- \rightarrow CH_2O + HOI^O + OH^-$	1 x 10^8
M395	$CH_3I(aq) + OH^O \rightarrow CH_3OH + I^O$	1 x 10^8
M396	$CH_3I + H_2O \rightarrow CH_3OH + H^+ + I^-$ $\log_{10}k = 93.14585 - 9661.274/T - 24.42937 \log_{10}T$	
M397	$CH_3I + OH^- \rightarrow CH_3OH + I^-$	6.5 x 10^{-5}
M398	$CH_3I(aq) + e^- \rightarrow CH_3 + I^-$	1.6 x 10^{10}
M399	$CH_3I(aq) + H^O \rightarrow CH_3 + H^+ + I^-$	1 x 10^{10}
M400	$CH_3I(aq) + OH^O \rightarrow CH_4OI$	1.4 x 10^9
M401	$CH_4OI \rightarrow CH_3I^+ + OH^-$	3.1
M402	$CH_4OI + I^- \rightarrow I_2^- + CH_3OH$	2 x 10^9
M403	$CH_3I^+ + OH^- \rightarrow CH_4OI$	1 x 10^{10}
M404	$CH_3I^+ + I^- + H_2O \rightarrow CH_3OH + I_2^- + H^+$	7.7 x 10^9
M405	$CH_2O + OH^O \rightarrow HCO + H_2O$	1 x 10^9
M406	$Fe^{3+} + 2I^- \rightarrow Fe^{2+} + I_2^-$	21
M407	$CH_3OH + OH^O \rightarrow CH_3O + H_2O$	1 x 10^9
M408	$2CH_3O \rightarrow product$	1 x 10^9
M409	$Fe^{3+} + CH_3O \rightarrow Fe^{2+} + product$	1 x 10^9

Number	Reaction	Rate Constant
M410	$H^+ + CH_3O + Fe^{2+} \rightarrow Fe^{3+} + CH_3OH$	1 x 10^9
M411	$CH_3OH + H^O \rightarrow CH_3O + H_2$	5 x 10^8
M412	$CH_3OH + e^- \rightarrow H^O + CH_3O$	1 x 10^4
M413	$CH_3 + CH_3OH \rightarrow CH_4(aq) + CH_3O$	2 x 10^6
M414	$CH_3O + H_2O_2 \rightarrow CH_2O + OH^O + H_2O$	4 x 10^4
M415	$CH_3O + O_2 \rightarrow$ products	4.2 x 10^9
M416	$CH_2O + e^- + H_2O \rightarrow CH_3O + OH^-$	1 x 10^7
M417	$CH_2O + H^O \rightarrow H_2 + HCO$	5 x 10^6
M418	$CH_2O + CH_3 \rightarrow CH_4(aq) + HCO$	5 x 10^6
M419	$CH_2O + O^- \rightarrow OH^- + HCO$	1 x 10^9

Table 2.8 is the basic water hydrolysis set from INSPECT [56]. Table 2.9 is the iodine reaction set from INSPECT [56,57] and Powers [58]; Table 2.10 is the iron reaction set [58]. Table 2.11 is the organic iodine set [58]. The framework for the organic reactions is in place, but the equations have not been entered, due to a lack of data to compare results. When data become available, the organic reactions can be activated by entering the equations into the EQINIT routine. The numbers for the reactions in the first column of the tables corresponds to the reactions as labeled in Powers [58]. The column labeled "Rate Constant" in the tables gives either a constant rate or refers to a calculated rate as given in Table 2.12.

Table 2.12 Variable Rates

M10	$e^- + H_2O \rightarrow H^0 + OH^-$	R16*K_{H2O}/K_{H0}
M20	$HO_2^0 \rightarrow H^+ + O_2^-$	R19*K_{HO2}
M22	$H_2O_2 \rightarrow H^+ + HO_2^-$	R21*K_{H2O2}
M24	$O^- + H_2O \rightarrow OH^0 + OH^-$	R23*K_{H2O}/K_{OH0}

M26	$O_2^- + H_2O \rightarrow HO_2^0 + OH^-$	R25*R19*K_{H2O}/R20
M102	$H_2O_2 \rightarrow 2OH^0$	6.4x10^5 exp(-8540/T)
M134	$I_2OH^- \rightarrow I_2(aq) + OH^-$	R132*R133*K_{H2O}/R131
M145	$IO^- + H_2O \rightarrow HOI^0 + OH^-$	R144*R152*K_{H2O}/R151
M149	$I_3^- \rightarrow I_2(aq) + I^-$	R148/K_3

The K_n in the third column of Table 2.12 are equilibrium constants, and the R_n are reaction rates for equation number n. Also needed are the acid dissociation constants (Table 2.13).

Table 2.13 Acid Dissociation Constants

$OH^0 \Leftrightarrow O^- + H^+$	$\log_{10} K_{OH0}$ = -4893.6/T + 60.701 – 22.629 $\log_{10}$ T
$H_2O_2 \Leftrightarrow HO_2^- + H^+$	$\log_{10} K_{H2O2}$ = -3789.7/T + 56.284 – 16.473 $\log_{10}$ T
$HO_2^0 \Leftrightarrow O_2^- + H^+$	$\log_{10} K_{HO2}$ = -519/T – 3.06
$H^0 + OH^- \Leftrightarrow e^- + H_2O$	$\log_{10} K_{H0}$ = -2317/T – 1.816
$I_3^- \Leftrightarrow I_2 + I^-$	$\log_{10} K_{I3-}$ = 945.5/T – 0.282
$HOI \Leftrightarrow IO^- + H^+$	$\log_{10} K_{HOI}$ = -80670/T + 0.7335 T +2800 – 1115.1 $\log_{10}$ T

The solution of the equation set proceeds as follows (box 10 in Figure 2.7):

(1) If this is the initial calculation of speciation (indicated by all species being zero other than the drivers), an initialization is performed to set the initial speciation. At present, this consists of solving for the iodine ion concentration from a set of five equations; these can be reduced to a cubic equation in I^-, which is then solved for directly. The other species in the five equations (I_{2aq}, HOI, IO_3^-, and I_3^-) could also be initialized, but this does not seem to be necessary. In actuality, iodine ion is approximately equal to the total iodine concentration over most of the pH range and only differs at low pHs.

(2) If the pool speciation calculation has been done previously (on the last timestep), the speciation from the last timestep is used as the initial speciation.

(3) The set of chemical reaction equations is solved via a stiff ODE solver [59]. As implemented in MELCOR, the equations are advanced in “time” using a default “timestep” of 2.0s until equilibrium is reached, indicated by the changes in the species concentrations being less than an error criterion, or 2000 steps are taken. This result is then taken as the pool speciation. This equilibrium approach is used, rather than advancing the equations in real time, because of the uncertainty in the actual time history of the pool. That is, the pool initial conditions are set to a simplified starting point when the pool model becomes activated. This initial starting point does not necessarily reflect the actual pool speciation at pool model activation time, and it is unknown to the pool model how long the pool has actually been in existence. Therefore, the time advancement of the pool equations is treated as an advancement in iteration time to equilibrium, rather than advancement in real time. The “iteration timestep”, the number of steps, and the convergence criteria are adjustable via sensitivity coefficients 7181.

Aqueous Radiolysis

The radiolysis model for the pool uses a set of temperature-dependent yields based on values recommended by Buxton et al. [60] at 298 K and Elliot et al. [61] at 573 K, as listed in Table 2.14.

Table 2.14 Primary Products of Water Radiolysis

Species	G (molecules/100 ev)
$e^- = H^+$	0.9204 + 5.364 T/1000
H	0.0798 + 1.7454 T/1000
OH^0	1.3238 + 4.6182 T/1000
H_2	0.2658 + 0.6182 T/1000
H_2O_2	0.1040 + 2.000 T/1000

This set of yields is used with the user-specified pool dose to calculate the radiolysis source terms for the aqueous chemistry reaction set, as

$$\dot{S}_i = G(i)\,2.88\times10^{-7}\,\dot{D}_{pool} \qquad (2\text{-}178)$$

where

$\dot{S}_i$ = radiolysis source for species *i* in pool (kmole/m^3-s)

$G(i)$ = yield factor for species *i*

$\dot{D}_{pool}$ = dose rate to pool (MRad/hr)

Speciation Initialization

The initialization of the pool species is done by combining a set of five iodine equations to eliminate all but the I^- concentration. This gives a cubic equation in the I^- concentration, which can be solved directly. The equation set does not include the effects of H_2O_2 on iodine, so is not a particularly good guess at high pHs.

2.12.7.5 Pool-Atmosphere Mass Transfer

Once the pool speciation is determined by the aqueous chemistry model, the mass exchange of iodine and methyl iodine with the atmosphere is calculated (see Figure 2.7). This is done via a two-film model, in which the concentration of iodine species in the pool at the pool surface is assumed to be in equilibrium via a partition coefficient with the species in the atmosphere in a film next to the pool surface at local saturation conditions.

Mass transfer is then done between this surface film and the bulk atmosphere based on the surface-bulk species concentration difference and a mass transfer coefficient. Transfer rates between the bulk pool and pool surface are ignored (the pool is assumed to be well-stirred). Partition coefficients are included for I_2, CH_3I, I^0, and HOI. The mass transfer equation for iodine is written as

$$\frac{d[I_{2atm}]}{dt} = k_{pool}\frac{A_{pool}}{V_{atm}}\left([I_{2aq}] \,/\, PC_{I2} - [I_{2atm}]\right) \qquad (2\text{-}179)$$

where

$[I_{2atm}]$ = atmospheric iodine concentration (kmole/m^3)

$[I_{2aq}]$ = bulk pool iodine concentration (kmole/m^3)

k_{pool} = mass transfer coefficient from pool surface to atmosphere (m/s)

PC_{I2} = partition coefficient for iodine.

The above equation can be written several ways, so care must be taken when comparing between codes.

The partition coefficient is defined as PC_I = (concentration of species *i* in aqueous phase) / (concentration of species *i* in gas-phase). The most important species released from the pool to the atmosphere is molecular iodine. The partition coefficient for iodine used in MELCOR is given as [62]

$$log_{10} P\,C_{I2} = 13.5467 - 0.0605142T + 7.166 \times 10^{-5}T^2 \qquad (2\text{-}180)$$

where T is in K. The partition coefficients for I^o and HOI in MELCOR are both the same and are given as

$$PC_{I0} = 0.0238\ PC_{I2} \qquad (2\text{-}181)$$

This is derived by taking the ratio of the *PC* for I^0 (1.9) [57] and that for I_2 at room temperature and pressure, and assuming the same temperature dependence for I^0 as for I_2. The partition coefficients for I^0 and HOI should be used with caution, as there is little proof for the contention that either can be released from the pool. Although a number of researchers have suggested partition coefficients for HOI, researchers have failed to measure its presence [63], and the partition coefficient for HOI should be regarded as a placeholder. Likewise, release of atomic iodine is controversial. These two *PC*s are defaulted to **OFF** in the iodine model but can be turned on via user input. The *PC* for methyl iodine is [64]

$$PC_{CI} = 9.4 \times 10^{-4}\, exp\left(\frac{2641}{T}\right) \qquad (2\text{-}182)$$

2.12.7.6 Iodine Atmospheric Radiolysis and Recombination

The atmospheric radiolysis model considers homogeneous radiolytic decomposition of iodine species, and subsequent recombination reactions.

The atmospheric reduction of iodine is represented by reactions with hydrogen and ozone, and radiolytic reduction. The thermal reduction reaction with hydrogen is

$$\frac{d[I_{2atm}]}{dt} = -k_{TIH}[I_{2atm}][H_2] \quad (2\text{-}183)$$

where the reaction rate is [65]

$$k_{TIH} = 1 \times 10^{11}\, exp\left(-\frac{20131}{T}\right)$$

and,

k_{TIH} = reaction coefficient with hydrogen (m^3/kmol-s)

T = atmospheric temperature (K).

The reaction with ozone is

$$\frac{d[I_{2atm}]}{dt} = -k_{TIO}[I_{2atm}][O_3] \quad (2\text{-}184)$$

where the reaction coefficient is [66]

$$k_{TIO} = 2.42 \times 10^{6}\, exp\left(-\frac{2050}{T}\right).$$

The radiolytic reduction effect is given as [67]

$$k_{RI} = 0.028\ \dot{D}_{atm} \quad (2\text{-}185)$$

where

k_{RI} = radiolytic reduction coefficient, and

$\dot{D}_{atm}$ = atmospheric dose rate (Mrad/hr).

The organic iodine is similarly reduced using an oxidation and a radiolytic reaction [68]. The oxidation reaction is

$$\frac{d[CH_3I_{atm}]}{dt} = -k_{TCIO}[CH_3I_{atm}][O_2] \quad (2\text{-}186)$$

where

$[CH_3I_{atm}]$ = atmospheric methyl iodide concentration (kmole/m^3),

k_{TCIO} = oxidation reaction rate, given as

$$k_{TCIO} = 10^9 \exp\left(-\frac{13235}{T}\right).$$

The radiolytic reduction rate is

$$K_{RCL} = 0.00164\ \dot{D}_{atm} \quad (2\text{-}187)$$

where, k_{RCI} is the radiolytic reduction coefficient for CH_3I. The effect of the decomposition is to increase the amount of elemental iodine in the atmosphere, decreasing the amount of I_2 and CH_3I.

