INL REPORT

INL/EXT-14-31366 (Revision 3)
Unlimited Release
Printed March 2018

RELAP-7 Theory Manual

R.A. Berry, L. Zou, H. Zhao, H. Zhang,
J.W. Peterson, R.C. Martineau, S.Y. Kadioglu, D. Andrs, J.E. Hansel

Prepared by
Idaho National Laboratory
Idaho Falls, Idaho 83415

The Idaho National Laboratory is a multiprogram laboratory operated by
Battelle Energy Alliance for the United States Department of Energy
under DOE Idaho Operations Office. Contract DE-AC07-05ID14517.

Issued by the Idaho National Laboratory, operated for the United States Department of Energy by Battelle Energy Alliance.

INL/EXT-14-31366 (Revision 3)
Unlimited Release
Printed March 2018

RELAP-7 Theory Manual

R.A. Berry, L. Zou, H. Zhao, H. Zhang,
J.W. Peterson, R.C. Martineau, S.Y. Kadioglu, D. Andrs, J.E. Hansel

Contents

Figures

Tables

Summary

The RELAP-7 code is the next generation nuclear reactor system safety analysis code being developed at the Idaho National Laboratory (INL). The code is based on the INL's modern scientific software development framework, MOOSE (Multi-Physics Object Oriented Simulation Environment). The overall design goal of RELAP-7 is to take advantage of the previous thirty years of advancements in computer architecture, software design, numerical integration methods, and physical models. The end result will be a reactor systems analysis capability that retains and improves upon RELAP5's and TRACE's capabilities and extends their analysis capabilities for all reactor system simulation scenarios.

RELAP-7 is a new project started in Fiscal Year 2012. It will become the main reactor systems simulation toolkit for the LWRS (Light Water Reactor Sustainability) program's RISMC (Risk Informed Safety Margin Characterization) effort and the next generation tool in the RELAP reactor safety/systems analysis application series. The key to the success of RELAP-7 is the simultaneous advancement of physical models, numerical methods, and software design while maintaining a solid user perspective. Physical models include both PDEs (Partial Differential Equations) and ODEs (Ordinary Differential Equations) and experimental based closure models. RELAP-7 utilizes well-posed governing equations for compressible two-phase flow, which can be strictly verified in a modern verification and validation effort. Closure models used in RELAP5 and newly developed models will be reviewed and selected to reflect the progress made during the past three decades and provide a basis for the closure relations that will be required in RELAP-7. RELAP-7 uses modern numerical methods, which allow implicit time integration, second-order schemes in both time and space, and strongly coupled multi-physics.

RELAP-7 is written with object oriented programming language C++. By using the MOOSE development environment, the RELAP-7 code is developed by following the same modern software design paradigms used for other MOOSE development efforts. The code is easy to read, develop, maintain, and couple with other codes. Most importantly, the modern software design allows the RELAP-7 code to evolve efficiently with time. MOOSE is an HPC development and runtime framework for solving computational engineering problems in a well planned, managed, and coordinated way. By leveraging millions of lines of open source software packages, such as PETSC (a nonlinear solver developed at Argonne National Laboratory) and LibMesh (a Finite Element Analysis package developed at University of Texas), MOOSE reduces the expense and time required to develop new applications. MOOSE provides numerical integration methods and mesh management for parallel computation. Therefore RELAP-7 code developers have been

able to focus more upon the physics and user interface capability. There are currently over 20 different MOOSE based applications ranging from 3-D transient neutron transport, detailed 3-D transient fuel performance analysis, to long-term material aging. Multi-physics and multi-dimensional analysis capabilities, such as radiation transport and fuel performance, can be obtained by coupling RELAP-7 and other MOOSE-based applications through MOOSE and by leveraging with capabilities developed by other DOE programs. This allows restricting the focus of RELAP-7 to systems analysis type simulations and gives priority to retain and significantly extend RELAP5's and TRACE's capabilities.

During the Fiscal Year 2012, MOOSE was extended to better support system analysis code development. The software structure for RELAP-7 had been designed and developed. Numerical stability schemes for single-phase flow, which are needed for continuous finite element analysis, have been developed. Major physical components have been completed (designed and tested) to support a proof of concept demonstration of RELAP-7. The case selected for initial demonstration of RELAP-7 was the simulation of a two-loop, steady state PWR system. During Fiscal Year 2013, both the homogeneous equilibrium two-phase flow model and the seven-equation two-phase flow model have been implemented into RELAP-7. A number of physical components with two-phase flow capability have been developed to support the simplified boiling water reactor (BWR) station blackout (SBO) analyses. The demonstration case includes the major components for the primary system of a BWR, as well as the safety system components for reactor core isolation cooling (RCIC) and the wet well of a BWR containment. The homogeneous equilibrium two-phase flow model was used in the simplified BWR SBO analyses. During Fiscal Year 2014, more detailed implementation of the physical models as well as the code performance improvements associated with the seven-equation two-phase flow model were carried out in order to demonstrate more refined BWR SBO analyses with more realistic geometries.

During Fiscal Year 2015 and 2016 (to date) the ability to use realistic equations of state based on the IAPWS-95 formulation for water/steam, using a numerically efficient Spline-Based Table Look-up approach, were incorporated. Also incorporated with this approach was an extension to include the metastable states needed by the 7-equation nonequilibrium two-phase model used by RELAP-7. An improved entropy viscosity method was implemented for solution stabilization. New and improved boundary conditions for both single-phase and nonequilibrium, 7-equation, two-phase flows, consistent with the method of characteristics, were included. Constitutive equations were added which depend upon the phase's topological sizes and arrangements, e.g. interfacial area concentration and its effects, wall friction and heat transfer, interfacevfriction and heat transfer, etc. Currently, the topology-dependent closures are limited to pre-CHF, vertical flows, but extensions to

CHF and to horizontal flows are ongoing. This revision (Revision 2) of the RELAP-7 Theory Manual is expanded to describe these new features in detail.

In summary, the MOOSE based RELAP-7 code development is an ongoing effort. The MOOSE framework enables rapid development of the RELAP-7 code. The developmental efforts and results demonstrate that the RELAP-7 project is on a path to success. This theory manual documents the main features implemented into the RELAP-7 code. Because the code is an ongoing development effort, this RELAP-7 Theory Manual will evolve with periodic updates to keep it current with the state of the development, implementation, and model additions/revisions.

1 Introduction

The RELAP-7 (Reactor Excursion and Leak Analysis Program) code is the next generation nuclear reactor system safety analysis code being developed at Idaho National Laboratory (INL). The code is based on the INL's modern scientific software development framework MOOSE (Multi-Physics Object Oriented Simulation Environment) [5]. The overall design goal of RELAP-7 is to take advantage of the previous thirty years of advancements in computer architecture, software design, numerical integration methods, and physical models. The end result will be a reactor systems analysis capability that retains, and improves upon, RELAP5's [6] and TRACE's [2] abilities and extends the analysis capability for all reactor system simulation scenarios.

The RELAP-7 project, which began in Fiscal Year 2012, will become the main reactor systems simulation toolkit for LWRS (Light Water Reactor Sustainability) program's RISMC (Risk Informed Safety Margin Characterization) effort and the next generation tool in the RELAP reactor safety/systems analysis application series. The key to the success of RELAP-7 is the simultaneous advancement of physical models, numerical methods, and software design while maintaining a solid user perspective. Physical models include both PDEs (Partial Differential Equations) and ODEs (Ordinary Differential Equations) and experimental based closure models. RELAP-7 will utilize well-posed governing equations for two-phase flow, which can be strictly verified in a modern verification and validation effort. Closure models used in RELAP5, TRACE, and other newly developed models will be reviewed and selected to reflect the progress made during the past three decades and provide a basis for the closure relations that will be required in RELAP-7. RELAP-7 uses modern numerical methods, which allow implicit time integration, second-order schemes in both time and space, and strongly coupled multi-physics.

MOOSE is INL's development and runtime framework for solving computational engineering problems in a well planned, managed, and coordinated way. By using the MOOSE development environment, the RELAP-7 code is developed by following the same modern software design paradigms used for other MOOSE development efforts. The code is easy to read, develop, maintain, and couple with other codes. Most importantly, the modern software design allows the RELAP-7 code to evolve efficiently with time. MOOSE provides numerical integration methods and mesh management for parallel computation. Therefore RELAP-7 code developers need primarily to focus upon the physics and user interface capability.

There are currently over 20 different MOOSE based applications ranging from 3-D

transient neutron transport, detailed 3-D transient fuel performance analysis, to long-term material aging. The advantage of multi-physics and multi-dimensional analyses capabilities, such as radiation transport and fuel performance, can be obtained by coupling RELAP-7 and other MOOSE-based applications (through MOOSE) and by leveraging with capabilities developed by other DOE programs. This allows restricting the focus of RELAP-7 to systems analysis-type simulations and gives priority to retain, and significantly extend RELAP5's capabilities.

Because RELAP-7 is an ongoing development effort, this theory manual will evolve with periodic updates to keep it current with the state of the development, implementation, and model revisions. It is noted that, in some instances, the models reported in this initial version of the theory manual cover phenomena which are not yet implemented, for example the species balance equation for two phase flows. But when it made sense to include derivations, which we have already developed, or descriptions of models which are currently ongoing, such as the entropy viscosity method, we have included such.

1.1 RELAP-7 Description of Approach

An overall description of the RELAP-7 architecture, governing theory, and computational approach is first given as an instructive, and executive overview of the RELAP-7 project direction.

1.1.1 Software Framework

MOOSE is INL's development and runtime environment for the solution of multi-physics systems that involve multiple physical models or multiple simultaneous physical phenomena. The systems are generally represented (modeled) as a system of fully coupled nonlinear partial differential equation systems (an example of a multi-physics system is the thermal feedback effect upon neutronics cross-sections where the cross-sections are a function of the heat transfer). Inside MOOSE, the Jacobian-Free Newton Krylov (JFNK) method [7, 8] is implemented as a parallel nonlinear solver that naturally supports effective coupling between physics equation systems (or Kernels). The physics Kernels are designed to contribute to the nonlinear residual, which is then minimized inside of MOOSE. MOOSE provides a comprehensive set of finite element support capabilities (LibMesh [9], a Finite Element library developed at University of Texas) and provides for mesh adaptation and parallel execution. The framework heavily leverages software libraries from DOE

SC and NNSA, such as the nonlinear solver capabilities in either the the Portable, Extensible Toolkit for Scientific Computation (PETSc [10]) project or the Trilinos project [11] (a collection of numerical methods libraries developed at Sandia National Laboratory). Argonne's PETSc group has recently joined with the MOOSE team in a strong collaboration wherein they are customizing PETSc for our needs. This collaboration is strong enough that Argonne is viewed as a joint developer of MOOSE.

A parallel and tightly coordinated development effort with the RELAP-7 development project is the Reactor Analysis Virtual control ENvironment (RAVEN). This MOOSE-based application is a complex, multi-role software tool that will have several diverse tasks including serving as the RELAP-7 graphical user interface, using RELAP-7 to perform RISMC focused analysis, and controlling the RELAP-7 calculation execution.

Together, MOOSE/RELAP-7/RAVEN comprise the systems analysis capability of the LWRS RISMC ToolKit.

1.2 Governing Theory

The primary basis of the RELAP-7 governing theory includes 7-equation two-phase flow, reactor core heat transfer, and reactor kinetics models. While RELAP-7 is envisioned to incorporate both single and two-phase coolant flow simulation capabilities encompassing all-speed and all-fluids, the main focus in the immediate future of RELAP-7 development is LWRs. Thus, the flow summary is restricted to the two-phase flow model.

1.2.1 7-Equation Two-Phase Model

To simulate light water (nuclear) reactor safety and optimization scenarios there are key issues that rely on in-depth understanding of basic two-phase flow phenomena with heat and mass transfer. Within the context of these two-phase flows, two bubble-dynamic phenomena boiling (or heterogeneous boiling) and flashing or cavitation (homogeneous boiling), with bubble collapse, are technologically very important. The main difference between boiling and flashing is that bubble growth (and collapse) in boiling is inhibited by limitations on the heat transfer at the interface, whereas bubble growth (and collapse) in flashing is limited primarily by inertial effects in the surrounding liquid. The flashing process tends to be far more explosive (or implosive), and is more violent and damaging (at least in the near term) than the bubble dynamics of boiling. However, other problematic phenomena,

such as departure from nucleate boiling (DNB) and CRUD deposition, are intimately connected with the boiling process. Practically, these two processes share many details, and often occur together.