The recombination reaction is assumed to be in equilibrium, using the new concentrations of I_2 and I^0. An equilibrium coefficient,

$$K_{I2I} = \frac{p(I_2)}{p(I)^2} \quad (2\text{-}188)$$

$$K_{I2I} = 5.44 \times 10^{-6}\, exp\left(\frac{18163}{T}\right) \quad (2\text{-}189)$$

and a mole balance on the iodine in the atmosphere as I_2 and I^0

$$M(I) = 2[I_2] + [I^0] \quad (2\text{-}190)$$

where *M(I)* is the molar concentration of I, is used to calculate the recombination of elemental iodine into I_2. Combining the equilibrium coefficient K_{I2I} with the mole balance gives the new concentration of I^0 as

$$\left[I^0\right] = \frac{1}{4K_{I2I}RT}\left[-1+\sqrt{1+8K_{I2I}M(I)RT}\right] \quad (2\text{-}191)$$

.

Since I^0 is not tracked in MELCOR, the I^0 is added to the I_2 for purposes of transport. The net effect of the decomposition-recombination reactions is to deplete CH_3I from the atmosphere and form I_2.

where

$[I_{2wall}]$ = wall surface iodine concentration (kmole/m^2)

k_{ad} = adsorption coefficient (m/s)

k_{de} = desorption coefficient (s^{-1}).

Default values for steel walls were selected to match results of RTF tests [69]. The coefficients are adjustable via sensitivity coefficients 7180. If the dry wall surfaces subsequently become wet, the water film is assumed to completely dissolve the adsorbed iodine and the film can drain to other surfaces or the pool via the MELCOR film model. The same model is used for steel or painted surfaces, although there is some evidence for a second stage chemical reaction process on painted adsorbing surfaces. There are not enough data presently available to determine the terms for such a model, so the physical model is used by itself.

2.12.7.7 Iodine Atmosphere-Wall Deposition

Iodine species in the atmosphere are allowed to deposit on dry wall surfaces via a physical adsorption-desorption model similar to the one in LIRIC [69]. The model is given as

$$\frac{d[I_{2wall}]}{dt} = k_{ad}[I_{2atm}] - k_{de}[I_{2wall}] \qquad (2\text{-}192)$$

where

$[I_{2wall}]$ = wall surface iodine concentration (kmole/m^2)

k_{ad} = adsorption coefficient (m/s), and

k_{de} = desorption coefficient (s^{-1}).

Default values for steel walls were selected to match results of RTF tests [69]. The coefficients are adjustable via sensitivity coefficients 7180. If the dry wall surfaces subsequently become wet, the water film is assumed to completely dissolve the adsorbed iodine and the film can drain to other surfaces or the pool via the MELCOR film model. The same model is used for steel or painted surfaces, although there is some evidence for a second stage chemical reaction process on painted adsorbing surfaces. There are not enough data presently available to determine the terms for such a model, so the physical model is used by itself.

2.12.8 Data Base Supporting Model Validation

There are three series of experiments that can be used for validating these models, the wide ranging Radioiodine Test Facility (RTF) experiments that are part of the Advanced

Containment Experiments (ACE) performed at (AECL) Whiteshell Nuclear Research Establishment [70], small scale radiolysis tests performed at (CEA) Cadarache [71], and the hydrolysis experiments performed at Oak Ridge National Laboratory (ORNL) [54]. In the RTF experiments, tests 2 and 3 varied the pH over a wide range and measured the iodine partition coefficient, that is, the ratio of the aqueous iodine to the airborne iodine concentrations. Qualitatively, they were able to show that as the pH increases, the partition coefficient increases, and the atmospheric iodine concentration decreases. In the CEA tests, a solution of iodine was exposed to a 0.4 MR/hr source and the iodine speciation was measured. The present MELCOR iodine model was used in a recent participation in International Standard Problem (ISP) 41.

In the ORNL experiments, the temperature and pH of a pool was varied from 25 to 90 degrees Centigrade and from 3 to 9, respectively. In these tests, the end product iodine speciation was measured.

Development and testing of the model initial testing was based on comparison with the results from other codes, in particular radiolysis results from the INSPECT code. Testing against ISP41 results validated the iodine pool-atmosphere partitioning variation with pH and the coefficients for the wall deposition on steel walls [72]. As mentioned, sufficient data to vallidate the organic reaction set are not yet available, so the organic reactions are not implemented, although all the framework is present.

In later testing, the model is compared with the available experimental data discussed previously, that is, the Canadian, French, and Oak Ridge data for validation. Finally, the effect of iodine chemistry on a late-phase accident is evaluated.

3. Discussion and Development Plans

3.1 RCS Deposition

The MELCOR Peer Review also placed the omission of some aerosol deposition processes, principally inertial impaction and turbulent deposition, on the list of the most important missing models in MELCOR. These processes, which are not generally important in containment and therefore are not included in MAEROS, may assume primary importance in the reactor coolant system. As discussed in the MELCOR Peer Review, experimental data and calculations using more comprehensive aerosol deposition models indicate that the neglect of these processes may result in a significant underestimate of the retention of aerosols in the primary system, especially for low-pressure sequences in which gas velocities are high. However, the Marviken assessment calculations [73] showed good agreement with primary system retention data for both aerosols and fission product vapors, indicating the possibility of compensating processes.

3.2 Chemical Reactions with Surfaces

The MELCOR Peer Review also identified the lack of explicit modeling in MELCOR for chemical reactions between deposited fission products and structures in the primary system as one of the most important missing models. Such reactions can greatly affect deposition (chemisorption) and revaporization rates. Although a framework exists in the RN package for allowing user specification of chemical reactions, it is largely untested and unused. Because user input is basically unconstrained, the generation of errors through unexpected reactions is quite possible. The MELCOR Peer Review noted that the lack of explicit modeling applies to all accident sequences and is particularly serious for cesium hydroxide and tellurium compounds. This has been addressed in release 1.8.4 via the surface chemisorption model.

3.3 Aqueous Chemistry

The MELCOR Peer Review separately identified fission product chemistry in water pools as a less critical but still important modeling omission. The chief concern expressed in the MELCOR Peer Review was that release of iodine to the environment may be understated because MELCOR neglects processes that can occur in water pools to transform cesium iodide into more volatile forms of iodine (e.g., reaction with methane to form methyl iodide). The MELCOR 1.8.5 code release includes a detailed iodine pool chemistry model, based largely on the INSPECT code and on work by Powers. The model has received limited testing and verification against the ISP-41 test data [72]. Future assessment against other experimental data is recommended in order to further evaluate and refine other important aspects of iodine chemistry including organic compounds and silver.

Appendix A: RN Package Sensitivity Coefficients

This appendix gives the sensitivity coefficients associated with various correlations and modeling parameters described in this reference manual.

Equation or Section	Coefficient	Value	Units
Section 2.4.2.3	C7000(1)	1.0E-18	-
	C7000(2)	0.001	-
	C7000(3)	0.1	-
	C7000(4)	0.1	-
	C7000(5)	1.0E-12	kg/m^3

Aerosol Coefficient Criteria			
	C7001(1)	1.0E-18	-
	C7001(2)	1.0E-6	-

Fission Product Decay Beta Range			
Section 2.6	C7002(1)	1.2	Kg/m^2

COR Material Release Multipliers			
Section 2.3.1	C7100(1)	1.0	-
	C7100(2)	0.0	-
	C7100(3)	0.0	-
	C7100(4)	0.0	-
	C7100(5)	0.0	-
	C7100(6)	0.0	-
	C7100(7)	0.0	-

CORSOR Coefficients – Section 2.3.1			
All classes, except for TE	C7101(1,1)	900.0	K
	C7101(2,1)	1400.0	K
	C7101(3,1)	2200.0	K
TE	C7101(1,1)	900.0	K
	C7101(2,1)	1600.0	K
	C7101(3,1)	2000.0	K
	C7101(1,2)	1.62E-11	min^{-1}
	C7101(2,2)	9.04E-8	min^{-1}
	C7101(3,2)	6.02E-6	min^{-1}
	C7101(1,3)	0.0106	$°C^{-1}$
	C7101(2,3)	0.00552	$°C^{-1}$
	C7101(3,3)	0.00312	$°C^{-1}$
XE	C7101(1,2)	7.02E-9	min^{-1}
	C7101(2,2)	2.02E-7	min^{-1}
	C7101(3,2)	1.74E-5	min^{-1}
	C7101(1,3)	0.00886	$°C^{-1}$
	C7101(2,3)	0.00667	$°C^{-1}$
	C7101(3,3)	0.00460	$°C^{-1}$
CS	C7101(1,2)	7.53E-12	min^{-1}
	C7101(2,2)	2.02E-7	min^{-1}
	C7101(3,2)	1.74E-5	min^{-1}
	C7101(1,3)	0.0142	$°C^{-1}$
	C7101(2,3)	0.00667	$°C^{-1}$
	C7101(3,3)	0.00460	$°C^{-1}$
BA	C7101(1,2)	7.50E-14	min^{-1}
	C7101(2,2)	8.26E-9	min^{-1}
	C7101(3,2)	1.38E-5	min^{-1}
	C7101(1,3)	0.0144	$°C^{-1}$

CORSOR Coefficients – Section 2.3.1			
	C7101(2,3)	0.00631	$^{o}C^{-1}$
	C7101(3,3)	0.00290	$^{o}C^{-1}$
I2	C7101(1,2)	7.02E-9	min^{-1}
	C7101(2,2)	2.02E-7	min^{-1}
	C7101(3,2)	1.74E-5	min^{-1}
	C7101(1,3)	0.00886	$^{o}C^{-1}$
	C7101(2,3)	0.00667	$^{o}C^{-1}$
	C7101(3,3)	0.00460	$^{o}C^{-1}$
RU	C7101(1,2)	1.36E-11	min^{-1}
	C7101(2,2)	1.36E-11	min^{-1}
	C7101(3,2)	1.40E-6	min^{-1}
	C7101(1,3)	0.00768	$^{o}C^{-1}$
	C7101(2,3)	0.00768	$^{o}C^{-1}$
	C7101(3,3)	0.00248	$^{o}C^{-1}$
MO	C7101(1,2,7)	5.01E012	min^{-1}
	C7101(2,2,7)	5.93E-8	min^{-1}
	C7101(3,2,7)	3.70E-5	min^{-1}
	C7101(1,3,7)	0.0115	$^{o}C^{-1}$
	C7101(2,3,7)	0.00523	$^{o}C^{-1}$
	C7101(3,3,7)	0.00200	$^{o}C^{-1}$
CE	C7101(1,2)	6.64E-12	min^{-1}
	C7101(2,2)	6.64E-12	min^{-1}
	C7101(3,2)	1.48E-7	min^{-1}
	C7101(1,3)	0.00631	$^{o}C^{-1}$
	C7101(2,3)	0.00631	$^{o}C^{-1}$
	C7101(3,3)	0.00177	$^{o}C^{-1}$
LA	C7101(1,2,9)	5.00E-13	min^{-1}

CORSOR Coefficients – Section 2.3.1			
	C7101(2,2,9)	5.00E-13	min^{-1}
	C7101(3,2,9)	5.00E-13	min^{-1}
	C7101(1,3,9)	0.00768	$°C^{-1}$
	C7101(2,3,9)	0.00768	$°C^{-1}$
	C7101(3,3,9)	0.00768	$°C^{-1}$
UO2	C7101(1,2,10)	5.00E-13	min^{-1}
	C7101(2,2,10)	5.00E-13	min^{-1}
	C7101(3,2,10)	5.00E-13	min^{-1}
	C7101(1,3,10)	0.00768	$°C^{-1}$
	C7101(2,3,10)	0.00768	$°C^{-1}$
	C7101(3,3,10)	0.00768	$°C^{-1}$
CD	C7101(1,2,11)	1.90E-12	min^{-1}
	C7101(2,2,11)	5.88E-9	min^{-1}
	C7101(3,2,11)	2.56E-6	min^{-1}
	C7101(1,3,11)	0.0128	$°C^{-1}$
	C7101(2,3,11)	0.00708	$°C^{-1}$
	C7101(3,3,11)	0.00426	$°C^{-1}$
AG	C7101(1,2,12)	1.90E-12	min^{-1}
	C7101(2,2,12)	5.88E-9	min^{-1}
	C7101(3,2,12)	2.56E-6	min^{-1}
	C7101(1,3,12)	0.0128	$°C^{-1}$
	C7101(2,3,12)	0.00708	$°C^{-1}$
	C7101(3,3,12)	0.00426	$°C^{-1}$
Otherwise	C7101(1,2)	0.0	min^{-1}
	C7101(2,2)	0.0	min^{-1}
	C7101(3,2)	0.0	min^{-1}
	C7101(1,3)	0.0	$°C^{-1}$

CORSOR Coefficients – Section 2.3.1			
	C7101(2,3)	0.0	$^{o}C^{-1}$
	C7101(3,3)	0.0	$^{o}C^{-1}$

CORSOR-M Coefficients – Section 2.3.2			
XE	C7102(1)	2.0E5	min^{-1}
	C7102(2)	63.8	kcal/mole
CS	C7102(1)	2.0E5	min^{-1}
	C7102(2)	63.8	kcal/mole
BA	C7102(1)	2.95E5	min^{-1}
	C7102(2)	100.2	kcal/mole
I2	C7102(1)	2.0E5	min^{-1}
	C7102(2)	63.8	kcal/mole
TE	C7102(1)	2.0E5	min^{-1}
	C7102(2)	63.8	kcal/mole
RU	C7102(1)	1.62E6	min^{-1}
	C7102(2)	152.8	kcal/mole
MO	C7102(1,7)*	23.15	min^{-1}
	C7102(2,7)*	44.1	kcal/mole
CE	C7102(1)	2.67E8	min^{-1}
	C7102(2)	188.2	kcal/mole
LA	C7102(1,9)**	1.46E7	min^{-1}
	C7102(2,9)**	143.1	kcal/mole
UO2	C7102(1,)	1.46E7	min^{-1}
	C7102(2)	143.1	kcal/mole
CD	C7102(1)**	5.95E3	min^{-1}
	C7102(2)**	70.8	kcal/mole
AG	C7102(1)	5.95E3	min^{-1}
	C7102(2)	70.8	kcal/mole
Otherwise	C7102(1,13)	0.0	min^{-1}
	C7102(2,13)	0.0	kcal/mole

CORSOR-M Coefficients – Section 2.3.2

Note. The CORSOR-M model does not consider release from Class 7 (Moly), Class 9 (La) or Class 11 (Cd) to be significant. Previous versions of MELCOR used zero values for these classes when using CORSOR-M. In MELCOR 1.8.5 non-zero release coefficients are supplied as described.

* Coefficients for CORSOR-M class 7 (Moly) are based on a curve fit to the CORSOR release model for Class 7.

** Coefficients for CORSOR-M Class 9 are set identical to the CORSOR-M Class 10 values, following the same assumption as used in the CORSOR model for Class 7. Likewise for Class 11 and 12.

CORSOR-Booth Class Scaling Factors: Nominal Values			
XE	C7103	1.0	-
CS	C7103	1.0	-
BA	C7103	4.0E-4*	-
I2	C7103	6.4E-1	-
TE	C7103	6.4E-1	-
RU	C7103	2.5E-3	-
MO	C7103	1.0E-3	-
CE	C7103	4.0E-8	-
LA	C7103	4.0E-8	-
UO2	C7103	3.2E-4	-
CD	C7103	2.5E-1	-
AG	C7103	1.6E-1	-
CSI	C7103	6.4E-1	
CSM	C7103	1.0	

Release Surface-to-Volume Ratio			
	C7104(1)	422.5	m^{-1}

Modification of Release Rates for Tellurium – Section 2.3.1			
	C7105(1)	0.70	-
	C7105(2)	0.025	-

Note. The previous versions of MELCOR accepted 3 indices (i.e., C7105(1), C7105(2), and C7105(3)). In MELCOR 2.x, since RN classes are identified by the name of the class, the first index that corresponds to the class number is no longer used. As a result, MELCOR 2.x only accepts 2 indices for c7105.

CORSOR-Booth Coefficients for Cesium			
2.3.3	C7106(1,1)	1.0E-6	m^2/s
	C7106(2,1)	1.0E-6	m^2/s
	C7106(3,1)	3.0E4	MWD/MTU
	C7106(4,1)	3.814E5	J/kg-mole
	C7106(5,1)	6.0E-6	M

CORSOR-Booth Class Scaling Factors: Oxidation Modified			
BA	C7107(6)	5.0E-1	-
	C7107(7)	2.0E-3	-
TE	C7107(6)	7.0E-1	-
	C7107(7)	6.4E-1	-
RU	C7107(1)	7.5E-1	-
	C7107(2)	2.3E3	K
	C7107(3)	2.5E-3	-
	C7107(4)	0.0	K^{-1}
	C7107(5)	2.7E3	K
LA	C7107(6)	5.0E-2	-
	C7107(7)	4.0E-8	-
CD	C7107(1)	7.5E-1	
	C7107(2)	2.0E3	
	C7107(3)	2.5E-1	
	C7107(4)	0.0	
	C7107(5)	2.3E3	
AG	C7107(1)	7.5E1	-
	C7107(2)	2.0E3	K
	C7107(3)	1.6E-1	-
	C7107(4)	0.0	K^{-1}
	C7107(5)	2.3E3	K
Otherwise	C7107(1,i)	1.1	-
	C7107(2,i)	0.0	K

CORSOR-Booth Class Scaling Factors: Oxidation Modified			
	C7107(3,i)	0.0	-
	C7107(4,i)	0.0	K^{-1}
	C7107(5,i)	0.0	K
	C7107(6,i)	-1.0	-
	C7107(7,i)	0.0	-

Vapor Pressure – Section 2.5.2			
XE	C7110(1,1)	0.0	K
	C7110(1,2)	-1.0	K
	C7110(2,1)	10000.	K
CS	C7110(1,1)	600.0	K
	C7110(1,2)	13600.	K
	C7110(1,3)	8.895	-
	C7110(1,4)	0.0	-
	C7110(2,1)	1229.5	K
	C7110(2,2)	12100.	K
	C7110(2,3)	7.675	-
	C7110(2,4)	0.	-
BA	C7110(1,1)	1000.	K
	C7110(1,2)	11000.	K
	C7110(1,3)	8.4	-
	C7110(1,4)	0.	-
	C7110(2,1)	10000.	K
I2	C7110(1,1)	273.	K
	C7110(1,2)	3578.0	K
	C7110(1,3)	17.72	-
	C7110(1,4)	-2.51	-
	C7110(2,1)	387.0	K
	C7110(2,2)	3205.0	K
	C7110(2,3)	23.66536399	-
	C7110(2,4)	-5.18	-
	C7110(3,1)	457.0	K
	C7710(3,2)	2176.912045	K
	C7110(3,3)	7.63735266	-

Vapor Pressure – Section 2.5.2			
	C7110(3,4)	0.	-
TE	C7110(1,1,5)	273.	K
	C7110(1,2,5)	13940.0	K
	C7110(1,3,5)	23.51	-
	C7110(1,4,5)	-3.52	-
	C7110(2,1,5)	10000.0	K
RU	C7110(1,1,6)	1500.0	K
	C7110(1,2,6)	33200.0	K
	C7110(1,3,6)	11.6088	-
	C7110(1,4,6)	0.0	-
	C7110(2,1,6)	10000.0	K
MO	C7110(1,1,7)	1500.0	K
	C7110(1,2,7)	32800.0	K
	C7110(1,3,7)	9.68	-
	C7110(1,4,7)	0.0	-
	C7110(2,1,7)	10000.0	K
CE	C7110(1,1,8)	1500.0	K
	C7110(1,2,8)	21570.0	K
	C7110(1,3,8)	8.74	-
	C7110(1,4,8)	0.0	-
	C7110(2,1,8)	10000.0	K
LA	C7110(1,1,9)	1500.0	K
	C7110(1,2,9)	21800.0	K
	C7110(1,3,9)	8.683	-
	C7110(1,4,9)	0.0	-
	C7110(2,1,9)	10000.0	K
UO2	C7110(1,1,10)	1500.0	K
	C7110(1,2,10)	32110.0	K
	C7110(1,3,10)	11.873	-
	C7110(1,4,10)	0.0	-
	C7110(2,1,10)	10000.0	K
CD	C7110(1,1,11)	1000.0	K
	C7110(1,2,11)	13730.0	K
	C7110(1,3,11)	8.43	-
	C7110(1,4,11)	0.0	-

Vapor Pressure – Section 2.5.2			
	C7110(2,1,11)	10000.0	K
AG	C7110(1,1,12)	1000.0	K
	C7110(1,2,12)	15400.0	K
	C7110(1,3,12)	8.15	-
	C7110(1,4,12)	0.0	-
	C7110(2,1,12)	10000.0	K
BO2	C7110(1,1,13)	1000.0	K
	C7110(1,2,13)	19520.0	K
	C7110(1,3,13)	11.125	-
	C7110(1,4,13)	0.0	-
	C7110(2,1,13)	10000.0	K
H2O, CON	C7110(1,1,14:15)	3000.0	K
	C7110(1,2,14:15)	18000.0	K
	C7110(1,3,14:15)	8.875	-
	C7110(1,4,14:15)	0.	-
	C7110(2,1,14:15)	10000.	K
CSI	C7110(1,1)	600.0	K
	C7110(1,2)	10420.0	K
	C7110(1,3)	19.70	-
	C7110(1,4)	-3.02	-
	C7110(2,1)	894.0	K
	C7110(2,2)	9678.0	K
	C7110(2,3)	20.34569113	-
	C7110(2,4)	-3.52	-
	C7110(3,1)	1553.0	K
	C7110(3,2)	7303.903158	K
	C7110(3,3)	7.58405103	-
	C7110(3,4)	0.0	-
CSM	C7110(1,1)	600.0	K
	C7110(1,2)	13600.0	K
	C7110(1,3)	8.895	-
	C7110(1,4)	0.0	-
	C7110(2,1)	1229.5	K
	C7110(2,2)	12100.0	K
	C7110(2,3)	7.675	-

Vapor Pressure – Section 2.5.2			
	C7110(2,4)	0.0	-
Otherwise	C7110(1,1)	3000.0	K
	C7110(1,2)	18000.0	K
	C7110(1,3)	8.875	-
	C7110(1,4)	0.	-
	C7110(2,1)	10000.	K

Vapor Diffusivity Constants – Section 2.5.2			
XE	C7111(1,1)	4.055	Å
	C7111(2,1)	229.0	K
CS, BA	C7111(1,2:3)	3.617	Å
	C7111(2,2:3)	97.0	K
I2	C7111(1,4)	4.982	Å
	C7111(2,4)	550.0	K
Otherwise	C7111(1)	3.617	Å
	C7111(2)	97.0	K

Class Molecular Weights – Section 2.1			
XE	C7120(1)	131.3	kg/kg-mole
	C7120(2)	131.3	kg/kg-mole
CS	C7120(1)	132.905	kg/kg-mole
	C7120(2)	149.913	kg/kg-mole
BA	C7120(1)	137.34	kg/kg-mole
	C7120(2)	137.34	kg/kg-mole
I2	C7120(1)	253.8008	kg/kg-mole
	C7120(2)	253.8008	kg/kg-mole
TE	C7120(1)	127.6	kg/kg-mole
	C7120(2)	143.6	kg/kg-mole
RU	C7120(1)	101.07	kg/kg-mole
	C7120(2)	101.07	kg/kg-mole

Class Molecular Weights – Section 2.1			
MO	C7120(1)	95.94	kg/kg-mole
	C7120(2)	95.94	kg/kg-mole
CE	C7120(1)	140.12	kg/kg-mole
	C7120(2)	140.12	kg/kg-mole
LA	C7120(1)	138.91	kg/kg-mole
	C7120(2)	138.91	kg/kg-mole
UO2	C7120(1)	238.03	kg/kg-mole
	C7120(2)	270.03	kg/kg-mole
CD	C7120(1)	112.4	kg/kg-mole
	C7120(2)	112.4	kg/kg-mole
AG	C7120(1)	118.69	kg/kg-mole
	C7120(2)	118.69	kg/kg-mole
BO2	C7120(1)	69.622	kg/kg-mole
	C7120(2)	69.622	kg/kg-mole
H2O	C7120(1)	18.016	kg/kg-mole
	C7120(2)	18.016	kg/kg-mole
CON	C7120(1)	28.97	kg/kg-mole
	C7120(2)	28.97	kg/kg-mole
CSI	C7120(1)	259.8054	kg/kg-mole
	C7120(2)	259.8054	kg/kg-mole
CSM	C7120(1)	361.75	kg/kg-mole
	C7120(2)	425.75	kg/kg-mole
Otherwise	C7120(1)	28.97	kg/kg-mole
	C7120(2)	28.97	kg/kg-mole

Solubility of RN Classes in Water Films - Section 2.4.2.2			
All classes	C7136	1.0	-

Not Used with LWR COR Package
7140 - Release from Molten U-Al Pools

Not Used with LWR COR Package
7141 - Solubility of Classes in Al-U Alloy
7142 - Debris Particle of Average Surface Area
7143 - Molten Fraction Criterion for Release from U-Al Pools
7144 - Temperature Criterion for Release from Intact Fuel

7150 - SPARC-90 Model Parameters			
Appendix E	7150(1)	10.	-
	7150(2)	5.	-
	7150(3)	1.E-4	-
	7150(4)	25.	-
	7150(5)	1.E-4	-
	7150(6)	25.	-
	7150(7)	1.E-3	-
	7150(8)	25.	-
	7150(9)	1.E12	-
	7150(10)	1.0	-

7151 - SPARC-90 Globule Size Correlation			
Appendix D	7151(1,1)	3.45	-
	7151(2,1)	0.46	-
	7151(1,2)	0.0891	-
	7151(2,2)	0.616	-
	7151(1,3)	0.857	-
	7151(2,3)	0.73	-

7152 - SPARC-90 Bubble Size/Shape Model			
E-1	7152(1)	0.007	m
	7152(2)	-0.2265	-
	7152(3)	0.0203	-
	7152(4)	0.0313	-
	7152(5)	0.5	-
E-2	7152(6)	0.84107	-
	7152(7)	1.13466	cm^{-1}
	7152(8)	-0.3795	cm^{-2}

7153 - SPARC-90 Bubble Rise Velocity Model			
E-3	7153(1)	7.876	cm/s
	7153(2)	0.5	cm
	7153(3)	1.40713	-
	7153(4)	0.49275	-

7154 - SPARC-90 Swarm Velocity Model			
E-5	7154(1)	5.33	liter/s
	7154(2)	3.011E-3	liter-s/cm^2
	7154(3)	0.5	-
	7154(4)	-3.975E-4	cm^{-1}
	7154(5)	170.	cm/s

7155 - SPARC-90 Particle Impaction Model			
D-11	7155(1)	1.79182	-
	7155(2)	3.3437E-11	-
	7155(3)	5.9244E-3	-

7155 - SPARC-90 Particle Impaction Model			
	7155(4)	0.65868	-
D-12	7155(5)	1.13893	-
	7155(6)	1.4173E-6	-
	7155(7)	4.2597E-3	-
	7155(8)	0.99	-