The state of the art in two-phase modeling exhibits a lack of general agreement amongst the so-called experts even regarding the fundamental physical models that describe the complex phenomena. There exist a large number of different models: homogeneous models, mixture models, two-fluid models, drift-flux models, etc. The various models have a different number of variables, a different number of describing equations, and even the definition of the unknowns varies with similar models. There are conservative formulations, non-conservative formulations, models and techniques for incompressible flows and also for compressible flows. Huge Mach number variations can exist in the same problems (Mach number variations of 0.001 to over 100 with respect to mixture sound speed) high-speed versus low-speed gives way to the need for all-speed. In their recent compilation [12], Prosperetti and Tryggvason made important statements that have generally been given insufficient attention in the past: "uncertainties in the correct formulation of the equations and the modeling of source terms may ultimately have a bigger impact on the results than the particular numerical method adopted." "Thus, rather than focusing on the numeric alone, it makes sense to try to balance the numerical effort with expected fidelity of the modeling"..."The formulation of a satisfactory set of average-equations models emerges as the single highest priority in the modeling of complex multiphase flows."

Because of the expense of developing multiple special-purpose simulation codes (at both the system and the detailed multi-dimensional level) and the inherent inability to couple information from these multiple, separate length- and time-scales, efforts at the INL have been focused toward development of multi-scale approaches to solve those multiphase flow problems relevant to light water reactor (LWR) design and safety analysis. Efforts have been aimed at developing well-designed unified physical/mathematical and high-resolution numerical models for compressible, all-speed multiphase flows spanning: (1) well-posed general mixture level (true multiphase) models for fast transient situations and safety analysis, (2) DNS (Direct Numerical Simulation)-like models to resolve interface level phenomena like flashing and boiling flows, and critical heat flux determination, and (3) multi-scale methods to resolve (1) and (2) automatically, depending upon specified mesh resolution, and to couple different flow models (single-phase, multiphase with several velocities and pressures, multiphase with single velocity and pressure, etc.). In other words, we are extending the necessary foundations and building the capability to simultaneously solve fluid dynamic interface problems as well as multiphase mixtures arising from boiling, flashing of superheated liquid, and bubble collapse, etc. in LWR systems. Our ultimate goal is to provide models that, through coupling of system level

and multi-dimensional detailed level codes, resolve interfaces for larger bubbles (DNS-like) with single velocity, single pressure treatment (interface capturing) and average (or homogenize) the two-phase flow field for small bubbles with two-velocity, two-pressure with well-posed models.

The primary, enabling feature of the INL (Idaho National Laboratory) advanced multi-scale methodology for multiphase flows involves the way in which we deal with multi-phase mixtures. This development extends the necessary foundations and builds the capability to simultaneously solve fluid dynamic interface problems as well as multiphase mixtures arising from boiling, flashing or cavitation of superheated liquid, and bubble collapse, etc. in light water reactor systems. Our multi-scale approach is essentially to solve the same equations everywhere with the same numerical method (in pure fluid, in multi-velocity mixtures, in artificially smeared zones at material interfaces or in mixture cells, in phase transition fronts and in shocks). Some of the advantages of this approach include: coding simplicity and robustness as a unique algorithm is used, conservation principles are guaranteed for the mixture, interface conditions are perfectly matched, and the ability to include the dynamic appearance/disappearance of interfaces. This method also allows the coupling of multi-velocities, multi-temperature mixtures to macroscopic interfaces where a single velocity must be present. This entails development on two main fronts. The first requires the derivation (design) of theoretical models for multiphase and interfacial flows whose mathematical description (equation system) is well-posed and exhibits hyperbolicity, exhibiting correct wave dynamics at all scales. The second requires the design of appropriate numerical schemes to give adequate resolution for all spatial and time scales of interest.

Because of the broad spectrum of phenomena occurring in light water nuclear reactor coolant flows (boiling, flashing, and bubble collapse, choking, blowdown, condensation, wave propagation, large density variation convection, etc.) it is imperative that models accurately describe compressible multiphase flow with multiple velocities, and that the models be well-posed and unconditionally hyperbolic. The currently popular state of the art two-phase models assume the pressures in each phase are equal, i.e. they are single pressure models, referred to herein as the "classical" 6-equation model. This approach leads to a system of equations that is ill-posed, not hyperbolic, and it has imaginary characteristics (eigenvalues) that give the wrong wave dynamics. The classical 6-equation model is inappropriate for transient situations and it is valid only for flows dominated by source terms. Numerical methods for obtaining the solution of the 6-equation model rely on dubious properties of the numerical scheme (for example truncation error induced artificial viscosity) to render them numerically well-posed over a portion of the computational spectrum. Thus they cannot obtain grid-converged solutions (the truncation error

goes down thus the artificial viscosity diminishes and the ill-posed nature returns). This calls into question the possibility of obtaining "verification", and thus, "validation" (what does it mean to validate a model that cannot be verified?).

To meet this criterion, we have adopted the 7-equation two-phase flow model [13–15]. This equation system meets our requirements, as described above it is hyperbolic, well-posed, and has a very pleasing set of genuinely nonlinear and linearly degenerate eigenvalues . This 7-equation system is being implemented in RELAP-7, via the INL MOOSE (Multi-physics, Object Oriented Simulation Environment) finite element framework, through a 7-step progression designed to go successively from single-phase compressible flow in a duct of spatially varying cross-sectional area to the compressible, two-phase flow with full thermodynamic and mechanical nonequilibrium. This same 7-equation model, along with its reduced subsystems, is being utilized as described above to build Bighorn, the next generation 3-D high-resolution, multiscale two-phase solver. This will give a unique capability of consistently coupling the RELAP-7 system analysis code to our multi-dimensional, multi-scale, high-resolution multiphase solver and the other MOOSE-based fuels performance packages.

There is yet another benefit to this approach alluded to above with the mention of reduced subsystems of the 7-equation model. Because of the way the 7-equation system for two-phase flow is constructed, it can evolve to a state of mechanical equilibrium (phasic pressure and velocity equilibrium) whereby a very nice 5-equation system results, and even further to thermodynamic equilibrium (phasic temperature and Gibb's energy equilibrium) whereby the classical 3-equation homogeneous equilibrium model (HEM) results. The rate at which these various equilibrium states are reached can be allowed to occur naturally or they can be controlled explicitly to produce a locally reduced model (reduced subsystem) to couple/patch with simpler models. For example this reduction method enables the coupling of zones in which total or partial nonequilibrium effects are present to zones evolving in total equilibrium; or it can be used to examine the admissible limits of a physical system because all limited models are included in this general formulation.

1.2.2 Core Heat Transfer and Reactor Kinetics

The nuclear reaction that takes place within the reactor core generates thermal energy inside the fuel. Also, the passive solid structures, such as piping and vessel walls and the internal vessel structures, represent significant metal masses that can store and release large amounts of thermal energy depending on the reactor fluid (coolant) temperature. The

RELAP-7 code must calculate the heat conduction in the fuel and the metal structures to simulate the heat-transfer processes involved in thermal-energy transport. Therefore, in addition to the two-phase fluid dynamics model described above, RELAP-7 necessarily simulates the heat transfer process with reactor kinetics as the heat source. The heat-conduction equation for cylindrical or slab geometries is solved to provide thermal history within metal structures such as fuel and clad. The volumetric power source in the heat conduction equation for the fuel comes from the point kinetics model with thermal hydraulic reactivity feedback considered [16]. The reactor structure is coupled with the thermal fluid through energy exchange (conjugate heat transfer) employing surface convective heat transfer [17] within the fluid . The fluid, heat conduction, conjugate heat transfer and point kinetics equations may be solved in a fully coupled fashion in RELAP-7 in contrast to the operator-splitting or loose coupling approach used in the existing system safety analysis codes. For certain specific transients, where three-dimensional neutronics effects are important (i.e., rod ejection), three-dimensional reactor kinetics capabilities are available through coupling with the RattleSNake [18] code. RattleSNake is a reactor kinetics code with both diffusion and transport capabilities being developed at INL based on the MOOSE framework.

1.3 Computational Approach

Stated previously, the MOOSE framework provides the bulk of the ”heavy lifting” available to MOOSE-based applications with a multitude of mathematical and numerical libraries. For RELAP-7, LibMesh [9] provides the second-order accurate spatial discretization by employing linear basis, one-dimensional finite elements. The Message Passing Interface (MPI, from Argonne National Laboratory) provides for distributed parallel processing. Intel Threading Building Blocks (Intel TBB) allows parallel C++ programs to take full advantage of multicore architecture found in most large-scale machines of today. PETSc (from Argonne), Trilinos (from Sandia), and Hypre [19] (from Lawrence Livermore National Laboratory) provide the mathematical libraries and nonlinear solver capabilities for JFNK. In MOOSE, a stiffly-stable, second-order backward difference (BDF2) formulation is used to provide second-order accurate time integration for strongly coupled physics in JFNK.

With the objective of being able to handle the flow all-fluids at all-speeds, RELAP-7 is also being designed to handle any systems-level transient imaginable. This can cover the typical design basis accident scenarios (on the order of one second, or less, time scales) commonly associated with RELAP5 simulations to reactor core fuel burnup simulations

(on the order of one year time scales). Unfortunately, the JFNK algorithm can be inefficient in both of these time scale extremes. For short duration transients, typically found in RELAP5 simulations, the JFNK approach requires a significant amount of computational effort be expended for each time step. If the simulation requires short time steps to resolve the physics coupling, JFNK may not be necessary to resolve the nonlinear coupling. The Pressure-Corrected Implicit Continuous-fluid Eulerian (PCICE) algorithm [20, 21] is an operator-split semi-implicit time integration method that has some similarities with RELAP5's time integration method but does not suffer from phase and amplitude errors, given a stable time step. Conversely for very long duration transients, JFNK may not converge for very large time steps as the method relies on resolving the nonlinear coupling terms, and thus, may require an initial estimate of the solution close to the advanced solution time which maybe unavailable. Recently, INL LDRD funds have been directed toward developing a point implicit time integration method for slow transient flow problems [22]. These topics may be addressed in future versions of RELAP-7. Thus, a three-level time integration approach is being pursued to yield an all-time scale capability for RELAP-7. The three integration approaches are described as follows:

1. The JFNK method easily allows implicit nonlinear coupling of dependent physics under one general computational framework. Besides rapid (second-order) convergence of the iterative procedure, the JFNK method flexibly handles multiphysics problems when time scales of different physics are significantly varied during transients. The key feature of the JFNK method is combining Newton's method to solve implicit nonlinear systems with Krylov subspace iterative methods. The Krylov methods do not require an explicit form of the Jacobian, which eliminates the computationally expensive step of forming Jacobian matrices (which also may be quite difficult to determine analytically), required by Newton's method. The matrix-vector product can be approximated by the numerical differentiation of nonlinear residual functions. Therefore, JFNK readily integrates different physics into one solver framework.

2. The PCICE computational fluid dynamics (CFD) scheme, developed for all-speed compressible and nearly incompressible flows, improves upon previous pressure-based semi-implicit methods in terms of accuracy and numerical efficiency with a wider range of applicability. The PCICE algorithm is combined with the Finite Element Method (FEM) spatial discretization scheme to yield a semi-implicit pressure-based scheme called the PCICE-FEM scheme. In the PCICE algorithm, the total energy equation is sufficiently coupled to the pressure Poisson equation to avoid iteration between the pressure Poisson equation and the pressure-correction equations.

Both the mass conservation and total energy equations are explicitly convected with the time-advanced explicit momentum. The pressure Poisson equation then has the time-advanced internal energy information it requires to yield an accurate implicit pressure. At the end of a time step, the conserved values of mass, momentum, and total energy are all pressure-corrected. As a result, the iterative process usually associated with pressure-based schemes is not required. This aspect is highly advantageous when computing transient flows that are highly compressible and/or contain significant energy deposition, chemical reactions, or phase change.

3. Semi-implicit methods can step over certain fine time scales (i.e., ones associated with the acoustic waves), but they still have to follow material Courant time stepping criteria for stability or convergence purposes. The new point implicit method is devised to overcome these difficulties [22]. The method treats only certain solution variables at particular nodes in the discretization (that can be located at cell centers, cell edges, or cell nodes) implicitly, and the rest of the information related to same or other variables at other nodes are handled explicitly. The point-wise implicit terms are expanded in Taylor series with respect to the explicit version of the same terms, at the same locations, resulting in a time marching method that is similar to the explicit methods and, unlike the fully implicit methods, does not require implicit iterations. This new method shares the characteristics of the robust implementation of explicit methods and the stability properties of the unconditionally stable implicit methods. This method is specifically designed for slow transient flow problems wherein, for efficiency, one would like to perform time integrations with very large time steps. Researchers at the INL have found that the method can be time inaccurate for fast transient problems, particularly with larger time steps. Therefore, an appropriate solution strategy for a problem that evolves from a fast to a slow transient would be to integrate the fast transient with a semi-implicit or implicit nonlinear technique and then switch to this point implicit method as soon as the time variation slows sufficiently. A major benefit of this strategy for nuclear reactor applications will reveal itself when fast response coolant flow is coupled to slow response heat conduction structures for a long duration, slow transient. In this scenario, as a result of the stable nature of numerical solution techniques for heat conduction one can time integrate the heat part with very large (implicit) time steps.