7156 - SPARC-90 Solute Ionization Correlations			
E-12	7156(1)	1.79417	-
	7156(2)	-3.34363	-
	7156(3)	0.021	-
	7156(4)	1.63439	-
	7156(5)	4.30022	-
	7156(6)	1.75467	-
	7156(7)	20.7974	-
	7156(8)	-0.002321	-
	7156(9)	25.	C
	7156(10)	2.0	-

7157 - SPARC-90 Settling Velocity Correlation			
E-19	7157(1)	9.6	-
	7157(2)	27.00	-
	7157(3)	1./1.130	-
	7157(4)	93.6	-
	7157(5)	24.32	-
	7157(6)	1./1.227	-
	7157(7)	410.	-

7157 - SPARC-90 Settling Velocity Correlation			
	7157(8)	15.71	-
	7157(9)	1./1.417	-
	7157(10)	1.07E4	-
	7157(11)	6.477	-
	7157(12)	1./1.609	-
	7157(13)	2.45E5	-
	7157(14)	1.194	-
	7157(15)	1./1.867	-

7158 - SPARC-90 HOI Correlation			
F-4	7158(1)	-1388.89	K
	7158(2)	6.461	-

7159 - SPARC-90 I_2 Chemistry Model Parameters			
Appendix F	7159(1)	1.3882E-3	-
	7159(2)	3279.3	K
	7159(3)	7.7606	$moles^{-1}$
	7159(4)	1370.	K
	7159(5)	1.0423E-2	$moles^2$
	7159(6)	-7148.	K
	7159(7)	4.2271E-9	moles
	7159(8)	-1748.5	K
	7159(9)	1.56531E-13	$moles^2$
	7159(10)	5462.81	K
	7159(11)	-1.87376E6	K^2
	7159(12)	1.E-6	$moles/cm^3$
	7159(13)	1.E-3	-

7160 - Chemisorption Rate Coefficients – Sec. 2.9.2			
	C7160(1,1)	0.139	m/s
	C7160(2,1)	5.96e7	J/kg
	C7160(1,2)	0.035	m/s
	C7160(2,2)	5.96e7	J/kg
	C7160(1,3)	2.0e-7	m/s
	C7160(2,3)	0.0	J/kg
	C7160(1,4)	2.0e-6	m/s
	C7160(2,4)	0.0	J/kg
	C7160(1,5)	5.5e-7	m/s
	C7160(2,5)	2.49e7	J/kg
	C7160(1,6)	9.0e-10	m/s
	C7160(2,6)	3.39e7	J/kg

7170 Hygroscopic Aerosol Sensitivity Coefficients			
	Coefficient	**Value**	**Units**
XE	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	1.0	Kg/m^3
CS	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	3.95	kg/kg H_2O
	C7170(4)	3.95	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	3675.0	kg/m^3

7170 Hygroscopic Aerosol Sensitivity Coefficients			
	Coefficient	**Value**	**Units**
BA	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	5720.0	kg/m^3
I2	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	1.0	kg/m^3
TE	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	5680.0	kg/m^3
RU	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K

7170 Hygroscopic Aerosol Sensitivity Coefficients			
	Coefficient	**Value**	**Units**
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	6970.0	kg/m^3
MO	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	7470.0	kg/m^3
CE	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	7000.0	kg/m^3
LA	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	6510.0	kg/m^3
UO2	C7170(1	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O

7170 Hygroscopic Aerosol Sensitivity Coefficients			
	Coefficient	**Value**	**Units**
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	10960.0	kg/m^3
CD	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	8150.0	kg/m^3
AG	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	6446.0	kg/m^3
BO2	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	2520.0	kg/m^3

7170 Hygroscopic Aerosol Sensitivity Coefficients			
	Coefficient	**Value**	**Units**
H2O	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	1000.0	kg/m^3
CON	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.0	kg/kg H_2O
	C7170(4)	0.0	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	2250.0	kg/m^3
CSI	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.44	kg/kg H_2O
	C7170(4)	2.25	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	4510.0	kg/m^3
CSM	C7170(1)	273.0	K
	C7170(2)	373.0	K
	C7170(3)	0.67	kg/kg H_2O
	C7170(4)	0.67	kg/kg H_2O
	C7170(5)	600.0	K
	C7170(6)	647.0	K

7170 Hygroscopic Aerosol Sensitivity Coefficients			
	Coefficient	**Value**	**Units**
	C7170(7)	2	ions/molecule
	C7170(8)	0	Ions/molecule
	C7170(9)	4030.0	kg/m^3

Note. Values colored blue indicates updated default values for MELCOR version 2.1. The old default values used in previous versions of MELCOR can be found in MELCOR User's Guide.

Appendix B: Agglomeration Kernels

The agglomeration kernels currently implemented in the MELCOR implementation of MAEROS are summarized in this appendix. These kernels are those that are recommended by Powers, Sprung, and Leigh [1].

Brownian

$$\beta = 2\pi (D_i + D_j)(\gamma_i d_i + \gamma_j d_j)/F \quad \text{(B-1)}$$

$$D_i = \frac{kT}{3\pi d_i \mu \chi_i} C_i \quad \text{(B-2)}$$

$$C_i = 1 + Kn_i\left[c_m + 0.4\ \exp\left(-1.1/Kn_i\right)\right] \quad \text{(B-3)}$$

$$F = \frac{d_i + d_j}{d_i + d_j + 2g_{ij}} + \frac{8\left(D_i + D_j\right)}{v_{i,j}\left(d_i + d_j\right)c_s} \quad \text{(B-4)}$$

$$g_{i,j} = \left(g_i^2 + g_j^2\right)^{1/2} \quad \text{(B-5)}$$

$$v_{i,j} = \left(v_i^2 + v_j^2\right)^{1/2} \quad \text{(B-6)}$$

$$g = \frac{1}{3\ d_i\ l_i}\left[\left(d_i + l_i\right)^3 - \left(d_i^2 + l_i^2\right)^{3/2}\right] - d_i \quad \text{(B-7)}$$

$$l = \frac{8D_i}{\pi v_i} \quad \text{(B-8)}$$

$$v = \left(\frac{8kT}{\pi m_i}\right)^{1/2} \quad \text{(B-9)}$$

$$Kn_i = 2\lambda / d_i \quad \text{(B-10)}$$

$$\lambda = \frac{\mu}{\rho_g}\left(1.89 \times 10^{-4}\ M_{w,g}/T\right)^{1/2} \quad \text{(B-11)}$$

$$\rho_g = 1.21 \times 10^{-4} P\, M_{w,g}/T \quad \text{(B-12)}$$

μ = values for air; from the Material Properties (MP) package

Gravitational

$$\beta = \varepsilon_g \tfrac{\pi}{4} c_s \left(\gamma_i d_i + \gamma_j d_j\right)^2 \left|V_{Ti} - V_{Tj}\right| \quad \text{(B-13)}$$

$$V_{Ti} = \frac{\rho_{pi}\ g d_i^2 C_i}{18 \mu \chi_i} \quad \text{(B-14)}$$

$$\varepsilon_g = 1.5 \left[\frac{\min\ \left(d_i, d_j\right)}{\left(d_i + d_j\right)}\right]^2 \quad \text{(B-15)}$$

Turbulent

$$\beta = c_s \left(\ \beta_{T1}^2 + \beta_{T2}^2\ \right)^{1/2} \quad \text{(B-16)}$$

$$\beta_{T1} = \left(\frac{\pi \varepsilon_T\ \rho_g}{120 \mu}\right)^{1/2} \left(\gamma_i d_i + \gamma_j d_j\right)^3 \quad \text{(B-17)}$$

$$\beta_{T2} = \frac{0.04029\ \rho_g^{1/4}\ \varepsilon_T^{3/4}}{\mu^{5/4}} \left(\gamma_i d_i + \gamma_j d_j\right)^2 \left|\frac{\rho_{p1} C_1 d_1^2}{\chi_1} - \frac{\rho_{p2} C_2 d_2^2}{\chi_2}\right| \quad \text{(B-18)}$$

Nomenclature	
c_m	particle slip coefficient
c_s	particle sticking coefficient
c_t	thermal accommodation coefficient
C	particle mobility
d	particle diameter
D	diffusion coefficient
k	Boltzmann constant
k_g/k_s	ratio of thermal conductivity of the gas over that for the particle
Kn	Knudsen number
m	particle mass
M_w	molecular weight
P	pressure
T	temperature
V	volume

Greek:	
β	coagulation kernel (m^3/s)
ε_T	turbulence dissipation density
ρ	density
μ	viscosity
λ	mean free path
γ	agglomeration shape factor
χ	dynamic shape factor

Subscripts	
b	bulk
g	gas (air assumed)
i,j	particle identifier
p	particle
s	Steam

Appendix C: Aerosol Surface Area

The aerosol surface area is used for fission product vapor condensation and evaporation of aerosols. The general equation for the surface area is:

$$A_T = \int_{x_1}^{x_2} n(x)A(x)dx \tag{C-1}$$

where

A_T total surface area

$A(x)$ area of a particle as a function of x

$n(x)$ number of particles as a function of x

MAEROS assumes that the aerosol size distribution in each section is constant with respect to the natural log of the mass, so the number density can be expressed as (Gelbard [7]):

$$n(x) = \frac{M}{m\ \left(\ln m_2 - \ln m_1\right)}\ d\left(\ln m\right) \tag{C-2}$$

A and m can be expressed in terms of ln m as follows:

$$m = e^{\ln m} \tag{C-3}$$

$$A = 4\pi r^2 \tag{C-4}$$

$$= 4\pi\left(\frac{3m}{4\pi\rho}\right)^{2/3} \tag{C-5}$$

$$= 4\pi\left(\frac{3e^{\ln m}}{4\pi\rho}\right)^{2/3} \tag{C-6}$$

Equation (C-1) becomes

$$A_T = 4\ \pi \left(\frac{3}{4\pi\rho} \right)^{2/3} \frac{M}{\ln\ (m_2/m_1)} \int_{\ln m_1}^{\ln m_2} \exp\left(-\tfrac{1}{3}\ \ln m\right) d(\ln m) \qquad \text{(C-7)}$$

and, after integration,

$$A_T = 12\pi \left(\frac{3}{4\pi\rho} \right)^{2/3} \frac{M}{\ln(m_2/m_1)} \left[m_1^{-1/3} - m_2^{-1/3} \right] \qquad \text{(C-8)}$$

Appendix D: Pool Scrubbing Vent Exit Region Modeling

D.1 Globule Formation

The initial globules formed have a unique size given by a correlation relating the normalized globule volume to the Weber number for each vent type considered. The correlation is

$$v_n = a \cdot We^b \tag{D-1}$$

$$D_{g,o}^3 = \frac{3}{2} v_n D_o^2 \left(\frac{\sigma}{\rho_l g} \right)^{1/2} \tag{D-2}$$

where

$$We = \frac{\rho_l D_o V_o^2}{\sigma} \tag{D-3}$$

and

ρ_l pool liquid density, kg/m^3,

σ pool liquid surface tension, N/m,

D_o vent equivalent diameter,m, and

V_o exit velocity of the gas, m/s.

It is assumed that $Q = V_o(\pi/4)D_o^2$, where Q is the gas volumetric flow rate at the vent in equilibrium with the pool conditions at the vent depth. The default correlation constants implemented in sensitivity coefficient array C7151 are:

Vent	a	b	Source
Multiple small holes	3.45	0.46	EPRI program
Downcomer	0.0891	0.616	PNL with EPRI data
Horizontal vent	0.857	0.73	EPRI program

These correlations only apply to inlet gases containing noncondensible gases. Very high steam fractions provide for residual bubbles. High steam fractions have a “cone”-shaped region that does not detach from the vent.

The globule diameter decreases linearly to zero over a distance of twelve times its initial value:

$$D_g = D_{g,o}\left[1 - \frac{x}{12D_{g,o}}\right] \tag{D-4}$$

where x has a value of zero at the elevation of the vent exit.

D.2 Vent Exit Region Scrubbing Models

In the vent exit region, aerosol capture occurs because of:

(1) Stefan flow from steam condensation during gas equilibration to pool conditions,

(2) inertial impaction of aerosol particles in rapidly decelerating gas flow, and

(3) centrifugal, diffusional and gravitational particle deposition during gas injection through small orifice, multihole vents.

D.2.1 Steam Condensation

It is assumed that the fraction of particles removed by steam condensation during globule breakup at the vent exit is simply equal to the fractional loss in gas volume caused by condensation at the temperature and pressure of the pool at the vent depth:

$$DF_{EC} = \frac{X_o}{X_i} \tag{D-5}$$

where X_i is the mole fraction of noncondensible gas in the inlet gas and X_o is the mole fraction of noncondensible gas in the gas after equilibration. X_i is determined from the flow composition in the vent provided by the FL package, and X_o is given by

$$X_o = 1 - \frac{P_{sat}(T_p)}{P_{surf} + \rho_l g h_p} \tag{D-6}$$

where T_p, P_{surf} and h_p are the pool temperature, pressure at the pool surface and pool depth at the vent exit, respectively. DF_{EC} is limited to a minimum value of one.

For iodine vapor scrubbing the value of DF_{EC} calculated above may need to be reduced significantly. The concentration of iodine in the condensate may not exceed the product of the equilibrium partition coefficient and the concentration of iodine in the vapor state remaining in the bubbles. Hence, if the concentration of iodine in the condensate would

exceed the equilibrium value consistent with the partition coefficient when iodine removal is assumed to be proportional to the volumetric reduction factor (DF_{EC}), then iodine vapor scrubbing does not occur to the extent given by DF_{EC}. Rather, the decontamination factor for vapor scrubbing is given by

$$DF_{EC,vap} = 1 + \min\left[H_{vap}\left(\frac{\rho_v}{\rho_l} \right), 1 \right] (DF_{EC} - 1) \qquad \text{(D-7)}$$

so that $DF_{EC,vap}$ is less than DF_{EC} if $H_{vap}(\rho_v / \rho_l) < 1$, and equal to DF_{EC} otherwise. (Additional vapor diffusion into the aqueous phase is considered to be too slow with respect to the time scale of steam condensation to increase $DF_{EC,vap}$ above DF_{EC} in those cases where $H_{vap}(\rho_v / \rho_l)$ exceeds unity.)

D.2.2 Inertial Impaction

If gas leaves the vent exit at a high velocity, the initial globules rapidly lose that velocity. The forward globular interface, as it slows and stops, can capture particles if they have sufficient inertia. Inertia of particle size *i* is represented by the Stokes number

$$Stk_i = \frac{\rho_i V_e d_i^2}{9 \mu D_o} \qquad \text{(D-8)}$$

where

d_i particle diameter

ρ_i particle density

V_e vent exit gas velocity (before equilibration with pool)

μ gas viscosity

D_o vent exit orifice diameter

The *DF* for this impaction process is

$$DF_{II,i} = \frac{1}{1 - \alpha_i} \qquad \text{(D-9)}$$

where

$$\alpha_i = 1.79182\left(3.3437\times10^{-11}\right)^{\left(5.9244\times10^{-3}\right)\sqrt{Stk_i}} \quad \text{if } \sqrt{Stk_i} \le 0.65868 \tag{D-10}$$

and

$$\alpha_i = 1.13893\left(1.4173\times10^{-6}\right)^{\left(4.25973\times10^{-3}\right)\sqrt{Stk_i}} \quad \text{if } 0.65868 < \sqrt{Stk_i} \tag{D-11}$$

The constants in these correlations have been implemented in sensitivity coefficient array 7155, and the maximum value of α_i permitted is constrained to 0.99, which is also included in sensitivity coefficient array 7155. The importance of inertial impaction is minimal unless near-sonic values of V_e occur.

D.2.3 Centrifugal, Diffusional and Gravitational Deposition

Centrifugal, diffusional and gravitational particle deposition are only evaluated in the vent exit region for small orifice, multihole vents (MVENT=1 on input record RN2_PLS). The bases for the model are assumptions about the vent injection bubble geometry and velocity relative to pool liquid.

Particle scrubbing is evaluated in two connected time intervals. The injection interval is defined as the time it takes to fill the globule and is given by

$$\tau_{fill} = \frac{\frac{\pi}{6}D_g^3}{\frac{\pi}{4}D_o^2 V_o} \tag{D-12}$$

During the detached globule interval, which follows the injection interval, the globule is slowed by drag forces. This interval is assumed to be three times the characteristic stopping time, which is the time required for the drag force to nullify the bubble momentum

$$\tau_{stop} = \frac{4\rho_g D_g}{3f\rho_l V_o} \tag{D-13}$$

where *f*, the friction factor, has a hard-wired value of 0.2. It is assumed that the final detached injection globule is spherical (of diameter D_g) and during globule formation, the forming globule is elongated with a hemispherical front of diameter D_o (orifice diameter) moving at velocity V_o relative to the bulk liquid.

For each particle size, denoted by subscript *i*, the centrifugal deposition velocity is calculated by scaling the gravitational settling velocity by the ratio of the centrifugal acceleration to the gravitational acceleration. The gas circulation velocity is assumed to equal the injection velocity and the radius of curvature is equal to the circular vent radius. The decontamination factors during the injection and detached globule periods are

proportional to the ratio of the volumes swept to the globule surface by centrifugal velocity to the total globule volume. The values are given by

$$DF_{C,i} = \exp\left(V_{C,i}/V_o\right) \quad \text{during injection}$$

$$DF_{C,i} = \exp\left(\frac{V_o V_{g,i}\rho_g}{9D_o fg\rho_l}\right) \quad \text{after detachment} \tag{D-14}$$

where $V_{c,i}$ is given by Equation (E.20) with $r_c = D_o/2$ and $V_s = V_o$. The method used to determine the settling velocity, $V_{g,i}$, is also described in Section E.3 below.

Particle deposition from Brownian diffusion during the injection and detached globule periods is modeled using film penetration theory, which is discussed in Section E.2 below. The decontamination factor during each period is proportional to the ratio of the volume swept to the globule surface by Brownian diffusion to the total globule volume. For each particle size, the decontamination factors are given by

$$DF_{D,i} = \exp\left[\frac{16\tau_{fill}}{3D_o}\left(\frac{D_i}{\pi\tau_{fill}}\right)^{1/2}\right] \quad \text{during injection}$$

$$DF_{D,i} = \exp\left[\frac{12\tau_{stop}}{D_o}\left(\frac{V_o D_i}{\pi D_o}\right)^{1/2}\right] \quad \text{after detachment} \tag{D-15}$$

Particle deposition from gravitational settling during the injection and detached globule periods both use a settling velocity based on Stokes's law for small particles and an empirical correlation based on the Reynolds number for larger particles. These correlations are presented in Section E.3 below. For each particle size, the decontamination factors are given by

$$DF_{g,i} = \exp\left(\frac{\overline{A}_s V_{g,i}\tau_{fill}}{(\pi/6)D_{g,o}^3}\right) \quad \text{during injection}$$

$$DF_{g,i} = \exp\left(\frac{3V_{g,i}\left(3\tau_{stop}\right)}{2D_{g,o}}\right) \quad \text{after detachment} \tag{D-16}$$

where the average settling area, $\overline{A}_s$, during injection is equal to one half the final settling area of the horizontally oriented bullet-shaped globule given by

$$A_{s,o} = \frac{2D_{g,o}^3}{3D_o} + D_o^2\left(\frac{\pi}{8} - \frac{1}{3}\right) \quad \text{(D-17)}$$

The overall decontamination factor resulting from centrifugal, diffusional and gravitational deposition, $DF_{ER,i}$, which is used in Equation 2.7.3 in Section 2.7.1, is given by

$$DF_{ER,i} = DF_{c,i} \cdot DF_{D,i} \cdot DF_{g,i} \quad \text{(D-18)}$$

Appendix E: Pool Scrubbing Swarm Rise Region Modeling

The primary modeling objective in the swarm rise region is to determine how evolving thermal-hydraulic conditions within the bubbles affect the removal of particulate aerosols and iodine vapors from the bubble. This is achieved by dividing the total rise height (the distance between the vent exit and the pool surface) into several equal sections (given by XNRISE implemented in sensitivity coefficient array C7150 with a default value of 10.), and then marching upward to update the thermal-hydraulic conditions in each section to evaluate the incremental removal in each section. In this procedure it is assumed that the swarm velocity is constant, so that the bubbles spend the same amount of time in each section.

E.1 Bubble Characteristics

The bubbles are modeled as oblate spheroids with an equivalent spherical diameter, d_{vm}, of 0.7 cm (this default value can be adjusted through sensitivity coefficient array C7152), if they contain no steam initially. The presence of steam reduces d_{vm} as follows

$$d_{vm} = 0.7 \cdot 10^{-0.2265+(0.0203+0.0313 x_{nc})^{1/2}} \qquad \text{(E.1)}$$

The constants in Equation (E.1) are implemented in sensitivity coefficient array 7152. The shape of the bubble is calculated using the following correlation

$$\frac{a}{b} = 0.84107 + 1.13466\, d_{vm} - 0.3795\, d_{vm}^2 \qquad \text{(E.2)}$$

where *a* and *b* are the lengths of the major and minor axes of the oblate spheroid, respectively. (NOTE: The SPARC-90 documentation erroneously had *a/b* inverted as *b/a*.) This correlation was established for $0.15 \leq d_{vm} \leq 1.3$ cm. All bubbles smaller than 0.15 cm are spheres (*a/b* = 1), and bubbles larger than 1.3 cm have *a/b* = 1.675. The constants in Equation (E.2) are implemented in sensitivity coefficient array C7152.

The bubble rise velocity relative to the liquid is given by the following correlation

$$\begin{aligned} V_r &= 7.876(\sigma/\rho_l)^{1/4}\ (cm/s) \quad for\ d_{vm} \leq 0.5\ cm \\ V_r &= 1.40713 d_{vm}^{0.49275} V_{r,T}\ (cm/s) \quad otherwise \end{aligned} \qquad \text{(E.3)}$$

Where $V_{r,T}$ is the bubble rise velocity evaluated at the correlation transition, i.e., at d_{vm}= 0.5cm = C7153(2). The constants in Equation (E.3) have been implemented as sensitivity coefficient array C7153. The swarm rise velocity is given by the average of the correlation value at the depth of the vent exit and at the pool surface

$$\overline{V_s} = 0.5 \cdot \left[V_s(x=0) + V_s(x=h_p) \right] \quad \text{(E.4)}$$

where

$$V_s(x) = \left[\left(\dot{Q}_s + 5.33 \right) / 3.011 \cdot 10^{-3} \right]^{1/2} \left[1 - 3.975 \cdot 10^{-4} x \right] \quad \text{(cm/s)} \quad \text{(E.5)}$$

where $\dot{Q}_s$ is the total gas volumetric flow rate (liter/s) at depth $h_p/2$, and the depth, x, is measured in cm. (NOTE: In the SPARC-90 documentation, Equation (E.4) is erroneous.) $\overline{V_s}$ is limited to a maximum value of 170 cm/s, which is implemented in sensitivity coefficient array C7154 along with the constants that appear in Equation (E.5).

E.2 Bubble Heat and Mass Transfer

During each spatial step the change in the internal energy of the gas in the bubbles is tracked by evaluating iteratively the work performed by bubble expansion and the heat and mass transfer from the pool to the bubble across the bubble boundary. Because the stable size of the bubbles is assumed to remain constant, as the bubbles expand as the static pressure decreases during their ascent, they are assumed to multiply by splitting. The particle concentration (g/cm^3) in the bubbles decreases as a result of bubble multiplication to conserve mass. Bubble expansion during each step is evaluated by assuming that the bubble is isothermal and that the increase in volume is inversely proportional to the decrease in static head (i.e., by assuming ideal gas behavior).

The work done by the expanding bubble during each step is also evaluated by assuming ideal gas behavior and is given by:

$$\begin{aligned} Pdv &= P\left[\left(\frac{\partial V}{\partial P}\right)_T dP + \left(\frac{\partial V}{\partial T}\right)_P dT + \Delta V_{evap}\right] \\ Pdv &= R\left[M_t T_t ln\left(\frac{P_1}{P_2}\right) + (T_2 - T_1)M_t + T_1 \Delta M_{evap}\right] \end{aligned} \quad \text{(E.6)}$$

where ΔM_{evap} is the net increase in vapor mass from evaporation into the bubble minus condensation onto particles in the bubble during the spatial step and M_t is the sum of the steam and noncondensible mass in the bubble.

Heat and mass transfer rates between the pool and rising bubbles are based on penetration theory, in which it is assumed that the top-to-bottom gas circulation in the bubble establishes a quasi-steady boundary layer with the leading edge at the top of the bubble. The transport velocity through the boundary layer at a distance ℓ from the leading edge of the boundary layer is then given by:

$$V_\phi = \left(\frac{D_\phi V_\ell}{\pi \ell} \right)^{1/2} \tag{E.7}$$

where $D_\phi = \alpha$ (thermal diffusivity of boundary layer fluid) for heat transfer, $D_\phi = D_i$ (mass diffusivity or diffusion coefficient of species *i* through the boundary layer) for mass transfer, V_ℓ is the gas circulation velocity parallel to the boundary layer and $T_e = \ell / V_\ell$ is commonly termed the “exposure” time of the surface. The heat transfer coefficients from the pool and bubble to the pool/bubble interface are:

$$h_p = \rho_l C_l (\alpha_l / \pi \tau_e)^{1/2}$$
$$h_b = \rho_g C_g (\alpha_g / \pi \tau_e)^{1/2} \tag{E.8}$$

respectively; and the rate of evaporation at the pool/bubble interface is:

$$N_s = \left(\frac{P}{R T_s} \right) \left(\frac{D_s}{\pi \tau_e} \right)^{\frac{1}{2}} \log_e \left[\frac{P - P_{sat}(T_b)}{P - P_{sat}(T_s)} \right] \tag{E.9}$$

where the subscript *s* indicates that the term is evaluated at the pool/bubble interface.

The rates of condensation and evaporation of water on aerosol particles are determined using the Mason equation:

$$r \frac{dr}{dt} = \frac{S - S_r}{a + b} \tag{E.10}$$

which gives the time rate of change of the radius, *r*, of the aerosol particle as a function of the difference between the actual saturation ratio, S (defined as $P_v/P_{sat}(T)$ inside the bubble), and its equilibrium value, S_r. *a*, *b* and S_r in Equation (E.10) are given by:

$$a = \frac{\rho_l \, M_w h_{fg}^2}{k_v \, RT^2}$$

$$b = \frac{\rho \, RT}{P_{sat} D_v M_w} \qquad \text{(E.11)}$$

$$S_r = A_l \exp\left[\frac{2\sigma M_w}{r \, R \, \rho_l T}\right]$$

The heat and mass transfer equations are solved iteratively during each spatial step in the bubble's ascent. During each iteration, the heat and mass transfer rates over the bubble surface are numerically integrated by dividing the surface from $0°$ at the top of the bubble to $180°$ at the bottom of the bubble into a number (XNCIRC, with a default value of 5. implemented in sensitivity coefficient array 7150) of equally spaced latitudinal strips. The heat and mass transfer over all the strips are summed to obtain the integral values, and the exposure time, τ_e, associated with each strip is saved for use in calculating decontamination factors after the thermal-hydraulic calculations. There are actually three separate iterations to determine the end-of-step values of saturation ratio, *S*, and the vapor/aerosol temperature, T_b, inside the bubbles. The error tolerances and iteration limits associated with these calculations for saturation, energy and temperature are implemented in sensitivity coefficient array 7150, and have default values that normally yield reasonable accuracy with acceptable computational cost.