Because it is the only integration/solution approach currenly implemented in RELAP-7, only the JFNK method will be discussed subsequently in this report.

2 Single-Phase Thermal Fluids Models

2.1 Flow Model

RELAP-7 treats the basic pipe, duct, or channel flow component as being one dimensional with a cross-sectional area that varies along its length. In this section the instantaneous, area-averaged balance equations are derived to approximate the flow physics. This derivation will begin with a three dimensional local (point-wise), instantaneous statement of the balance equations. For economy of derivation these local balance equations are represented in generic form. The area-averaged balance equations will then be derived from this local generic form, from which the specific area averaged mass, momentum, energy, and entropy equations will be given.

A *local generic transport equation* can be stated as

$$\frac{\partial}{\partial t}(\rho\psi) + \nabla \cdot (\rho\psi \boldsymbol{u}) + \nabla \cdot \boldsymbol{J} - \rho\phi = 0 \tag{1}$$

where ρ is the local material mass density, $\boldsymbol{u}$ is the local material velocity, and ψ, $\boldsymbol{J}$, and ϕ are generic "place holder" variables that can take on different meanings to represent different physical balance equations. To represent balance of mass, momentum, energy, and entropy these generic variables take on the meaning of the variables shown in Table 1. Notice that these variables can take on scalar, vector, or second order tensor character as needed in the equation of interest. In particular, the symbol $\boldsymbol{J}$ is used to represent either a vector or tensor, depending on the equation in question.

Table 1. Balance Equation Variable Definitions.

Balance Equation	ψ	$\boldsymbol{J}$	ϕ
mass	1	0	0
momentum	$\boldsymbol{u}$	$p\boldsymbol{I} - \boldsymbol{\tau}$	$\boldsymbol{g}$
total energy	E	$\boldsymbol{q} + p\boldsymbol{I} \cdot \boldsymbol{u} - \boldsymbol{\tau} \cdot \boldsymbol{u}$	$\boldsymbol{g} \cdot \boldsymbol{u} + \frac{r}{\rho}$
entropy	s	$\frac{1}{T}\boldsymbol{q}$	$\frac{1}{\rho}\Delta$

It is assumed that an instantaneous section of the variable duct can be represented as shown in Figure 1. It is necessary to introduce specific forms of the Leibnitz and Gauss

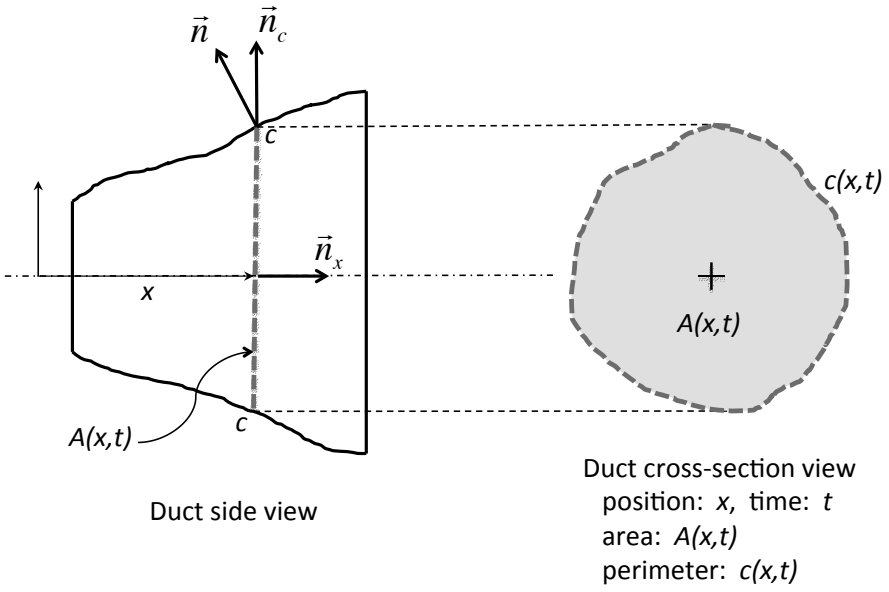

Figure 1. Diagram showing the variable-area duct used in the derivation of the governing equations.

rules, or theorems, from advanced calculus that are specialized to the specific geometry of Figure 1. These rules will be used as tools to shorten the derivations. First, the "Leibnitz Rule" states:

$$\frac{\partial}{\partial t}\int_{A(x,t)} f(x,y,z,t)\,\mathrm{d}A = \int_{A(x,t)} \frac{\partial f}{\partial t}\,\mathrm{d}A + \int_{c(x,t)} f\boldsymbol{u}_w \cdot \hat{\boldsymbol{n}}\,\mathrm{d}s \tag{2}$$

where

$$\mathrm{d}s \equiv \frac{\mathrm{d}c}{\hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{n}}_c} \tag{3}$$

and $\boldsymbol{u}_w$ is the velocity of the (possibly) moving wall. Next, the "Gauss Theorem" is given by

$$\int_{A(x,t)} \nabla \cdot \boldsymbol{B}\,\mathrm{d}A = \frac{\partial}{\partial x}\int_{A(x,t)} \boldsymbol{B} \cdot \hat{\boldsymbol{n}}_x\,\mathrm{d}A + \int_{c(x,t)} \boldsymbol{B} \cdot \hat{\boldsymbol{n}}\,\mathrm{d}s \tag{4}$$

For brevity, in the following derivations we shall suppress the explicit dependence on (x,t) of the area A and boundary c in the relevant integrals. Integrating the local, instantaneous relation (1) over A, gives

$$\int_A \frac{\partial}{\partial t}(\rho\psi)\ \mathrm{d}A + \int_A \nabla\cdot\rho\psi\boldsymbol{u}\ \mathrm{d}A + \int_A \nabla\cdot\boldsymbol{J}\ \mathrm{d}A - \int_A \rho\phi\ \mathrm{d}A = 0. \tag{5}$$

Applying the the Leibnitz and Gauss rules listed above to this equation results in

$$\frac{\partial}{\partial t}A\langle\rho\psi\rangle_A + \frac{\partial}{\partial x}A\langle\rho\psi\boldsymbol{u}\cdot\hat{\boldsymbol{n}}_x\rangle_A + \frac{\partial}{\partial x}A\langle\boldsymbol{J}\cdot\hat{\boldsymbol{n}}_x\rangle_A - A\langle\rho\phi\rangle_A = -\int_c(\dot{m}\psi + \boldsymbol{J}\cdot\hat{\boldsymbol{n}})\ \mathrm{d}s \tag{6}$$

where

$$\langle f\rangle_A \equiv \frac{1}{A}\int_A f(x,y,z,t)\ \mathrm{d}A \tag{7}$$

$$\dot{m} \equiv \rho(\boldsymbol{u}-\boldsymbol{u}_w)\cdot\hat{\boldsymbol{n}}. \tag{8}$$

Finally, because the walls are impermeable and $\boldsymbol{u}\cdot\hat{\boldsymbol{n}}|_c = \boldsymbol{u}_w\cdot\hat{\boldsymbol{n}}|_c$, Equation (6) reduces to

$$\frac{\partial}{\partial t}A\langle\rho\psi\rangle_A + \frac{\partial}{\partial x}A\langle\rho\psi\boldsymbol{u}\cdot\hat{\boldsymbol{n}}_x\rangle_A + \frac{\partial}{\partial x}A\langle\boldsymbol{J}\cdot\hat{\boldsymbol{n}}_x\rangle_A - A\langle\rho\phi\rangle_A = -\int_c \boldsymbol{J}\cdot\hat{\boldsymbol{n}}\ \mathrm{d}s. \tag{9}$$

This is the *instantaneous, area-averaged generic balance equation*.

2.1.1 Field Equations

To obtain mass, momentum, energy, and entropy forms, the variables from Table 1 are substituted into the instantaneous, area-averaged generic balance equation to produce the respective balance equations. The conservation of mass equation is given by:

$$\frac{\partial}{\partial t}A\langle\rho\rangle_A + \frac{\partial}{\partial x}A\langle\rho u\rangle_A = 0 \tag{10}$$

where $u = \boldsymbol{u}\cdot\hat{\boldsymbol{n}}_x$ is the x-component of velocity. The momentum balance equation is:

$$\begin{aligned}\frac{\partial}{\partial t}A\langle\rho\boldsymbol{u}\rangle_A + \frac{\partial}{\partial x}A\langle\rho u\boldsymbol{u}\rangle_A - A\langle\rho\boldsymbol{g}\rangle_A \\ + \frac{\partial}{\partial x}A\langle p\hat{\boldsymbol{n}}_x - \boldsymbol{\tau}\cdot\hat{\boldsymbol{n}}_x\rangle_A = \int_c(-p\boldsymbol{I}\cdot\hat{\boldsymbol{n}} + \boldsymbol{\tau}\cdot\hat{\boldsymbol{n}})\ \mathrm{d}s\end{aligned} \tag{11}$$

where $\boldsymbol{I}$ is the identity tensor. To reduce this equation further, note that

$$\frac{\partial A}{\partial x} = -\int_c \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{n}}_x \, \mathrm{d}s \tag{12}$$

Now take the projection of the momentum equation along the duct axis, i.e. take the scalar product of this equation with $\hat{\boldsymbol{n}}_x$, and use identity (12) to get the final version of the instantaneous, area averaged momentum balance equation

$$\frac{\partial}{\partial t} A\langle \rho u \rangle_A + \frac{\partial}{\partial x} A\langle \rho u^2 \rangle_A + \frac{\partial}{\partial x} A\langle p \rangle_A - \frac{\partial}{\partial x} A\langle (\boldsymbol{\tau} \cdot \hat{\boldsymbol{n}}_x) \cdot \hat{\boldsymbol{n}}_x \rangle_A = \tilde{p} \frac{\partial A}{\partial x} + A\langle \rho g_x \rangle_A + \int_c (\boldsymbol{\tau} \cdot \hat{\boldsymbol{n}}) \cdot \hat{\boldsymbol{n}}_x \, \mathrm{d}s \tag{13}$$

where g_x is the component of gravity along the duct axis and $\tilde{p}$ is the average pressure around curve c on the wall, which can generally differ from $\langle p \rangle_A$. Here the term which accounts for deviations of the wall pressure from this mean wall pressure has been neglected, i.e. the local wall pressure has been assumed constant along c giving $\tilde{p}(x,t)$; the deviatoric term could be included if a higher order approximation is warranted. In the past, the average wall pressure has typically been assumed equal to the area averaged pressure, i.e. $\tilde{p}(x,t) = \langle p \rangle_A$. More will be said of this later. The total energy conservation equation is

$$\frac{\partial}{\partial t} A\langle \rho E \rangle_A + \frac{\partial}{\partial x} A\langle \rho E \boldsymbol{u} \cdot \hat{\boldsymbol{n}}_x \rangle_A + \frac{\partial}{\partial x} A\langle (\boldsymbol{q} + p\boldsymbol{I} \cdot \boldsymbol{u} - \boldsymbol{\tau} \cdot \boldsymbol{u}) \cdot \hat{\boldsymbol{n}}_x \rangle_A - A\langle \rho \boldsymbol{g} \cdot \boldsymbol{u} \rangle_A - A\left\langle \rho \frac{r}{\rho} \right\rangle_A = -\int_c (\boldsymbol{q} + p\boldsymbol{I} \cdot \boldsymbol{u} - \boldsymbol{\tau} \cdot \boldsymbol{u}) \cdot \hat{\boldsymbol{n}} \, \mathrm{d}s \tag{14}$$

or, as is typically done, by assuming the shear stress terms are small enough to be neglected in the total energy equation

$$\frac{\partial}{\partial t} A\langle \rho E \rangle_A + \frac{\partial}{\partial x} A\langle \rho E u \rangle_A + \frac{\partial}{\partial x} A\langle q_x + pu \rangle_A - A\langle \rho \boldsymbol{g} \cdot \boldsymbol{u} \rangle_A - A\langle r \rangle_A = -\int_c p\boldsymbol{u} \cdot \hat{\boldsymbol{n}} \, \mathrm{d}s - \int_c \boldsymbol{q} \cdot \hat{\boldsymbol{n}} \, \mathrm{d}s \tag{15}$$

where $E = e + \frac{\boldsymbol{u} \cdot \boldsymbol{u}}{2}$ is the specific total energy and e is the specific internal energy. This equation can be reduced further by noting the identity

$$\frac{\partial A}{\partial t} = \int_c \boldsymbol{u}_w \cdot \hat{\boldsymbol{n}} \, \mathrm{d}s \tag{16}$$