E.3 Particle Scrubbing in the Bubbles

Particle scrubbing in the bubbles is the result of a net flux of particles to the bubble boundary, where they are assumed to be absorbed into the pool with perfect efficiency. The decontamination factor during a time interval is defined to be the mass of particles in the bubble at the beginning of the interval divided by the mass of particles in the bubble at the end of the interval. It is assumed that particle removal in the bubble can be modeled as a first-order process as follows:

$$\frac{d \, c_i}{d \, t} = -\left[\frac{1}{v_b}\int A_{surf} V_{n,i} \, d \, A\right] c_i \qquad \text{(E.12)}$$

where c_i is the concentration of particles of size i in the bubble, v_b is the bubble volume and $V_{n,i}$ is the velocity of the particles normal to (toward) the surface of the bubble, A_{surf}. The decontamination factor during a time interval, Δt, is obtained from the solution to this equation and is given by:

$$DF_{BB,i} = \exp\left[\frac{\Delta t}{V_b}\int A_{surf} V_{n,i}\, dA\right] \quad \text{(E.13)}$$

The particle velocity normal to the surface is given by the normal component of the vector sum of the velocities associated with the individual deposition mechanisms.

$$V_{n,j} = V_{c,i} + V_{D,i} - V_{g,i}\ \cos\beta - V_v \quad \text{(E.14)}$$

where β is the angle between the normal vector and vertical. For the assumed elliptical geometry (the cross section of an oblate spheroid), β is given by:

$$\beta = \tan^{-1}\left[\frac{b^2 \tan\theta}{a^2}\right] \quad \text{(E.15)}$$

where θ is the polar angle between the vertical and the ray that runs from the origin to the given point on the ellipse (θ runs from 0 to π from the top to the bottom of the bubble).

For particles with a diameter less than about 70 microns, the gravitational settling velocity, $V_{g,i}$, follows Stokes's law and is given by:

$$V_{g,i} = \frac{\rho_i g S_i d_i^2}{18\,\mu} \quad \text{(E.16)}$$

where S_i is the Cunningham slip correction factor. For larger particles, a set of empirical correlations is used to determine the Reynolds number, from which $V_{g,i}$ follows:

$$f(\mathrm{Re}) = \frac{4\,\rho_i\,\rho_g\,g\,d_i^3}{3\,\mu^2} \quad \text{(E.17)}$$

$$\begin{aligned} Re &= \left[\frac{f(Re)}{27.00}\right]^{1/1.130} \quad if\ 9.6 < f(Re) < 93.6 \\ &= \left[\frac{f(Re)}{24.32}\right]^{1/1.227} \quad if\ 93.6 \le f(Re) < 410. \\ &= \left[\frac{f(Re)}{15.71}\right]^{1/1.417} \quad if\ 410. \le f(Re) < 1.07 \times 10^4 \end{aligned} \quad \text{(E.18)}$$

$$= \left[\frac{f(Re)}{6.477}\right]^{1/1.609} \quad if\ 1.07 \times 10^4 \leq f(Re) < 2.45 \times 10^5$$

$$= \left[\frac{f(Re)}{1.194}\right]^{1/1.867} \quad if\ 2.45 \times 10^5 \leq f(Re)$$

$$V_{g,i} = \frac{\mu \mathrm{Re}}{\rho_g d_i} \tag{E.19}$$

The centrifugal capture velocity, $V_{c,i}$, can be obtained from the gravitational settling velocity by scaling it by the ratio of the centrifugal acceleration to the acceleration of gravity (even though the original derivation is based on the concept of particle mobility):

$$V_{c,i} = V_{gi}\left(\frac{V_s^2/r_c}{g}\right) \tag{E.20}$$

The local surface velocity, V_s, is calculated by assuming that flow around the rising bubbles is essentially steady three-dimensional, axisymmetric, inviscid, incompressible, irrotational (potential) flow around an oblate spheroid. The stream function, ψ, for this flow can be found by solving the irrotational vorticity equation $(\nabla x V = 0)$ with $V = \nabla x (0, 0, \Psi/\sigma)$ chosen to satisfy the continuity equation $(\nabla \bullet V = 0)$ identically. The solution is effected by using a conformal mapping to transform the equations from cylindrical (radial coordinate σ) to elliptical coordinates. (It can be shown that the equations reduce to those for flow around a sphere as $a/b \rightarrow 1$.) V_s and the radius of curvature of the surface, r_c, which are used to calculate centrifugal acceleration in Equation (E.20) above are given by the following:

$$V_s = \frac{-V_r\,(\,r \sin\theta / a\,)}{\left[(r\cos\theta/b)^2 + \sinh^2\xi_o\right]^{1/2}} \left[\sinh\xi - \cosh^2\xi_o \cot^{-1}(\sinh\xi_o)\right] \tag{E.21}$$

$$r_c = \frac{1}{a\,b}\left[\frac{a^4\cos^2\theta + b^4\sin^2\theta}{a^2\cos^2\theta + b^2\sin^2\theta}\right]^{\frac{3}{2}}$$

where

$\sinh\xi_o$ $[\,(a/b)_2 - 1\,]^{-1/2}$

$\cosh \xi_o$	$[1 - (b/a)^2]^{-1/2}$
r	$[(\sin \theta /a)^2 + (\cos \theta /b)^2]^{-1/2}$ (radial coordinate)
θ	spherical polar coordinate $(0 \le \theta \le \pi)$

Note that V_s is presented in spherical coordinates rather than the elliptical coordinates used in the derivation.

The local diffusional deposition velocity, $V_{D,i}$, from Brownian diffusion can be estimated from penetration theory of mass transfer:

$$V_{D,i} = \left(\frac{D_i}{\pi \tau_e} \right)^{1/2} \quad \text{(E.22)}$$

where the diffusion coefficient for the aerosol particles, D_i, can be calculated using the Stokes-Einstein equation:

$$D_i = \frac{k_B T S_i}{3 \pi \mu D_i} \quad \text{(E.23)}$$

where k_B is the Boltzmann constant and T is the gas temperature in the bubble. The exposure time of the moving surface, τ_e, in Equation (E.22) is the integrated value of the arc length divided by the local surface velocity starting from $\tau_e = 0$ at the top of the bubble. When evaporation is occurring at the surface of the bubble, the diffusion velocity from Equation (E.22) is reduced as follows:

$$V'_{D,i} = \xi_i V_{D,i} \quad \text{(E.24)}$$

where

$$\xi_i = \frac{\exp(-\phi_i^2)}{2 - \exp(-1.85\, \phi_i)} \quad \text{(E.25)}$$

and the parameter ϕ is equal to $V_v/V_{D,i}$.

The normal component of the deposition velocity given by Equation (E.14) is limited to a minimum value of $V_{D,i}$ and then integrated over the entire bubble surface in Equation (E.13) to get $DF_{BB,i}$, which is used in Equation 2.7.2 in Section 2.7.1.

Appendix F: Iodine Vapor Scrubbing in the Swarm Rise Region

In the swarm rise region, the scrubbing of I_2 and CH_3I is controlled by diffusion of those species. The resistance to diffusion on both sides of the pool/bubble interface is considered to evaluate effective diffusion velocities through the gas and through the liquid. As discussed in Appendix D, in the swarm rise region it is assumed that bubble circulation continually renews the bubble interface and that the film theory of mass transfer resistance holds on both sides of the interface. Because the boundary layer thickness and mass transport through it are functions of the angular position around the rising bubbles, the decontamination factors for each spatial rise step must be evaluated by numerically integrating diffusion velocity over the polar angle of the assumed spherical bubble geometry. Hence, the decontamination factors are given by Equation (E.13) with the particle deposition velocity, $V_{n,i}$, replaced by the gaseous diffusion velocities for I_2 and CH_3I, $V_{D,j}$ (j = I_2 or CH_3I), given by (brackets refer to concentrations)

$$V_{D,j} = V'_{D(g),j} \left\{ \frac{\exp\left[-\left(V_v / V'_{D(g),j}\right)^2\right]}{2 - \exp\left[-1.85\left(V_v / V'_{D(g),j}\right)\right]} \right\}$$

$$V'_{D(g),j} = \frac{1}{\left[I_{j(g)}\right]} \left\{ V''_{D(g),j} \left(\left[I_{j(g)}\right] - \left[I_{j(g)}\right]_i \right) \right\} \qquad \text{(F.1)}$$

$$V''_{D(g),j} = \left(\frac{D_{g,j} V_\ell}{\pi \ell} \right)^{\frac{1}{2}}$$

where the subscript i indicates the concentration is evaluated at the gas/liquid interface. $\left[I_{j(g)}\right]_i$ is determined by equating the concentration flux from the gas to the interface with the flux from the interface to the pool as follows:

$$V''_{D(g),j} \left(\left[I_{j(g)}\right] - \left[I_{j(g)}\right]_i \right) = V''_{D(g),j} \left(H(I_j) \left[I_{j(g)}\right]_i - \sum \left[I_{j(aq)}\right] \right) \qquad \text{(F.2)}$$

where (g) and (l) indicate the diffusion coefficient applies to the gas and liquid phase, respectively, and $H(I_j)$ is the partition coefficient for species j. Gaseous concentrations in the bubble are updated after each spatial step. However, it is assumed that the total iodine concentration in the pool does not change significantly during the transit time of bubbles (vent depth divided by average swarm velocity) so that its value is updated only at the beginning of each MELCOR system timestep.

The total liquid molar concentration of iodine is given by

$$\sum\left[I_{2(aq)}\right] = 0.5\,[I^-] + 1.5\,[I_3^-] + \left[I_{2(aq)}\right] + 0.5\,[HIO] + 0.5\left[H_2OI^+\right] \qquad (F.3)$$

CH_3I is the only organic iodine species considered, and its temperature-dependent partition coefficient is given by the following correlation:

$$H_{IO} = \frac{T}{10^{-1388.89/T + 6.461}} \qquad (F.4)$$

and the constants in the exponent of the denominator have been implemented as sensitivity coefficient array 7158. The partition coefficient of inorganic iodine, $H(I_2)$, is updated during each spatial step by determining the pool equilibrium inorganic iodine species concentrations and solving for $H(I_2)$ as follows:

$$H(I_2) = \frac{\sum\left[I_{2(aq)}\right]_{(eq)}}{\sum\left[I_{2(aq)}\right]_{(eq)}/K_1} = \frac{\sum\left[I_{2(aq)}\right]_{(eq)}}{\left[I_{2(aq)}\right]_{(eq)}} \qquad (F.5)$$

where K_1 is the equilibrium constant for the first reaction listed in Equation (F.6) below. Hence, in the SPARC-90 treatment of I_2 vapor scrubbing, the major task involves determining the equilibrium concentrations of inorganic iodine species, which can be used to calculate an appropriate value of $H(I_2)$ from Equation (F.5). The equilibrium concentrations of the inorganic iodine species are obtained by considering a limited set of chemical reactions involving inorganic iodine. At equilibrium, this set of reactions yields a set of simultaneous algebraic equations that relate the equilibrium concentrations of the various reactants and products to one another. The solution of this set of equations determines the required equilibrium concentrations of the inorganic iodine species.

In SPARC-90 it is assumed that the equilibrium concentration of the most important inorganic iodine species is determined by the equilibrium solution for the following set of fast reactions

$$I_{2(g)} \leftrightarrow I_{2(aq)}$$

$$I_{2(aq)} + I^- \leftrightarrow I_3^-$$

$$I_{2(aq)} + H_2O \leftrightarrow H^+ + I^- + HIO \qquad (F.6)$$

$$I_{2(aq)} + H_2O \leftrightarrow H_2OI^+ + I^-$$

$$H_2O \leftrightarrow H^+ + OH^-$$

Slow aqueous reactions that affect the concentrations of these species, including radiation-induced pH changes, are not modeled in SPARC-90. However, if accident sequences provide excess CsOH as expected (pH remains high), the models might still be adequate.

At equilibrium the relationship between the reactants and products in Equation (F.6) is given by

$$\left[I_{2(aq)}\right]_{(eq)} = K_1 \left[I_{2(g)}\right]_{(eq)} \quad \text{(F.7)}$$

$$\left[I_3^-\right]_{(eq)} = K_2 \left[I_{2(aq)}\right]_{(eq)} \left[I^-\right]_{(eq)} \quad \text{(F.8)}$$

$$\left[H^+\right]_{(eq)} \left[I^-\right]_{(eq)} \left[HIO\right]_{(eq)} = K_3 \left[I_{2(aq)}\right]_{(eq)} \quad \text{(F.9)}$$

$$\left[H_2OI^+\right]_{(eq)} \left[I^-\right]_{(eq)} = K_4 \left[I_{2(aq)}\right]_{(eq)} \quad \text{(F.10)}$$

$$\left[H^+\right]_{(eq)} \left[OH^-\right]_{(eq)} = K_5 \quad \text{(F.11)}$$

When Equation (F.10) is solved for $[H_2OI^+]_{(eq)}$ and the result is substituted into Equation (F.3) along with Equation (F.8) for $\left[I_3^-\right]_{(eq)}$ the result is [where subscript (eq) is henceforth suppressed]

$$\Sigma\left[I_{2(aq)}\right] = 0.5\,[I^-] + 1.5K_2\left[I_{2(aq)}\right][I^-] + \left[I_{2(aq)}\right] + 0.5\,[HIO] + 0.5\,\frac{K_4\left[I_{2(aq)}\right]}{[I^-]} \quad \text{(F.12)}$$

The electric charge balance between the reactants and products of Equation (F.6) can be reduced to

$$\left([I^-] - [I^-]_p\right) + [I_3^-] - [H_2OI^+] - [H^+]^* = 0 \quad \text{(F.13)}$$

where the first term in parentheses is $[I^-]$ exclusive of those ions associated with dissolved CsI particles, $[I^-]_p$, which is known and balanced by $[Cs^+]$, and $[H^+]^*$ are protons released by the third reaction in Equation (F.6) (since protons released by the fifth reaction are always balanced by the simultaneous release of $[OH^-]$). Since $[H^+]^*$ must be equal to [HIO] because of the stoichiometry of the third reaction in Equation (F.6), Equation (F.14) reduces to

$$\left([I^-] - [I^-]_p\right) + [I_3^-] - [H_2OI^+] - [HIO] = 0 \quad \text{(F.14)}$$

Equation (F.15) can be written as

$$\left([I^-] - [I^-]_p\right) + K_2\left[I_{2(aq)}\right][I^-] - \frac{K_4\left[I_{2(aq)}\right]}{[I^-]} - [HIO] = 0 \qquad (F.15)$$

by using the same substitutions that were used to reduce Equation (F.3) to Equation (F.12). From Equation (F.9) [HIO] becomes

$$HIO = \frac{K_3\left[I_{2(aq)}\right]}{[H^+][I^-]} \qquad (F.16)$$

where $[H^+]$ is given by

$$\left[H^+\right] = [HIO] + \left(\left[OH^-\right] - \left[OH^-\right]_p\right) \qquad (F.17)$$

The first term on the right-hand side ([HIO]) are protons released by the third reaction in Equation (F.6) and the term in parentheses are protons released by the fifth reaction, which also releases that portion of $[OH^-]$ exclusive of the $[OH^-]$ associated with dissolved CsOH ($[OH^-]_p$, which is known). Substituting Equation (F.11) into Equation (F.17) and rearranging the result yields

$$[H^+]^2 - \left([HIO] - [OH^-]_p\right)[H^+] - K_5 = 0 \qquad (F.18)$$

This quadratic equation for $[H^+]$ has the solution

$$[H^+] = \frac{1}{2}\left[\left([HIO] - [OH^-]_p\right) + \left\{\left([HIO] - [OH^-]_p\right)^2 + 4K_5\right\}\right] \qquad (F.19)$$

which may be substituted into Equation (F.16) to give

$$[HIO] = \frac{2K_3\left[I_{2(aq)}\right]/[I^-]}{\left([HIO] - [OH^-]_p\right) + \left\{\left([HIO] - [OH^-]_p\right)^2 + 4K_5\right\}^{1/2}} \qquad (F.20)$$

Equation (F.20) can be put in the form of a quadratic equation for [HIO] and solved to give

$$[H|O] = \frac{[OH^-]_p + \left\{[OH^-]_p{}^2 + 4\left(K_5 + \frac{K_3\left[I_{2(aq)}\right]}{[I^-]}\right)\right\}^{\frac{1}{2}}}{2\left(1 + \frac{K_5[I^-]}{K_3\left[I_{2(aq)}\right]}\right)} \qquad (F.21)$$

Equation (F.21) can be substituted into Equations (F.12) and (F.15) to yield two equations in two unknowns, $[I^-]$ and $[I_{2(aq)}]$, which may be solved iteratively to determine the desired equilibrium concentrations.

A simplification to the procedure just described arises if $[OH^-]$, including $[OH^-]_p$, is very large (pH > 9 because of large amounts of dissolved CsOH). Then $[OH^-]$ remains essentially constant so that $[H^+]$ is given from Equation (F.11) as

$$[H^+] = \frac{K_5}{[OH^-]_p} \tag{F.22}$$

and Equation (F.15) becomes

$$\left([I^-] - [I^-]_p\right) + K_2\left[I_{2(aq)}\right][I^-] - \frac{K_4\left[I_{2(aq)}\right]}{[I^-]} - \frac{K_3\left[I_{2(aq)}\right]}{[H^+][I^-]} = 0 \tag{F.23}$$

which may be written as a quadratic in $[I^-]$ as follows

$$\left(1 + K_2\left[I_{2(aq)}\right]\right)[I^-]^2 - [I^-]_p[I^-] - \left(K_4 + \frac{K_3}{[H^+]}\right)\left[I_{2(aq)}\right] = 0 \tag{F.24}$$

Equation (F.24) has the solution

$$[I^-] = \frac{[I^-]_p + \left\{[I^-]_p^2 + 4\left(1 + K_2\left[I_{2(aq)}\right]\right)\left(K_4 \frac{K_3}{[H^+]}\right)\left[I_{2(aq)}\right]\right\}^{1/2}}{2\left(1 + K_2\left[I_{2(aq)}\right]\right)} \tag{F.25}$$

in terms of an assumed $[I_{2(aq)}]$. Then [HIO] follows immediately from Equation (F.16) and the assumed $[I_{2(aq)}]$. Now if the resulting $[I^-]$, [HIO] and assumed $[I_{2(aq)}]$ are substituted into the right-hand side of Equation (F.12), the result may be compared to the known value of $\sum\left[I_{2(aq)}\right]$ (obtained from the MELCOR RN data base at the beginning of each system timestep). If the discrepancy is significant, then a new value of $[I_{2(aq)}]$ is assumed and the procedure is repeated until convergence is obtained.

Hence, at each spatial step in the rise of the bubbles from the vent exit region to the pool surface, the equilibrium concentrations of all the species in Equation (F.6) are updated. This is accomplished iteratively (with an error tolerance implemented in sensitivity coefficient array 7159) using the equilibrium constants for each reaction (which are temperature dependent and implemented in sensitivity coefficient array 7159) and requiring conservation of the total iodine mass and electric charge.

References

1. D. A. Powers, J. L. Sprung, and C. D. Leigh, “Isotopes, Elements, and Chemical Classes,” in Fission Product Behavior During Severe LWR Accidents: Recommendations for the MELCOR Code System, Sandia National Laboratories, Albuquerque, NM (Unpublished 1987).

2. D. E. Bennett, SANDIA-ORIGEN User’s Manual, NUREG/CR-0987, SAND79-0299, Sandia National Laboratories, Albuquerque, NM (October 1979).

3. R. M. Ostmeyer, An Approach to Treating Radionuclide Decay Heating for Use in the MELCOR Code System, SAND84-1404, NUREG/CR-4169 (May 1985).

4. M. R. Kuhlman, D. J. Lehmicke, and R. O. Meyer, CORSOR User’s Manual, BMI-2122, NUREG/CR-4173 (March 1985).

5. M. Ramamurthi and M. R. Kuhlman, Final Report on Refinement of CORSOR—An Empirical In-Vessel Fission Product Release Model, Battelle Memorial Institute (October 31, 1990).

6. D. A. Powers, J. E. Brockmann, and A. W. Shiver, Vanesa: A Mechanistic Model of Radionuclide Release and Aerosol Generation During Core Debris Interactions with Concrete, SAND85-1370, NUREG/CR-4308 (September 1985) Draft.

7. F. Gelbard, MAEROS User Manual, SAND80-0822, NUREG/CR-1391 (December 1982).

8. H. Jordan and M. R. Kuhlman, TRAP-MELT2 User’s Manual, BMI-2124, NUREG/CR-4205 (May 1985).

9. B. J. Mason, The Physics of Clouds, Clarendon, Oxford, UK (1971).

10. P. C. Owczarski and K. W. Burk, SPARC-90: A Code for Calculating Fission Product Capture in Suppression Pools, NUREG/CR-5765, PNL-7723, Pacific Northwest Laboratory, Richland, WA (October 1991).

11. S. E. Dingman, A. L. Camp. C. C. Wong, D. B. King, and R. D. Gasser, HECTR Version 1.5 User’s Manual, NUREG/CR-4507, SAND86-0101, Sandia National Laboratories, Albuquerque, NM (February 1986).

12. R. A. Lorenz, E. C. Beahm, and R. P. Wichner, “Review of Tellurium Release Rates from LWR Fuel Elements Under Accident Conditions,” Proceedings International Meeting on Light Water Reactor Severe Accident Evaluation (1983).

13. T. Nakamura and R. A. Lorenz, “Effective Diffusion Coefficients Calculated from ORNL FP Release Test Results,” Oak Ridge National Laboratory Research Paper (April 1989).

14. T. Nakamura and R. A. Lorenz, “A Study of Cesium and Krypton Releases Observed in HI and VI Tests Using a Booth Diffusion Model,” Oak Ridge National Laboratory Research Paper (May 1987).

15. J. A. Gieseke, P. Cybulskis, R. S. Denning, M. R. Kuhlman, K. W. Lee, and H. Chen, Radionuclide Release Under Specific LWR Accident Conditions, BMI-2104 (July 1984).

16. R. K. Cole, Jr., D. P. Kelly, and M. A. Ellis, CORCON-Mod2: A Computer Program for Analysis of Molten-Core Concrete Interactions, SAND84-1246, NUREG/CR-3920, Sandia National Laboratories, Albuquerque, NM (August 1984).

17. I. H. Dunbar and S. N. Ramsdale, “Improvements in the Modeling of Sedimentation and Gravitational Agglomeration,” CSNI Specialists’ Meeting on Nuclear Aerosols in Reactor Safety, Karlsruhe, Germany (September 4-6, 1984).

18. D. C. Williams, K. D. Bergeron, P. E. Rexroth, and J. L. Tills, “Integrated Phenomenological Analysis of Containment Response to Severe Core Damage Accidents,” Progress in Nuclear Energy, Vol. 19 (1987).

19. F. Gelbard, L. A. Mondy, and S. E. Ohrt, “A New Method for Determingin Hydrodynamic Effects on the Collision of Two Spheres,”” J. of Statistical Physics, Vol. 62 (1991).

20. W. C. Hinds, Aerosol Technology, John Wiley & Sons, New York (1982).

21 Davies, C.N., Deposition of aerosols from turbulent flow through pipes.

Proceedings of the Royal Society of London Series A, 289: 235-246 (1966).

22 Wood, N.B., A Simple Method For The Calculation Of Turbulent Deposition To Smooth And Rough Surfaces, Journal of Aerosol Science, 12: 275-290 (1981).

23 B.J. Merrill and D.L. Hagrman, “MELCOR Aerosol Transport Module Modification for NSSR-1”, INEL-96/0081, ITER/US96i/TE/SA-03, 1996.

24 N.E. Bixler, VICTORIA 2.0: A Mechanistic Model for Radionuclide Behavior in a Nuclear Reactor Coolant System Under Sever Accident Conditions, SAND93-2301, (1998).

25 G.A. Sehmel, Particle Deposition from Turbulent Air Flow, J. Geophysical Research, 75 1766-1781 (1970).

26 Pui, David Y. H. , Romay-Novas, Francisco and Liu, Benjamin Y. H.(1987) Experimental Study of Particle Deposition in Bends of Circular Cross Section, Aerosol Science and Technology, 7: 3, 301 — 315, First published: January 1987.

27 A.R. McFarland, H. Gong, A. Muyshondt, W.B Wente, and N.K. Anand, Aerosol Deposition in Bends with turbulent Flow, Environmental Science and Technology, 31 3371-3377 (1997).

28. L. F. Shampine and H. A. Watts, Practical Solution of Ordinary Differential Equations by Runge-Kutta Methods, SAND76-0585, Sandia National Laboratories, Albuquerque, NM (1976).

29 M. F. Young, Liftoff Model for MELCOR, NUREG/CR-0987, SAND2015-6119, Sandia National Laboratories, Albuquerque, NM (July 2015).

30. J. R. Welty, C. E. Wicks, and R. E. Wilson, Fundamentals of Momentum, Heat, and Mass Transfer, John Wiley & Sons, New York (1984).

31. R. B. Bird, W. E. Stewart, and E. N. Lightfoot, Transport Phenomena, John Wiley & Sons, New York (1960).

32. A. K. Postma, et al., Models for Predicting the Removal of Airborne Contaminants by Reactor Containment Sprays, BNWL-B-417 (1975).

33. A. K. Postma, et al., Technological Bases for Models of Spray Washout of Airborne Contaminants in Containment Vessels, NUREG/CR-0009 (1978).

34. A. E. J. Eggleton, Theoretical Examination of Iodine-Water Partition Coefficients, AERE-R-4887 (1967).

35. A. K. Postma and W. F. Pasedag, A Review of Mathematical Models for Predicting Spray Removal of Fission Products in Reactor Containment Vessels, WASH-1329 (1973).

36. D. R. Grist, Spray Removal of Fission Products in PWR Containments, Report SRD R267, Safety and Reliability Directorate, UKAEA, England (1982).

37. N. A. Fuchs, The Mechanics of Aerosols, Pergamon Press, New York (1964).

38. I. Langmuir, *J. Meteor*. Vol. 5, 175 (1948).

39. T. J. Heames and R. M. Summers, "Fission Product Surface Chemistry Model Preliminary Design Report," (1995).

40. Vine, M.D. and P.N. Clough, "A Modelling Approach for Heterogeneous Chemistry in the VICTORIA Source Term Code," AEA TRS 5082, AEA Safety and Reliability, Culcheth, UK (1991).