Again, because $\boldsymbol{u}\cdot\hat{\boldsymbol{n}}|_c = \boldsymbol{u}_w\cdot\hat{\boldsymbol{n}}|_c$, the identity (16) allows the energy equation to be finally written as

$$\frac{\partial}{\partial t}A\langle\rho E\rangle_A + \frac{\partial}{\partial x}A\langle\rho E u\rangle_A + \frac{\partial}{\partial x}A\langle q_x + pu\rangle_A - A\langle\rho\boldsymbol{g}\cdot\boldsymbol{u}\rangle_A - A\langle r\rangle_A = -\tilde{p}\frac{\partial A}{\partial t} - \int_c \boldsymbol{q}\cdot\hat{\boldsymbol{n}}\,\mathrm{d}s \tag{17}$$

where the last term on the right hand side is the net heat transfer from the fluid to the duct wall. The entropy inequality relation is next written as an *equality* (an entropy production equation) as:

$$\frac{\partial}{\partial t}A\langle\rho s\rangle_A + \frac{\partial}{\partial x}A\langle\rho s u\rangle_A + \frac{\partial}{\partial x}A\left\langle\frac{q_x}{T}\right\rangle_A - A\langle\Delta\rangle_A = -\int_c \frac{\boldsymbol{q}}{T}\cdot\hat{\boldsymbol{n}}\,\mathrm{d}s \tag{18}$$

where the last term on the right hand side is the entropy flux due to heat transfer to the duct wall and Δ is the *entropy production* per unit volume due to the process being irreversible.

With this form of the balance equations a closure equation will need to be supplied describing how the local cross-sectional area will change, both spatially and temporally, e.g. stretching or expanding due to pressure. Also, the usual assumption is made (though not necessarily accurate) that the covariance terms of the averaging process are negligible, i.e. if $f = \langle f\rangle_A + f'$ and $g = \langle g\rangle_A + g'$ then

$$\langle fg\rangle_A = \langle f\rangle_A\langle g\rangle_A + \underbrace{\langle f'g'\rangle_A}_{=0} = \langle f\rangle_A\langle g\rangle_A, \tag{19}$$

wherein the notational simplification $\langle f\rangle_A \Rightarrow f$ can be utilized. With this assumption the *mass*, *momentum*, *total energy*, and *entropy balances* can be respectively written as

$$\frac{\partial\rho A}{\partial t} + \frac{\partial\rho u A}{\partial x} = 0 \tag{20}$$

$$\frac{\partial\rho u A}{\partial t} + \frac{\partial\left(\rho u^2 A + pA\right)}{\partial x} = \tilde{p}\frac{\partial A}{\partial x} - F^{\text{friction}}_{\text{wall}} - F^{\text{form}} \tag{21}$$

$$\frac{\partial\rho E A}{\partial t} + \frac{\partial(\rho E + p)uA}{\partial x} = -\tilde{p}\frac{\partial A}{\partial t} + Q^{\text{wall}} \tag{22}$$

$$\frac{\partial\rho s A}{\partial t} + \frac{\partial\rho s u A}{\partial x} + \frac{\partial}{\partial x}\left(\frac{q_x A}{T}\right) - A\Delta = \frac{Q^{\text{wall}}}{\tilde{T}} \tag{23}$$

where the $F^{\text{friction}}_{\text{wall}}$ is the average duct wall shear force (friction) per unit length, F^{form} is the form loss force per unit length, Q^{wall} is the linear heat transfer rate ([W/m]) *from the duct*

wall to the fluid and $\tilde{T}$ is the average fluid temperature along the line c on the duct wall. Also note that in writing the momentum equation (21) the last term on the left hand side of the momentum equation (13) has been neglected as being insignificant. Of course, if the duct wall is rigid, the cross-sectional area is not a function of time and is a function of spatial position only; i.e. $A = A(x)$ only, and $\frac{\partial A}{\partial t} = 0$.

2.2 Constitutive Models

2.2.1 Wall Friction Factor Model

The Fanning friction factor f_{Fanning} is defined as the ratio of shear stress to kinetic energy density:

$$f_{\text{Fanning}} = \frac{\tau}{\frac{1}{2}\rho u^2}\,, \tag{24}$$

and the Darcy-Weisbach friction factor is related to the Fanning friction factor as follows:

$$f_{\text{Darcy}} = 4 f_{\text{Fanning}}\,. \tag{25}$$

Multiplying the shear stress by the surface area S gives the shear force due to friction. Also consider the definition of hydraulic diameter:

$$d_h = \frac{4A}{P_{\text{wet}}}\,, \tag{26}$$

where P_{wet} is the "wetted perimeter", i.e., the perimeter of the flow channel in contact with the fluid. Then, noting that the surface area can be related as $S = P_{\text{wet}} L$, the friction force per unit length can be expressed as the following, where for the remainder of this section, f represents the Darcy friction factor:

$$F_{\text{wall}}^{\text{friction}} = \frac{f}{2d_h}\rho u\,|u|\,A\,. \tag{27}$$

Because of its dependencies, f is usually a function of x, along with the other flow variables. Furthermore, in the case of a variable-area duct or pipe, both the cross-sectional area and the wetted perimeter are functions of x, and therefore d_h is also a function of x. In the particular case of a pipe with circular cross section and radius $r(x)$, we have $A = \pi r^2$, $P_{\text{wet}} = 2\pi r$, and consequently

$$d_h = 2r(x) = 2\sqrt{\frac{A}{\pi}} \tag{28}$$

so that (27) becomes

$$F_{\text{wall}}^{\text{friction}} = \frac{f}{4}\rho u\,|u|\,\sqrt{\pi A} \tag{29}$$

This relationship simply states that the wall shear force due to the fluid flow is proportional to the bulk kinetic energy of the flow.

Currently, the same wall friction factor model is used for single-phase flow as that used in RELAP5 [23]. The friction factor model is simply an interpolation scheme linking the laminar, laminar-turbulent transition, and turbulent flow regimes. The wall friction model consists of four regions which are based on the Reynolds number (Re):

1. $f = f_{\text{max}}$ for $0 \leq Re < 64$.
2. Laminar flow for $64 \leq Re < 2200$.
3. Transitional flow for $2200 \leq Re < 3000$.
4. Turbulent flow for $Re \geq 3000$.

where Re is defined as

$$Re = \frac{\rho|u|d_h}{\mu} \tag{30}$$

where μ is the fluid viscosity, which in general depends on the fluid temperature. The laminar friction factor depends on the cross-sectional shape of the channel and assumes steady state and fully-developed flow (and a variety of other assumptions). It is defined as

$$f = \frac{64}{Re\Phi_S}, \qquad 64 \leq Re < 2200 \tag{31}$$

where Φ_S is a user-defined shape factor for noncircular flow channels, and has a value of 1 for circular pipes. For the transition from laminar to turbulent flow, a reciprocal interpolation method is employed. This choice is motivated by the form of (31), and is valid over the region $Re_{\text{min}} \equiv 2200 \leq Re \leq Re_{\text{max}} \equiv 3000$. Solving for the parameter N in the relation

$$\frac{N}{Re_{\text{min}}} - \frac{N}{Re_{\text{max}}} = 1 \tag{32}$$

yields

$$N = \frac{Re_{\text{max}}Re_{\text{min}}}{Re_{\text{max}} - Re_{\text{min}}}. \tag{33}$$

The reciprocal weighting function w is then defined as

$$w = \frac{N}{Re_{\text{min}}} - \frac{N}{Re} \tag{34}$$

and varies from 0 to 1 as the Reynolds number varies from Re_{min} to Re_{max}. Finally, the transition friction factor formula is defined as

$$f = (1 - w) f_{\text{lam},Re_{\text{min}}} + w f_{\text{turb},Re_{\text{max}}}. \tag{35}$$

Formula (35) is valid for $2200 \leq Re \leq 3000$, $f_{\text{lam},Re_{\text{min}}}$ is the laminar friction factor at Re_{min}, and $f_{\text{turb},Re_{\text{max}}}$ is the turbulent friction factor at Re_{max}. The turbulent friction factor is given by a Zigrang-Sylvester approximation [24] to the Colebrook-White correlation [25], for $Re \geq 3000$:

$$\frac{1}{\sqrt{f}} = -2 \log_{10} \left\{ \frac{\epsilon}{3.7D} + \frac{2.51}{Re} \left[1.14 - 2 \log_{10} \left(\frac{\epsilon}{D} + \frac{21.25}{Re^{0.9}} \right) \right] \right\} \tag{36}$$

where ϵ is the surface roughness, D is the pipe diameter, and the factor 1.14 corrects the value of 1.114 present in the original document.

2.2.2 Distributed Form Loss Model

Form losses, such as those those arising from flow obstructions or pipe bends, can be accounted for using the concept of a loss coefficient K, which is the fraction of dynamic pressure head that is lost due to a flow feature i:

$$(\Delta p)_i = K_i \left(\frac{1}{2} \rho u^2 \right)_i . \tag{37}$$

Suppose a pipe/channel segment has a known form pressure loss over its length L but that the point-wise form losses are not known. In this case, it is useful to use a *linear* form loss coefficient K', meaning a form loss coefficient per unit length:

$$\Delta p = \int_0^L K' \left(\frac{1}{2} \rho u^2 \right) dx \, , \tag{38}$$

where in general K' can be a function of space, though without knowing spatial loss information, one may assume a uniform form loss distribution:

$$K' = \frac{K}{L} \, . \tag{39}$$

The resulting linear force expression can then be expressed similarly to that of the friction factor:

$$F^{\text{form}} = K' \left(\frac{1}{2} \rho u \left| u \right| \right) A \,. \tag{40}$$

2.2.3 Convective Heat Transfer Model

The general form of the convective heat transfer term in (22) is

$$Q_{\text{wall}} = H_w a_w \left(T_{\text{wall}} - T \right) A \tag{41}$$

where a_w is the so-called heat transfer area density, H_w is the convective wall heat transfer coefficient, $T_{\text{wall}} = T_{\text{wall}}(x,t)$ is the average temperature around perimeter $c(x,t)$, and $T = T(x,t)$ is the area average bulk temperature of the fluid for cross-section at (x,t). In the constant-area case, the heat transfer area density is roughly defined as:

$$a_w \equiv \lim_{\Delta x \to 0} \frac{\text{wetted area of pipe section of length } \Delta x}{\text{volume of pipe section of length } \Delta x} \tag{42}$$

For a constant-area pipe with radius r and circular cross-section, formula (42) yields

$$a_w = \lim_{\Delta x \to 0} \frac{2\pi r \Delta x}{\pi r^2 \Delta x} = \frac{2}{r} \tag{43}$$

For a variable-area duct or pipe, if we consider the "projected area" through which heat transfer can occur, we observe that the rate of change of the pipe's area, $\frac{\partial A}{\partial x}$, also plays a role (though it may be neglected). If we wish to account for the rate of change of the pipe's area, in (41) we can set

$$a_w A \Delta x \equiv \text{“projected area of a pipe segment of length } \Delta x\text{”} \tag{44}$$

and then take the limit as $\Delta x \to 0$. The right-hand side of (44) of course depends on the geometric shape of the pipe cross section. For a circular pipe with cross sectional area $A(x)$ and associated radius $r(x)$, the formula for the lateral surface area of a right-circular frustum of height Δx implies that (44) can be written as:

$$a_w A \Delta x = \pi \left(2r + \frac{\partial r}{\partial x} \Delta x \right) \Delta x \sqrt{1 + \left(\frac{\partial r}{\partial x} \right)^2} \tag{45}$$

In the limit as $\Delta x \to 0$, we obtain

$$a_w A = 2\pi r \sqrt{1 + \left(\frac{\partial r}{\partial x}\right)^2} = \sqrt{4\pi A + \left(\frac{\partial A}{\partial x}\right)^2} \tag{46}$$

where (46) arises upon substitution of the cross-sectional area formula for a circle. Note also that we recover

$$a_w A = 2\pi r \tag{47}$$

from (46) in the constant area case. The resulting wall heating term in this case is

$$Q_{\text{wall}} = H_w \left(T - T_{\text{wall}}\right) \left[4\pi A + \left(\frac{\partial A}{\partial x}\right)^2\right]^{\frac{1}{2}} \tag{48}$$

Clearly, pipes with rapidly changing cross-sectional area, i.e. $\frac{\partial A}{\partial x} \gg 1$, have a larger projected area than pipes with slowly-varying cross-sectional areas. Conversely, if the area is not changing rapidly with x, this additional term can safely be neglected.

It is possible to derive an analogous formula to (46) for polygonal cross sections other than circles. For example, for a square cross section with side length $L(x)$, the analog of (45) is

$$a_w A \Delta x = 2\left(2L + \frac{\partial L}{\partial x}\Delta x\right)\Delta x \sqrt{1 + \frac{1}{4}\left(\frac{\partial L}{\partial x}\right)^2} \tag{49}$$

which, as $\Delta x \to 0$ yields,

$$a_w A = 4L \sqrt{1 + \frac{1}{4}\left(\frac{\partial L}{\partial x}\right)^2} = \sqrt{16A + \left(\frac{\partial A}{\partial x}\right)^2} \tag{50}$$

where we have used the relations $A(x) = L^2(x)$, $\frac{\partial A}{\partial x} = 2L\frac{\partial L}{\partial x}$.

Currently, the same wall heat transfer model for single-phase flow is used as in RELAP5 [26]. The convective heat transfer coefficient is determined by many factors, i.e., hydraulic geometry, fluid types, and several Buckingham π-group dimensionless numbers. For single-phase, different flow regimes can be involved, including laminar forced convection, turbulent forced convection, and natural convection. For the current version, all the heat transfer models are based on steady-state and fully-developed flow assumptions. These assumptions may become questionable, for example, in a short pipe with strong entrance effect. Effects that account for flow regions which are not fully developed will be added in the future.

2.2.3.1 Internal Pipe Flow For internal pipe flow, (the default geometry) the maximum of the forced-turbulent, forced-laminar, and free-convection coefficients is used for non-liquid metal fluids in order to avoid discontinuities in the heat transfer coefficient. The forced laminar heat convection model is an exact solution for fully-developed laminar flow in a circular tube with a uniform wall heat flux and constant thermal properties. The laminar Nusselt number (Nu) is here defined to be

$$Nu = \frac{H_w d_h}{k} = 4.36 \tag{51}$$

where k is the fluid thermal conductivity, based on fluid bulk temperature. The turbulent forced convection model is based on the Dittus-Boelter correlation

$$Nu = CRe^{0.8}Pr^n \tag{52}$$

where $C = 0.023$, Pr is the Prandtl Number, $n = 0.4$ for heating, and $n = 0.3$ for cooling. The applicable ranges and accuracy of the correlation are discussed in Section 4.2.3.1.1 of [26]. The Churchill and Chu Nu-correlation,

$$Nu = \left(0.825 + \frac{0.387 Ra^{\frac{1}{6}}}{\left(1 + \left(\frac{0.492}{Pr}\right)^{\frac{9}{16}}\right)^{\frac{8}{27}}} \right)^2 \tag{53}$$

is used for free convection along a vertical flat plate, where $Ra = GrPr$ is the Rayleigh number. The Grashof number Gr is defined as

$$Gr = \frac{\rho^2 g \beta (T_w - T) L^3}{\mu^2} \tag{54}$$

where β is the coefficient of thermal expansion and L is the natural convection length scale. The default natural convection length scale is the heat transfer hydraulic diameter. For liquid metal fluids (with $Pr < 0.1$), the following correlation is used for all the convective heat transfer regimes:

$$Nu = 5.0 + 0.025 Pe^{0.8} \tag{55}$$

where $Pe = RePr$ is the Peclet number.

2.2.3.2 Vertical Bundles with In-line Rods, Parallel Flow Only The correlations for vertical bundles with in-line rods and parallel flow differs from the default internal pipe flow only in the implementation of the turbulent flow multiplier of Inayatov [27], which

is based on the rod pitch to rod diameter ratio. The pitch is the distance between the centers of the the adjacent rods. If the bundle consists of in-line tubes on a square pitch or staggered tubes on an equilateral triangle pitch, the coefficient C in (52) becomes

$$C = 0.023\frac{P}{D} \tag{56}$$

where P is the pitch and D is the rod diameter. As in RELAP5, if $\frac{P}{D} > 1.6$, then $\frac{P}{D}$ is reset to 1.6. If $\frac{P}{D}$ is not provided, or is less than 1.1, a default value of 1.1 is used. For liquid metals (with $Pr < 0.1$), the following correlation is used for all the convective heat transfer regimes in vertical bundles

$$Nu = 4.0 + 0.33\left(\frac{P}{D}\right)^{3.8}\left(\frac{Pe}{100}\right)^{0.86} + 0.16\left(\frac{P}{D}\right)^{5}. \tag{57}$$

Equation (57) is valid for $1.1 < \frac{P}{D} < 1.4$. If $\frac{P}{D}$ is outside this range, it is "clipped" to either the maximum or minimum value.

2.2.4 Equations of State

In the following sections, we discuss several equations of state employed for the various thermal-fluid models used in RELAP-7. When we say "equation of state," we really mean a so-called "incomplete" equation of state defined by a pair of equations

$$p = p(\rho, e) \tag{58}$$

$$T = T(\rho, e) \tag{59}$$

i.e., both the pressure and the temperature can be computed if the density and internal energy are given. Reformulations of (58) and (59) which consist of two equations relating the four quantities p, T, ρ, and e are also acceptable and useful in practice.

The pair of equations (58) and (59) may be contrasted with the case of a single thermodynamically consistent "complete" equation of state $e = e(\vartheta, s)$ where $\vartheta = 1/\rho$ is the specific volume, and s is the specific entropy. Note that the existence of a complete equation of state implies the existence of an incomplete equation of state through the relations $p = -\left(\frac{\partial e}{\partial \vartheta}\right)_s$, and $T = \left(\frac{\partial e}{\partial s}\right)_\vartheta$, but the converse is not true [28]. The partial derivative notation $\left(\frac{\partial f}{\partial x}\right)_y$ is used to denote the fact that $f = f(x, y)$ and the derivative is taken with respect to x while holding y constant. Solution of the Euler equations requires only an incomplete equation of state (for smooth flows), hence we focus on the form (58)–(59) in the present work. More will be said subsequently, when discussing selection and stabilization of "weak" solutions.

2.2.4.1 Barotropic Equation of State The barotropic equation of state is suitable for a two-equation (isothermal) fluid model. It describes only isentropic (reversible) processes, and implies a constant sound speed. Shocks do not form from initially smooth data in fluids modeled with the barotropic equation of state; discontinuities present in the initial data may be retained and propagated without "sharpening or steepening". This equation of state, described here only for reference because it is used in RELAP-7 primarily for testing and verification purposes, is given by

$$\begin{aligned} p &= p_0 + a^2(\rho - \rho_0) \\ &= p_0 + a^2(U_0 - \rho_0) \end{aligned} \tag{60}$$

where a is a constant, roughly the sound speed. The derivatives of p with respect to the conserved variables are

$$p_{,0} = a^2 \tag{61}$$

$$p_{,1} = 0 \tag{62}$$

$$p_{,2} = 0 \tag{63}$$

2.2.4.2 Isentropic Stiffened Gas Equation of State The isentropic stiffened gas equation of state is more general than the barotropic equation of state. In this equation of state, the pressure and density are related by:

$$\frac{p + p_\infty}{p_0 + p_\infty} = \left(\frac{\rho}{\rho_0}\right)^\gamma \tag{64}$$

which is sometimes rearranged to read:

$$p = (p_0 + p_\infty)\left(\frac{\rho}{\rho_0}\right)^\gamma - p_\infty \tag{65}$$

where p_∞, γ, and ρ_0 are constants which depend on the fluid. Representative values for water are $p_\infty = 3.3 \times 10^8$ Pa, $\gamma = 7.15$, $\rho_0 = 10^3$ kg/m^3. Note that although the symbol γ is used in (65), it should not be confused with the ratio of specific heats (the ratio of specific heats is approximately 1 for most liquids). The isentropic equation of state is, of course, not valid for flows with shocks, but for weak pressure waves and weak shocks the approximation is not bad. The speed of sound in this fluid can be computed as

$$c^2 = \frac{\partial p}{\partial \rho} = \frac{\gamma}{\rho}(p + p_\infty) \tag{66}$$

Hence, unlike the barotropic equation of state, the sound speed of this model varies with the density and pressure values. In terms of conserved variables, we have:

$$p = (p_0 + p_\infty)\left(\frac{U_0}{\rho_0}\right)^\gamma - p_\infty \tag{67}$$

with derivatives

$$p_{,0} = \frac{\gamma}{U_0}(p + p_\infty) = c^2 \tag{68}$$

$$p_{,1} = 0 \tag{69}$$

$$p_{,2} = 0 \tag{70}$$

Finally, we note that Courant and Friedrichs [29] also discuss this equation of state in the form

$$p = A\left(\frac{\rho}{\rho_0}\right)^\gamma - B \tag{71}$$

Approximate values for the constants in (71) are given in Table 2.

Table 2. Constants for Courant and Friedrich's form of the isentropic stiffened gas equation of state.

	SI	Imperial
ρ_0	999.8 kg/m^3	1.94 slug/ft^3
γ	7	
A	3.04076×10^8 Pa	3001 atm
B	3.03975×10^8 Pa	3000 atm

2.2.4.3 Linear Equation of State A more general "linear" equation of state (a straightforward extension of (60)) which takes into account variations in temperature as well as density, is given by

$$p = p_0 + K_\rho(\rho - \rho_0) + K_T(T - T_0) \tag{72}$$

$$e = e_0 + c_v(T - T_0). \tag{73}$$

Since $K_\rho \equiv \left(\frac{\partial p}{\partial \rho}\right)_T$ and $K_T \equiv \left(\frac{\partial p}{\partial T}\right)_\rho$ (evaluated at p_0) are large for liquids (like water), we see that large changes in pressure are required to produce changes in density, assuming T is approximately constant. This observation is in accordance with what we expect for a nearly incompressible fluid. If the working fluid is water, representative values for the constants in (72) and (73) are given in Tables 3 ($p_0 = 1$ MPa) and 4 ($p_0 = 5$ MPa) for several temperatures. The tables demonstrate that the various constants are not strongly dependent on the absolute magnitude of the pressure. These constants are obtained from the thermodynamic data for water available on the NIST website[1].

In terms of conserved variables, (72) and (73) can be written as

$$p = p_0 + K_\rho(U_0 - \rho_0) + \frac{K_T}{c_v}\left(\frac{U_2}{U_0} - \frac{U_1^2}{2U_0^2} - e_0\right) \tag{74}$$

$$T = T_0 + \frac{1}{c_v}\left(\frac{U_2}{U_0} - \frac{U_1^2}{2U_0^2} - e_0\right), \tag{75}$$

and the derivatives of p with respect to the conserved variables are

$$p_{,0} = K_\rho + \frac{K_T}{c_v U_0}\left(\frac{U_1^2}{U_0^2} - \frac{U_2}{U_0}\right) = K_\rho + \frac{K_T}{c_v\rho}\left(u^2 - E\right) \tag{76}$$

$$p_{,1} = -\frac{K_T U_1}{c_v U_0^2} = -\frac{K_T u}{c_v \rho} \tag{77}$$

$$p_{,2} = \frac{K_T}{c_v U_0} = \frac{K_T}{c_v \rho}\ . \tag{78}$$

The derivatives of T with respect to the conserved variables are

$$T_{,0} = \frac{1}{c_v U_0}\left(\frac{U_1^2}{U_0^2} - \frac{U_2}{U_0}\right) = \frac{1}{c_v\rho}\left(u^2 - E\right) \tag{79}$$

$$T_{,1} = -\frac{U_1}{c_v U_0^2} = -\frac{u}{c_v \rho} \tag{80}$$

$$T_{,2} = \frac{1}{c_v U_0} = \frac{1}{c_v \rho}\ . \tag{81}$$

For completeness, the density is given as a function of pressure and temperature, and the

[1] `http://webbook.nist.gov/chemistry/fluid`

temperature as a function of pressure and density, for the linear equation of state:

$$\rho = \rho_0 + \frac{p - p_0}{K_\rho} - \frac{K_T}{K_\rho}(T - T_0) \tag{82}$$

$$T = T_0 + \frac{p - p_0}{K_T} - \frac{K_\rho}{K_T}(\rho - \rho_0)\,. \tag{83}$$

Table 3. Constants for the linear equation of state for $p_0 = 1$ MPa and $T_0 = 375$, 400, 425, and 450K.