41. Sallach, R.A., "Chemical Aspects of Cesium Iodide Interaction in Steam with 304 Stainless Steel and Inconel 600," NUREG/CR-4241, SAND84-0749, Sandia National Laboratories, Albuquerque, NM (1986).

42. Williams, D.A., "Assessment and Extension of the Chemistry Models in VICTORIA," AEA RS 5410, PWR/SATRG/P(93)L96, AEA Reactor Services, Winfrith, UK (1992).

43. Sallach, R.A. et al., "Chemical Interactions of Tellurium Vapors with Reactor Materials," NUREG/CR-2921, SAND82-1145, Sandia National Laboratories, Albuquerque, NM (1984),

44. Sabathier, F., "The Interaction of Cesium Hydroxide Vapour with 304 Stainless Steel and Inconel 600," AEA RS 5164, AEA Technology, Winfrith, UK (1991).

45. Elrick, R.M. et al., "Reaction Between Some Cesium-Iodine Compounds and the Reactor Materials 304 Stainless Steel, Inconel 600 and Silver, Volume 1, Cesium Hydroxide Reactions," SAND83-0395, NUREG/CR-3197, 1 of 3, Sandia National Laboratories, Albuquerque, NM (1984).

46. Pruppacher, H.R. and J.D. Klett, *Microphysics of Clouds and Precipitation*, D. Reidel Publishing Co. Dordrecht, Holland (1980).

47. Gauntt, R.O., et al, "MELCOR Computer Code Reference Manual," NUREG/CR-6119, Vol. 2, Rev. 1 (May 1998).

48. Murata, K.K. et al, “Code Manual for CONTAIN 2.0,” p. 7-20, NUREG/CR-6533, Dec 1997.

49. Cheng, H-W, “A Model for the Removal of Water Droplet Aerosols from a Flashing Jet Impinging onto a Plate”, J. Aerosol Sci. Vol. 31, pp.999-1014 (2000).

50. ANSI (American National Standard), Design Basis for Protection of Light Water Nuclear Power Plants Against Effects of Postulated Pipe Rupture, ANSI/ANS-58.2 (1988).

51. Giralt, F., Chia, C-J, and Trass, O., “Characterization of the Impingement Region in an Axisymmetric Turbulent Jet”, Ind. Eng. Chem. Fundam. Vol. 16, pp.21-28 (1977).

52. L. Soffer et al., “Accident Source Terms for Light Water Nuclear Power Plants,” NUREG-1465 (1995).

53. E. C. Beahm, R.A. Lorenz, and C.F. Weber, “Iodine Evolution and pH Control,” NUREG/CR-5950, TI93-005714 (1992).

54. C. F. Weber, E. C. Beahm, T. S. Kress, “Models of Iodine Behavior in Reactor Containments,” ORNL/TM-12202 (1992).

55. J. N. Butler, *Ionic Equilibrium, a Mathematical Approach*, Addison Wesley (1964).

56. H. Sims et al., “Iodine Code Comparison,” EUR 16507 EN, European Commission, Luxembourg (1995).

57. S. Dickinson and H. E. Sims, “Modifications to the INSPECT Model”, Proc. 4th OECD Workshop on Iodine Chemistry in Reactor Safety, Paul Scherrer Institute, Switzerland (1996).

58. D. A. Powers, R. K. Cole, and T. J. Heames, “A Simplified Model of Iodine Chemistry for MELCOR,” submitted as part of the Preliminary Design Report for Iodine Aqueous Chemistry: Task 1.2 W6203, Sandia National Laboratories, Albuquerque, NM (1998).

59. K. H. Haskell, W. H. Vandevender, and W. L. Walton, “The SLATEC Common Mathematical Subprogram Library: SNLA Implementation,” SAND80-2792, Sandia National Laboratories, Albuquerque, NM (1980).

60. G. V. Buxton et al., *J. Phys. Chem. Ref. Data*, Vol. 17, p. 513 (1988).

61. A. J. Elliott, M. P. Chenier, and D. C. Ouellette, *Canadian J. Chemistry*, Vol. 68, p. 712 (1990).

62. R. A. Barton and H. E. Sims, "A Comparison of the Predictions of the INSPECT Reaction Set with Experimental Data", Proc. Third CSNI Workshop on Iodine Chemistry, p. 346, JAERI-M-92-012. NEA/CSNI/R(91)15, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken, Japan (1992).

63. L. M. Toth, K. D. Pannell, and O. L. Kirkland, "The Aqueous Chemistry of Iodine", Proc. Fission Product Behavior and Source Term Research, NP-4113-SR, American Nuclear Society, Utah (1984).

64. M. Furrer, R. C. Cripps, and E. Frick, "Iodine Severe Accident Behavior Code IMPAIR 2", PSI Bericht Nr. 25, Paul Scherrer Institute, CH-5232 Villigen/PSI, Switzerland (1989).

65. A. A. Frost and R. G. Pearson, *Kinetics and Mechanism*, 2nd Edition, John Wiley & Sons (1961).

66. A. C. Vikis and R. MacFarlane, J. Phys. Chem. Vol. 89, p. 812 (1985).

67. N. H. Sagert, "Radiolysis of Iodine in Moist Air: A Computer Study", Proc. 2nd CSNI Workshop on Iodine Chemistry in Reactor Safety, AECL-9923, CSNI-149, Atomic Energy of Canada Ltd., Toronto, Canada (1989).

68. A. M. Deane, "Organic Iodine Chemistry Relevant to Nuclear Reactors: A Review", AERE-R-12359, Harwell, Oxfordshire, UK (1988).

69. J. C. Wren, G. A. Glowa, and J. M. Ball, "Modeling Iodine Behaviour Using LIRIC 3.0," Proc. Fourth CSNI Workshop on the Iodine Chemistry in Reactor Safety, Wurenlingen, Switzerland, pp.507 – 530 (1996).

70. G. J. Evans et al., "Preliminary Results from the ACE/RTF Tests 1, 2, and 3", ACE-TR-B3 (1991).

71. C. Hueber, M. Lucas, and J. Gauvain, "Validation of the IODE Code on Analytical Experiments", in "Third CSNI Workshop on Iodine Chemistry in Reactor Safety", JAERI-M-92-012 (1991).

72. M. F. Young, and R. O. Gauntt, "Simulation of the ISP41 Iodine Test Problem with the MELCOR Iodine Model", in ISP-41 Containment Iodine Computer Code Exercise

Based on a Radioiodine Test Facility Experiment, NEA/CSNI/R(2000)6/Vol.1 and 2 (1999).

73. L. N. Kmetyk, MELCOR 1.8.1 Assessment: Marviken-V Aerosol Transport Tests ATT-2b/ATT-4, SAND92-2243, Sandia National Laboratories, Albuquerque, NM (January 1993).

Containment Sprays (SPR) Package

The Containment Sprays (SPR) package models the heat and mass transfer between spray water droplets and the containment building atmosphere. The SPR package models were extracted from the HECTR 1.5 code.

This reference manual describes the models employed in the SPR package. Detailed descriptions of the user input requirements can be found in the SPR package Users' Guide.

Contents

1. Introduction

Where possible, MELCOR uses a generic building-block approach to modeling engineered safety features (ESFs) through use of control volumes, flow paths, heat structures, and control functions. However, for containment sprays, separate models tailored to this system have been implemented in MELCOR.

The MELCOR Containment Sprays (SPR) package models the heat and mass transfer resulting from operation of containment spray systems. The removal of fission product vapors and aerosols by ESFs is modeled within the RadioNuclide (RN) package. See the RN package Reference Manual for details on this modeling.

2. Model Description

The Containment Sprays (SPR) package models the heat and mass transfer between spray droplets and the containment building atmosphere. The modeling in the SPR package was taken virtually intact from the HECTR 1.5 code [1], following the recommendations of the MELCOR phenomena assessment on modeling containment spray systems [2]. The model assumes, among other things, that spray droplets are spherical and isothermal and that they fall through containment atmospheres at their terminal velocity with no horizontal velocity components.

An arbitrary number of spray sources may be placed at various heights in any containment control volume. The source of water for each spray may be associated with the pool in any CVH control volume or it may be left unspecified. If a CVH pool is specified as the spray source reservoir, then input ("dryout" pool depth) may be specified to determine whether there is sufficient water in the pool to permit spray operation. Input (resumption pool depth) may also be specified to determine when spray operation may resume following "dryout". If the pool depth for spray source resumption exceeds the pool depth for spray source "dryout", then there is hysteresis in the spray operation curve that prevents excessive cycling between episodes of spray operation. In a special application, the spray model also receives water from the Heat Structures (HS) package film-tracking model whenever rain from inverted HS surfaces enters the containment atmosphere.

For each spray source, except for sources associated with rain from the HS film-tracking model, the user must specify an initial droplet temperature and flow rate, each of which may be controlled by a control function. The user may turn the sprays on and off with a separate control function for each spray source. A droplet size distribution also may be input for each spray source. In other words, the spray droplets for each source may be divided into a number of different size bins, with individual drops representing the average droplet size being tracked during their fall through the control volume; the total heat and mass transfer for the spray source is obtained by summing the heat and mass transfer calculated for each size.

For each droplet type in each control volume, the following differential equations are solved to determine the heat and mass transfer rates and the terminal fall velocity as a function of drop size:

$$\frac{dm}{dt} = -2\pi\ \rho_g D\left(1 + 0.25\, Re^{1/2}\, S\, c^{1/3}\right) D_c\, ln(1+B) \tag{2-1}$$

$$\frac{dT}{dt} = \frac{1}{m\ c_{pl}}\left[\frac{c_{pv}(T - T_{cv})}{(1+B)^{1/Le} - 1} + h_{fg}\right]\frac{dm}{dt} \tag{2-2}$$

$$\frac{dz}{dt} = \left[\frac{4\left(\rho_d - \rho_g\right) g\ D}{3\rho_g C_d}\right]^{1/2} \tag{2-3}$$

In these equations, the terms are defined as

m = droplet mass, kg,

T, T$_{cv}$ = droplet, control volume atmosphere temperatures, K,

z = droplet fall height, m,

ρ_d, ρ_g = droplet, atmosphere densities, kg/m^3,

c$_{pl}$ = droplet specific heat capacity, J/kg-K

c$_{pv}$ = control volume atmosphere specific heat capacity, J/kg-K

h$_{fg}$ = latent heat of vaporization, J/kg,

D = droplet diameter, m

Re = Reynolds number, dimensionless,

Sc = Schmidt number, dimensionless,

Le = Lewis number, dimensionless,

D$_c$ = diffusion coefficient, m^2/s

C$_d$ = drag coefficient, dimensionless

and *B* is the mass transfer driving force

$$B = \frac{x_b - x_i}{x_i - 1} \tag{2-4}$$

where x_b and x_i are H_2O mass fractions in the bulk atmosphere and at the liquid-gas interface (corresponding to saturation). Equations (2-1) through (2-4) are based on forced convection heat transfer and evaporation and condensation correlations that have been formulated specifically for high temperature atmospheres, such as might be encountered during a hydrogen burn [3]. The constants in Equation (2-1) have been implemented in sensitivity coefficient array 3001.

Correlations for the drag coefficient of spheres, C_d, are used for the following Reynolds number regimes, with the various constants implemented in sensitivity coefficient array C3000:

$$c_d = 27\,Re^{-0.84} \quad \text{for Re < 78} \tag{2-5}$$

$$c_d = 0.271\,Re^{0.217} \quad \text{for 78 < Re < 10000} \tag{2-6}$$

$$c_d = 2 \quad \text{for 10,000 < Re} \tag{2-7}$$

The transfer rates given by Equations (2-1) through (2-3) are integrated by a Runge-Kutta method over the fall height of the spray droplet to obtain the final droplet mass and temperature. By comparing the droplet mass and temperature at the bottom of the compartment to the inlet conditions, the heat transfer and mass transfer to a given droplet are computed. Total heat and mass transfer rates are calculated by multiplying the rates for one droplet by the total number of droplets of that size and summing over all droplet sizes. It is assumed that this total heat and mass transfer rate is constant over a given timestep, and it is also assumed that the containment atmosphere conditions do not change significantly during the fall time of the drop.

The user can describe how droplets falling from one control volume are to be carried over to lower volumes. A control volume may be designated as the containment spray sump. Droplets leaving designated control volumes and not carried over to other volumes are placed in the pool of the sump. Droplets reaching the bottom of a control volume and not being carried over to other volumes or placed in the sump are put into the pool of the control volume.

It should be noted that the SPR package does not model interactions between spray droplets and other structures (nor does any other MELCOR package). Thus, it is not possible to model either core sprays or steam generator auxiliary feed water sprays properly using the SPR package.

The SPR package is coupled to the RadioNuclide (RN) package for the calculation of aerosol washout and atmosphere decontamination by sprays. Current limitations of this interface require some restrictions on the input to the SPR package to avoid nonphysical results associated with multiple calculations in the same control volume. When the SPR and RN packages are both active, the user should limit the spray input so that only one

spray train passes through each control volume and only a single drop size is used in this spray train.

References

1. S. E. Dingman, A. L. Camp, C. C. Wong, D. B. King, and R. D. Gasser, HECTR Version 1.5 User's Manual, NUREG/CR-4507, SAND86-0101, Sandia National Laboratories, Albuquerque, NM (February 1986).

2. G. G. Weigand, ed., Thermal-Hydraulic Process Modeling in Risk Analysis: An Assessment of the Relevant Systems, Structures, and Phenomena, SAND84-1219, NUREG/CR-3986, Sandia National Laboratories, Albuquerque, NM (August 1984).

3. F. A. Williams, Combustion Theory, Addison-Wesley, Reading, MA (1965).