$T = 375$K		
p_0	10^6	Pa
K_ρ	2.1202×10^6	Pa-m^3/kg
ρ_0	957.43	kg/m^3
K_T	1.5394×10^6	Pa/K
T_0	375	K
c_v	4.22×10^3	J/kg-K
e_0	4.27×10^5	J/kg

$T = 400$K		
p_0	10^6	Pa
K_ρ	1.9474×10^6	Pa-m^3/kg
ρ_0	937.87	kg/m^3
K_T	1.6497×10^6	Pa/K
T_0	400	K
c_v	4.22×10^3	J/kg-K
e_0	5.32×10^5	J/kg

$T = 425$K		
p_0	10^6	Pa
K_ρ	1.7702×10^6	Pa-m^3/kg
ρ_0	915.56	kg/m^3
K_T	1.6643×10^6	Pa/K
T_0	425	K
c_v	4.22×10^3	J/kg-K
e_0	6.39×10^5	J/kg

$T = 450$K		
p_0	10^6	Pa
K_ρ	1.5552×10^6	Pa-m^3/kg
ρ_0	890.39	kg/m^3
K_T	1.6303×10^6	Pa/K
T_0	450	K
c_v	4.22×10^3	J/kg-K
e_0	7.48×10^5	J/kg

2.2.4.4 Stiffened Gas Equation of State In the single-phase model discussed in this section, the fluid (whether it be liquid or vapor) is compressible and behaves with its own convex equation of state (EOS). For initial development purposes it was decided to use a simple form capable of capturing the essential physics. For this purpose, the stiffened gas equation of state (SGEOS) was selected (LeMetayer et al. [4])

$$p(\rho, e) = (\gamma - 1)\rho(e - q) - \gamma p_\infty \tag{84}$$

Table 4. Constants for the linear equation of state for $p_0 = 5$ MPa and $T_0 = 375$, 400, 425, and 450K.

$T = 375$K		
p_0	5×10^6	Pa
K_ρ	2.1202×10^6	Pa-m^3/kg
ρ_0	959.31	kg/m^3
K_T	1.5559×10^6	Pa/K
T_0	375	K
c_v	4.26×10^3	J/kg-K
e_0	4.25×10^5	J/kg

$T = 400$K		
p_0	5×10^6	Pa
K_ρ	1.9474×10^6	Pa-m^3/kg
ρ_0	939.91	kg/m^3
K_T	1.6406×10^6	Pa/K
T_0	400	K
c_v	4.26×10^3	J/kg-K
e_0	5.31×10^5	J/kg

$T = 425$K		
p_0	5×10^6	Pa
K_ρ	1.7702×10^6	Pa-m^3/kg
ρ_0	917.83	kg/m^3
K_T	1.6659×10^6	Pa/K
T_0	425	K
c_v	4.26×10^3	J/kg-K
e_0	6.37×10^5	J/kg

$T = 450$K		
p_0	5×10^6	Pa
K_ρ	1.5552×10^6	Pa-m^3/kg
ρ_0	892.99	kg/m^3
K_T	1.6370×10^6	Pa/K
T_0	450	K
c_v	4.26×10^3	J/kg-K
e_0	7.46×10^5	J/kg

where p, ρ, e, and q are the pressure, density, internal energy, and the binding energy of the fluid considered. The parameters γ, q, and p_∞ are the constants (coefficients) of each fluid. The parameter q defines the zero point for the internal energy, which will be relevant later when phase transitions are involved with two-phase flows. The parameter p_∞ gives the "stiffened" properties compared to ideal gases, with a large value implying "nearly-incompressible" behavior.

The first term on the right-hand side of (84) is a repulsive effect that is present for any state (gas, liquid, or solid), and is due to molecular motions and vibrations. The second term on the right represents the attractive molecular effect that guarantees the cohesion of matter in the liquid or solid phases. The parameters used in this equation of state are determined by using a reference curve, usually in the $\left(p, \frac{1}{\rho}\right)$ plane. In LeMetayer et al. [4], the saturation curves are utilized as this reference curve to determine the stiffened gas parameters for liquid and vapor phases. The SGEOS is the simplest prototype that contains the main physical properties of pure fluids — repulsive and attractive molecular

effects — thereby facilitating the handling of the essential physics and thermodynamics with a simple analytical formulation. Thus, a fluid, whether liquid or vapor, has its own thermodynamics.

The pressure law, equation (84), is incomplete. A *caloric law* is also needed to relate the fluid temperature to the other fluid properties (for example, $T = T(p, \rho)$) and thereby completely describe the thermodynamic state of the fluid. For the fluid, whether liquid or vapor, it is assumed that the thermodynamic state is determined by the SGEOS as:

$$e(p, \rho) = \frac{p + \gamma p_\infty}{(\gamma - 1)\rho} + q \tag{85}$$

$$\rho(p, T) = \frac{p + p_\infty}{(\gamma - 1)c_v T} \tag{86}$$

$$h(T) = \gamma\, c_v T + q \tag{87}$$

$$g(p, T) = (\gamma c_v - q')T - c_v T \ln \frac{T^\gamma}{(p + p_\infty)^{(\gamma-1)}} + q \tag{88}$$

where T, h, and g are the temperature, enthalpy, and Gibbs free enthalpy, respectively, of the fluid considered. In this system, equation (86) is the caloric law. In addition to the three material constants mentioned above, two additional material constants have been introduced, the constant volume specific heat c_v and the parameter q'. These parameters will be useful when two-phase flows are considered later. The values for water and its vapor from [4] are given in Table 5. These parameter values appear to yield reasonable approximations over a temperature range from 298 to 473 K [4]. Equation (87) can also

Table 5. Stiffened gas equation of state parameters for water and its vapor, from [4].

Water	γ	q (J kg^{-1})	q' (J kg^{-1} K^{-1})	p_∞ (Pa)	c_v (J kg^{-1} K^{-1})
Liquid	2.35	-1167×10^3	0	10^9	1816
Vapor	1.43	2030×10^3	-23×10^3	0	1040

be written as

$$h = c_p\, T + q \tag{89}$$

if we define $c_p = \gamma c_v$. Combining (85) and (86) also allows us to write the temperature as

$$T = \frac{1}{c_v}\left(e - q - \frac{p_\infty}{\rho}\right). \tag{90}$$

In terms of conserved variables, the pressure is given by

$$p = (\gamma - 1)\left(U_2 - \frac{U_1^2}{2U_0} - U_0 q\right) - \gamma p_\infty. \tag{91}$$

The derivatives of p with respect to the conserved variables are

$$p_{,0} = (\gamma - 1)\left(\frac{1}{2}\frac{U_1^2}{U_0^2} - q\right) = (\gamma - 1)\left(\frac{1}{2}u^2 - q\right) \tag{92}$$

$$p_{,1} = (\gamma - 1)\left(-\frac{U_1}{U_0}\right) = (\gamma - 1)(-u) \tag{93}$$

$$p_{,2} = \gamma - 1. \tag{94}$$

In terms of conserved variables, the temperature is given by

$$T = \frac{1}{c_v}\left(\frac{U_2}{U_0} - \frac{U_1^2}{2U_0^2} - q - \frac{p_\infty}{U_0}\right). \tag{95}$$

The derivatives of T with respect to the conserved variables are

$$T_{,0} = \frac{1}{c_v U_0^2}\left(p_\infty + \frac{U_1^2}{U_0} - U_2\right) = \frac{1}{c_v \rho^2}\left(p_\infty + \rho u^2 - \rho E\right) \tag{96}$$

$$T_{,1} = -\frac{U_1}{c_v U_0^2} = -\frac{u}{c_v \rho} \tag{97}$$

$$T_{,2} = \frac{1}{c_v U_0} = \frac{1}{c_v \rho}\ . \tag{98}$$

The sound speed for this equation of state can be computed as

$$\begin{aligned} c^2 &= \frac{p}{\rho^2}(\gamma - 1)\rho + (\gamma - 1)(e - q) \\ &= \gamma\left(\frac{p + p_\infty}{\rho}\right). \end{aligned} \tag{99}$$

2.2.4.5 Ideal Gas Equation of State The ideal gas equation of state is fundamental; many other equations of state are more-or-less based on the ideal gas equation of state in some way. Although RELAP-7 is primarily concerned with flows involving liquids and their vapors, there are certainly nuclear reactor applications, such as helium cooling, where

the ideal gas equation of state is relevant. The pressure and temperature in a (calorically-perfect) ideal gas are given by

$$p = (\gamma - 1)\rho e \tag{100}$$

$$T = \frac{e}{c_v} \tag{101}$$

where $\gamma = \frac{c_p}{c_v}$ is the ratio of specific heats, and c_v is the specific heat at constant volume, which in a calorically-perfect gas is assumed to be constant. This equation of state is a particular form of the stiffened gas equation of state already described in Section 2.2.4.4, with $q = p_\infty = 0$. We therefore omit giving a detailed listing of the derivatives of this equation of state with respect to the conserved variables. The reader should instead refer to Section 2.2.4.4, and the derivatives listed therein.

3 Two-Phase Thermal Fluids Models

3.1 Flow Model

Many important fluid flows involve a combination of two or more materials or phases having different properties. For example, in light water nuclear reactor safety and optimization there are key issues that rely on in-depth understanding of basic two-phase flow phenomena with heat and mass transfer. Within the context of these multiphase flows, two bubble-dynamic phenomena: boiling (heterogeneous) and flashing or cavitation (homogeneous boiling), with bubble collapse, are technologically very important to nuclear reactor systems. The main difference between boiling and flashing is that bubble growth (and collapse) in boiling is inhibited by limitations on the heat transfer at the interface, whereas bubble growth (and collapse) in flashing is limited primarily by inertial effects in the surrounding liquid. The flashing process tends to be far more explosive (and implosive), and is more violent and damaging (at least in the near term) than the bubble dynamics of boiling. However, other problematic phenomena, such as crud deposition, appear to be intimately connected with the boiling process. In reality, these two processes share many details, and often occur together.

The multiple phases or components often exhibit relative motion among the phases or material classes. The microscopic motions of the individual constituents are complex and the detailed solution to the micro-level evolutionary equations is very difficult. Characteristic of such flows of multi-component materials is an uncertainty in the exact locations

of the particular constituents at any particular time. For most practical purposes, it is not possible to exactly predict or measure the evolution of the details of such systems, nor is it even necessary or desirable. Usually, more gross features of the motion, or the "average" behavior of the system are of greater interest. Here we present descriptive equations that will predict the evolution of this averaged behavior. Due to the complexities of interfaces and resultant discontinuities in fluid properties, as well as from physical scaling issues, it is essential to work with averaged quantities and parameters. The rational approach pursued here to examine two-phase flow must be based on the fundamental postulates of continuum mechanics and upon careful use of averaging procedures. We begin by rigorously specifying our concept of an average. There are several types of averaging. The published literature predominantly contains two types of averaging: "volume averaging" [30,31] and "time averaging" [32]. Occasionally variants, such as the "area averaging" described in the single-phase flow section above for one-dimensional variable cross-sectional area, or combinations of the two, such as "volume-time averaging," are used. However, a more general approach (least restrictions) will be utilized here, adopting what is known as "ensemble averaging." The equation forms that result from these different averaging approaches can appear quite similar, though the physical/mathematical interpretation of the various terms are certainly different and there are subtle differences in the inherent restrictions associated with each.

When the physical system has a large amount of variability, a natural interpretation of the meaning of predictions is in terms of expected values and variances. If there are many different events, or "realizations," possible, then the expected value is naturally an "average" over all of these events, or the ensemble of realizations. The ensemble is then the set of all experiments with the same boundary and initial conditions, with some properties that we would like to associate with the mean and distribution of the components and their velocities. A realization of the flow is a possible motion that could have happened. Implicit in this concept is the intuitive idea of a "more likely" and a "less likely" realization in the ensemble. Therefore, as we shall see, each ensemble of realizations, corresponding to a given physical situation, has a probability measure on subsets of realizations. The ensemble average is the generalization of the elementary idea of adding the values of the variable for each realization, and dividing by the number of observations. The ensemble average then allows the interpretation of phenomena in terms of repeatability of multi-component flows.

One of the nice features of ensemble averaging, as opposed to volume averaging, is that ensemble averaging does not require that a control volume contain a large quantity of a particular component in any given realization. Consider the following example, taken directly from Drew and Lahey [33], where the average of a particle-fluid mixture is of

interest. Gas turbines are eroded by particulate matter (or droplets) suspended in the gas stream passing through the inlet and impacting on the various parts of the machine, e.g. the turbine blades. The trajectories of individual particles moving through the gas turbine are very complicated, depending on where and when the particles enter the inlet of the device. Such predictions are usually not required. A prediction, however, that is of interest to the designer is the average, or expected values, of the particle flux (or the concentration and velocities of particles) near parts in the device that are susceptible to erosion. Since the local concentration of particles is proportional to the probability that particles will be at the various points in the device at various times, and the particle velocity field will be the mean velocity that the particles will have if they are at that position in the device, the design engineer will be able to use this information to assess the places where erosion due to particle impact may occur.

It may be that there are no times for which there will be many particles in some representative control volume (or representative elementary volume, REV). So, volume averaging, which depends on the concept of having many representative particles in the averaging volume at any instant, will fail. The appropriateness of ensemble averaging is obvious. Here the ensemble is the set of motions of a single particle through the device, given that it started at a random point at the inlet at a random time during the transient flow through the device. Clearly the solution for the average concentration and average velocity gives little information about the behavior of a single particle in the device; however, the information is very appropriate for assessing the probability of damage to the device. Similar examples could be given where time averaging will fail, but where ensemble averaging is again appropriate. The ensemble average is more fundamental than either time or volume averaging. In fact, both time and volume averaging can be viewed as approximations to the ensemble average, which can be justified, respectively, for steady or homogeneous flow [34].

3.1.1 Ensemble Averaging

A general method is presented here, based on the ensemble averaging concept [34–38] for developing averaged balance or conservation equations for multiple materials, any one of which may be at point $\boldsymbol{x}$, at a given instant t. With this procedure, the most likely state at a point, i.e. the *expected value*, will be determined simultaneously with which material is most likely to be found at that point. Imagine running an experiment many times and collecting data about the state of the flow at each point $\boldsymbol{x}$ and time t. This information could include which material or phase is present, material density, velocity,

pressure, temperature, concentration, etc. From this information, one can compute the *ensemble average*. The ensemble average of a generic property Q_0 of a fluid or material in a process is an average over the realizations

$$\langle Q_0 \rangle(\boldsymbol{x}, t) = \frac{1}{N_R} \sum_{r=1}^{N_R} Q_{0,r}(\boldsymbol{x}, t) \tag{102}$$

where N_R is the number of times the process or experiment is repeated, and is a large number. Now imagine that many of the realizations are near duplicates, i.e. they are essentially the same state, with N occurrences. We can then rewrite the sum over the realizations as a sum over the number of states N_R

$$\begin{aligned} \langle Q_0 \rangle(\boldsymbol{x}, t) &= \frac{1}{N_R} \sum_{r=1}^{N_R} N(\boldsymbol{x}, t, \Gamma) Q_0(\Gamma) \\ &= \sum_{r=1}^{N_\Gamma} \frac{N(\boldsymbol{x}, t, \Gamma)}{N_R} Q_0(\Gamma) \\ &= \int_{\text{all } \Gamma} Q_0(\Gamma) f(\boldsymbol{x}, t, \Gamma) \, \mathrm{d}\Gamma \end{aligned} \tag{103}$$

where $f(\boldsymbol{x}, t, \Gamma) = \frac{N(\boldsymbol{x}, t, \Gamma)}{N_R}$ is the probability of the state Γ in the ensemble. Note that in the limit of an infinite number of repetitions of the experiment, with a sum over all of the states, the summation is replaced with an integral form in the definition of the ensemble average. More correctly, because $\int_{\text{all } \Gamma} f(\boldsymbol{x}, t, \Gamma) \, \mathrm{d}\Gamma = 1$, $f(\boldsymbol{x}, t, \Gamma)$ is referred to as the *probability density*.

The state is the full thermodynamic/kinematic description of the matter at a point $\boldsymbol{x}$ and time t; for example, the set

$$\Gamma = \begin{bmatrix} \rho_0, \boldsymbol{u}_0, h_0, p_0, \tau_0, \dots \\ \rho_0^1, \boldsymbol{u}_0^1, h_0^1, \rho_0^2, \boldsymbol{u}_0^2, h_0^2, \dots \\ X_1, X_2, \dots \end{bmatrix} \tag{104}$$

where the various symbols used in (104) are described in Table 6, and

$$\rho_0 = \sum_s \rho_0^s \tag{105}$$

$$\rho_0 \boldsymbol{u}_0 = \sum_s \rho_0^s \boldsymbol{u}_0^s \tag{106}$$

$$\rho_0 h_0 = \sum_s \rho_0^s h_0^s \ . \tag{107}$$

Other properties may also appear in the above thermodynamic/kinematic state such as the phase or material temperature, θ_0, the phase or material specific internal energy, e_0, and the phase or material specific entropy, s_0.

Table 6. State variable definitions.

Symbol	Description
$X_k(\boldsymbol{x}, t)$	Phase or material indicator function: equal to 1 if material k is present, 0 otherwise
ρ_0	Phase or material density
$\boldsymbol{u}_0$	Phase or material velocity
h_0	Phase or material specific enthalpy
p_0	Pressure
$\boldsymbol{\tau}_0$	Deviatoric stress
ρ_0^s	Species partial density
$\boldsymbol{u}_0^s$	Species velocity
h_0^s	Species partial enthalpy

In a typical multiphase flow, the ensemble averages of interest may include those listed in Table 7. From a physical viewpoint, the bulk average density of a phase represents a summation of all of the density values that occurred for that phase, divided by the total number of experiments run. The bulk average density corresponds intuitively to the idea of the mass of phase per unit volume of mixture, or the observed material density. On the other hand, the intrinsic average density physically corresponds to a summation of all of the density values that occurred for that phase, dividing by the number of times in which that phase occurred in the experiments. The intrinsic average density corresponds intuitively to the idea of the mass of phase per unit volume of phase k , or the true material density. Some researchers prefer to work with bulk average densities, e.g. Kashiwa and Rauenzahn [35], while others prefer working with intrinsic densities, e.g. Drew and Passman [34]. This is mostly an issue of convenience, since one can easily be converted to the other. Here intrinsic averages will be used, and henceforth, when an average is mentioned, mean intrinsic average will be implied unless indicated otherwise.

Table 7. Multiphase flow ensemble averages of interest.

Ensemble Average	Description
$\alpha_k \equiv \langle X_k \rangle$	Material k volume fraction
$\hat{\rho}_k \equiv \langle X_k \rho_0 \rangle$	Material k bulk average density
$\rho_k \equiv \frac{X_k \rho_0}{\alpha_k}$	Material k intrinsic average density
$\hat{\rho}_k^s \equiv \langle X_k \rho_0^s \rangle$	Species s in material k bulk average density
$\rho_k^s \equiv \frac{X_k \rho_0^s}{\alpha_k}$	Species s in material k intrinsic average density
$\boldsymbol{u}_k \equiv \frac{\langle X_k \rho_0 \boldsymbol{u}_0 \rangle}{\hat{\rho}_k} = \frac{\langle X_k \rho_0 \boldsymbol{u}_0 \rangle}{\alpha_k \rho_k}$	Material k velocity
$E_k \equiv \frac{\langle X_k \rho_0 E_0 \rangle}{\hat{\rho}_k} = \frac{\langle X_k \rho_0 E_0 \rangle}{\alpha_k \rho_k}$	Material k total energy
$s_k \equiv \frac{\langle X_k \rho_0 s_0 \rangle}{\hat{\rho}_k} = \frac{\langle X_k \rho_0 s_0 \rangle}{\alpha_k \rho_k}$	Material k entropy
$\boldsymbol{T} \equiv \langle \boldsymbol{T}_0 \rangle$	Mean mixture stress
$\boldsymbol{T}_k \equiv \frac{\langle X_k \boldsymbol{T}_0 \rangle}{\alpha_k}$	Mean k-material stress
$p \equiv \langle p_0 \rangle$	Pressure (single pressure model)
$p_k \equiv \frac{\langle X_k p_0 \rangle}{\alpha_k}$	Pressure in k-material

3.1.2 Field Equations

For a reasonably broad range of conditions (with common substances), the exact balance equations, valid at a point inside each material, are

$$\dot{\rho}_0 = -\rho_0 \nabla \cdot \boldsymbol{u}_0 \tag{108}$$

$$\dot{\rho}_0^s = -\rho_0^s \nabla \cdot \boldsymbol{u}_0 - \nabla \cdot \rho_0^s (\boldsymbol{u}_0^s - \boldsymbol{u}_0) + \dot{r}_0^s \tag{109}$$

$$\rho_0 \dot{\boldsymbol{u}}_0 = \nabla \cdot \boldsymbol{T}_0 + \rho_0 \boldsymbol{g} \tag{110}$$

$$\rho_0 \dot{E}_0 = \nabla \cdot (\boldsymbol{T}_0 \cdot \boldsymbol{u}_0) + \nabla \cdot \boldsymbol{q}_0 + \rho_0 \boldsymbol{g} \cdot \boldsymbol{u}_0 + \rho_0 \varepsilon_0 \tag{111}$$

$$\rho_0 \dot{s}_0 \geqslant \frac{\rho_0 \varepsilon_0}{\theta_0} - \nabla \cdot \left(\frac{\boldsymbol{q}_0}{\theta_0} \right). \tag{112}$$

For these macroscopic balance laws the material derivative has been used, which is defined as

$$\dot{Q}_0 \equiv \frac{\partial Q_0}{\partial t} + \boldsymbol{u}_0 \cdot \nabla Q_0 \ . \tag{113}$$

Let the total variation of f in the phase space $(\boldsymbol{x}, t, \Gamma)$ be given by [35]

$$\frac{\partial f}{\partial t} + \boldsymbol{u}_0 \cdot \nabla f + \dot{\Gamma} \frac{\partial f}{\partial \Gamma} = \frac{df}{dt} = 0 \tag{114}$$

where it is assumed that, as a material point is followed through phase space, its probability of occurrence remains constant. Various moments of this equation can be formed by first multiplying this equation by Q_0, and then averaging this result. It can be shown (see also Kashiwa and Rauenzahn [35], here corrected) that the resulting equation is

$$\frac{\partial}{\partial t}\langle Q_0\rangle + \nabla\cdot\langle Q_0\boldsymbol{u}_0\rangle = \langle \dot{Q}_0 + Q_0\nabla\cdot\boldsymbol{u}_0\rangle \ . \tag{115}$$

This result is called the *moment evolution equation* and the details of its derivation are given in [14, 15]. The averaged balance or conservation equations are obtained by letting the generic Q_0 be replaced by various "meaningful" functions and then by performing judicious manipulations on the equations to bring about physically useful forms of the equation.

3.1.3 Mass Balance

Letting $Q_0 = X_k\rho_0$ in (115) results in

$$\frac{\partial\langle X_k\rho_0\rangle}{\partial t} + \nabla\cdot\langle X_k\rho_0\boldsymbol{u}_0\rangle = \left\langle \dot{X}_k\rho_0 + X_k(\dot{\rho}_0 + \rho_0\nabla\cdot\boldsymbol{u}_0)\right\rangle \ . \tag{116}$$

Introducing the pure material (microscopic) mass balance equation and the definition of average into this equation gives

$$\frac{\partial\alpha_k\rho_k}{\partial t} + \nabla\cdot\alpha_k\rho_k\boldsymbol{u}_k = \left\langle \dot{X}_k\rho_0\right\rangle \ . \tag{117}$$

Because the time- and spatial-derivatives are being taken of functions that are not smooth, this averaged mass balance equation is to be interpreted in the sense of distributions, or generalized functions [39]. To examine the right hand side of this equation in more detail the definition of the material derivative is first considered. It is defined by

$$\dot{X}_k = \frac{\partial X_k}{\partial t} + \boldsymbol{u}_0\cdot\nabla X_k \tag{118}$$

in a generalized function sense. By noting that for points not on the interface where either $X_k = 0$ or $X_k = 1$ the partial derivatives both vanish, while for points on the interface (which also move with the interface velocity) the function X_k is a jump that remains constant so their material derivatives following the interface vanish, it is seen that the material derivative of X_k following the interface vanishes,

$$\frac{\partial X_k}{\partial t} + \boldsymbol{u}_{int}\cdot\nabla X_k = 0 \tag{119}$$

where $\boldsymbol{u}_{int}$ denotes the velocity of an interface of phase or material k. Thus,

$$\left\langle \dot{X}_k \rho_0 \right\rangle = \langle \rho_0 (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) \cdot \nabla X_k \rangle \tag{120}$$

and the averaged mass balance equation becomes

$$\begin{aligned} \frac{\partial \alpha_k \rho_k}{\partial t} + \nabla \cdot \alpha_k \rho_k \boldsymbol{u}_k &= \langle \rho_0 (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) \cdot \nabla X_k \rangle \\ &\equiv \Omega_k^{mass} \ . \end{aligned} \tag{121}$$

Because ∇X_k has the sifting property of the Dirac delta function(al), the only contributors (on the right hand side) are the material interfaces. As shown in [40, 41], ∇X_k is aligned with the surface unit normal vector pointing to phase k, $\nabla X_k = \hat{\boldsymbol{n}}_k \delta(\boldsymbol{x} - \boldsymbol{x}_{int}, t)$. Thus the Ω_k^{mass} represents the flux of mass to phase k from the other phases via the interface, usually just referred to as phase change. With no storage of mass at an interface, mass balance requires further that

$$\sum_{k=1}^{no.\,of\,phases} \Omega_k^{mass} = 0 \ . \tag{122}$$

For later use, it is convenient to introduce the concept of *interfacial area density* of phase or component k, defined as

$$A_k = - \langle \hat{\boldsymbol{n}}_k \cdot \nabla X_k \rangle \tag{123}$$

where $\hat{\boldsymbol{n}}_k$ is the unit exterior normal to phase or component k. A_k is the expected value of the ratio of the interfacial area (in a small volume) to the (small) volume, in the limit as that volume approaches zero.

3.1.4 Generic Balance Equation

To more expeditiously derive the other conservation equations, the averaged balance equation resulting from a generic, microscopic balance equation will be derived first. Then the other balance equations can be found by judicious substitution of pertinent quantities into the generic balance equation. Consider the generic, microscopic balance equation

$$\frac{\partial \rho_0 \psi_0}{\partial t} + \nabla \cdot \rho_0 \psi_0 \boldsymbol{u}_0 = \nabla \cdot J_0 + \rho_0 g_0 \tag{124}$$

or

$$\rho_0 \dot{\psi}_0 = \frac{d(\rho_0 \psi_0)}{dt} + (\rho_0 \psi_0) \nabla \cdot \boldsymbol{u}_0 = \nabla \cdot J_0 + \rho_0 g_0 \ . \tag{125}$$

Equations (124) and (125) hold at each point where sufficient smoothness occurs for the derivatives to be taken, otherwise at simple discontinuities its generic jump balance condition

$$[\![\rho_0\psi_0(\boldsymbol{u}_0 - \boldsymbol{u}_{int}) + J_0]\!] \cdot \hat{\boldsymbol{n}} = m \tag{126}$$

holds, where ψ_0 is the conserved quantity, J_0 is a molecular or diffusive flux, g_0 is a source density, and m is the interfacial source of ψ_0. The notation $[\![\cdot]\!]$ here denotes the jump in the enclosed quantity across an interface. Obviously, these generic quantities must be included in our state space, e.g.

$$\Gamma = \begin{bmatrix} \rho_0, \boldsymbol{u}_0, \psi_0, J_0, \dots \\ X_1, X_2, \dots \end{bmatrix} . \tag{127}$$

Let us also define averages of these quantities as

$$\psi_k \equiv \frac{\langle X_k \rho_0 \psi_0 \rangle}{\alpha_k \rho_k} \tag{128}$$

$$J_k \equiv \frac{\langle X_k J_0 \rangle}{\alpha_k} \tag{129}$$

$$g_k \equiv \frac{\langle X_k \rho_0 g_0 \rangle}{\alpha_k \rho_k} . \tag{130}$$

Letting $Q_0 = X_k \rho_0 \psi_0$ in (115) gives

$$\begin{aligned} \frac{\partial \langle X_k \rho_0 \psi_0 \rangle}{\partial t} + \nabla \cdot \langle X_k \rho_0 \psi_0 \boldsymbol{u}_0 \rangle = \nabla \cdot \langle X_k J_0 \rangle + \langle X_k \rho_0 g_0 \rangle \\ + \langle [\rho_0 \psi_0 (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) - J_0] \cdot \nabla X_k \rangle \ . \end{aligned} \tag{131}$$

Introducing the fluctuating velocity

$$\boldsymbol{u}'_k \equiv \boldsymbol{u}_0 - \boldsymbol{u}_k \tag{132}$$

into this expression finally results in

$$\begin{aligned} \frac{\partial \alpha_k \rho_k \psi_k}{\partial t} + \nabla \cdot \alpha_k \rho_k \psi_k \boldsymbol{u}_k = \nabla \cdot \alpha_k J_k + \nabla \cdot \alpha_k \boldsymbol{J}_k^{Fluct} + \alpha_k \rho_k g_k \\ + \Omega_k^{mass} \psi_k^{int} + \Omega_k^{\psi} \end{aligned}$$

where $\boldsymbol{J}_k^{Fluct} = -\frac{\langle X_k \rho_0 \psi_0 \boldsymbol{u}'_k \rangle}{\alpha_k}$ is the flux of ψ due to fluctuations in the phase k velocity, ψ_k^{int} is the effective value of ψ that is transferred to phase k from the other phases due to mass transfer, or phase change, and Ω_k^{ψ} is a flux of ψ to phase k *not* due to bulk mass

transfer from the other phase(s). This is the *averaged generic balance equation*. To obtain balance at the interface, the generic jump balance equation requires the constraint

$$\sum_{k=1}^{no.\,of\,phases} \Omega_k^{mass}\psi_k^{int} + \Omega_k^{\psi} = M \tag{133}$$

where $M = \langle m \rangle$ is the expected net effect of all the interfacial $\psi - source$ terms. With this generic balance equation, the phasic species mass, momentum, and energy equations, as well as the phasic entropy inequality, can readily be determined.

3.1.5 Species Mass Balance

The microscopic species mass balance equation can be written as

$$\frac{\partial \rho_0^s}{\partial t} + \nabla \cdot \rho_0^s \boldsymbol{u}_0^s = \dot{r}^s \tag{134}$$

where ρ_0^s is the species partial density, $\boldsymbol{u}_0^s$ is the species bulk velocity, and $\dot{r}^s$ is the generation or source of the species due to chemical reactions. The species mass balance equation is not usually written this way because not much is usually known about individual species velocities. Instead, it is usually cast as

$$\frac{\partial \rho_0^s}{\partial t} + \nabla \cdot \rho_0^s \boldsymbol{u}_0 = \nabla \cdot \rho_0^s(\boldsymbol{u}_0 - \boldsymbol{u}_0^s) + \dot{r}^s \tag{135}$$

because of the availability (to a certain extent) of acquired empirical knowledge of the behavior of the first term on the right hand side of this equation (species diffusion). This equation is in the form of the generic balance equation (124) with the assignments of

$$\psi_0 = \frac{\rho_0^s}{\rho_0}, \qquad J_0 = \rho_0 \frac{\rho_0^s}{\rho_0}(\boldsymbol{u}_0 - \boldsymbol{u}_0^s), \qquad g_0 = \frac{\dot{r}^s}{\rho_0}\ . \tag{136}$$

Thus the averaged species mass balance equation takes the form

$$\begin{aligned}\frac{\partial}{\partial t}\langle X_k \rho_0^s \rangle + \nabla \cdot \langle X_k \rho_0^s \boldsymbol{u}_0 \rangle = \nabla \cdot \langle X_k \rho_0^s (\boldsymbol{u}_0 - \boldsymbol{u}_0^s) \rangle + \langle X_k \dot{r}^s \rangle \\ + \langle [\rho_0^s(\boldsymbol{u}_0 - \boldsymbol{u}_{int}) - \rho_0^s(\boldsymbol{u}_0 - \boldsymbol{u}_0^s)] \cdot \nabla X_k \rangle\ .\end{aligned} \tag{137}$$

Again introducing the fluctuating velocity along with the definitions of averaged quantities, the final form of the averaged species mass balance equation is

$$\begin{aligned}\frac{\partial \alpha_k \rho_k^s}{\partial t} + \nabla \cdot \alpha_k \rho_k^s \boldsymbol{u}_k = & \nabla \cdot \langle X_k \rho_0^s (\boldsymbol{u}_0 - \boldsymbol{u}_0^s) \rangle \\ & - \nabla \cdot \langle X_k \rho_0^s \boldsymbol{u}_k' \rangle \\ & + \langle \rho_0^s (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) \cdot \nabla X_k \rangle \\ & - \langle \rho_0^s (\boldsymbol{u}_0 - \boldsymbol{u}_0^s) \cdot \nabla X_k \rangle \\ & + \dot{R}_k^s \end{aligned} \tag{138}$$

where the terms on the right-hand side of (138) are the relative species flux, fluctuational diffusion, phase change, mass exchange, and average generation rate in phase k due to chemical reactions, $\dot{R}_k^s \equiv \frac{\langle X_k \dot{r}^s \rangle}{\alpha_k}$, respectively.

3.1.6 Momentum Balance

The averaged momentum balance equation results from the generic averaged balance equation with the assignments of

$$\psi_0 = \boldsymbol{u}_0, \qquad J_0 = \boldsymbol{T}_0, \qquad g_0 = \boldsymbol{g}_0 \tag{139}$$

to give:

$$\begin{aligned}\frac{\partial \alpha_k \rho_k \boldsymbol{u}_k}{\partial t} + \nabla \cdot \alpha_k \rho_k \boldsymbol{u}_k \otimes \boldsymbol{u}_k = & \nabla \cdot \alpha_k (\boldsymbol{T}_k + \boldsymbol{T}_k^{Fluct}) + \alpha_k \rho_k \boldsymbol{g}_k \\ & + \boldsymbol{\Omega}_k^{mom} + \boldsymbol{u}_k^{int} \Omega_k^{mass} \end{aligned} \tag{140}$$

where the fluctuating stress $\boldsymbol{T}_k^{Fluct}$ and the interfacial momentum source $\boldsymbol{\Omega}_k^{mom}$ are given by

$$\boldsymbol{T}_k^{Fluct} \equiv -\frac{\langle X_k \rho_0 \boldsymbol{u}_k' \otimes \boldsymbol{u}_k' \rangle}{\alpha_k} \tag{141}$$

$$\boldsymbol{\Omega}_k^{mom} \equiv -\langle \boldsymbol{T}_0 \cdot \nabla X_k \rangle \ . \tag{142}$$

The averaged interfacial momentum balance constraint (jump condition) is

$$\boldsymbol{\gamma} = \sum_{k=1}^{no.\,of\,phases} \boldsymbol{\Omega}_k^{mom} + \boldsymbol{u}_k^{int} \Omega_k^{mass} \tag{143}$$

where $\boldsymbol{\gamma}$ is the interfacial momentum source, i.e. surface tension source.

3.1.7 Energy Balance

The assignment of

$$\psi_0 = E_0 = e_0 + \frac{1}{2}\boldsymbol{u}_0 \cdot \boldsymbol{u}_0 \tag{144}$$

$$J_0 = \boldsymbol{T_0} \cdot \boldsymbol{u}_0 + \boldsymbol{q}_0 \tag{145}$$

$$g_0 = \boldsymbol{g}_0 \cdot \boldsymbol{u}_0 + \varepsilon_0 \tag{146}$$

to the variables of the generic averaged balance equation give the averaged energy balance equation

$$\begin{aligned}
&\frac{\partial}{\partial t}\alpha_k\rho_k\left(e_k + \frac{1}{2}\boldsymbol{u}_k \cdot \boldsymbol{u}_k + e_k^{Fluct}\right) + \nabla \cdot \alpha_k\rho_k\boldsymbol{u}_k\left(e_k + \frac{1}{2}\boldsymbol{u}_k \cdot \boldsymbol{u}_k + e_k^{Fluct}\right) \\
&\quad = \nabla \cdot [\alpha_k(\boldsymbol{T}_k + \boldsymbol{T}_k^{Fluct}) \cdot \boldsymbol{u}_k] \\
&\quad - \nabla \cdot \alpha_k(\boldsymbol{q}_k + \boldsymbol{q}_k^{Fluct}) + \alpha_k\rho_k(\varepsilon_k + \boldsymbol{g}_k \cdot \boldsymbol{u}_k) \\
&\quad + \Omega_k^{energy} + \boldsymbol{\Omega}_k^{mom} \cdot \boldsymbol{u}_k^{int} + \Omega_k^{mass}\left(e_k^{int} + \frac{1}{2}\boldsymbol{u}_k^{int} \cdot \boldsymbol{u}_k^{int}\right)
\end{aligned} \tag{147}$$

where

$$e_k^{Fluct} \equiv \frac{1}{2}\frac{\langle X_k\rho_0\boldsymbol{u}_k' \cdot \boldsymbol{u}_k'\rangle}{\alpha_k\rho_k} \tag{148}$$

is the fluctuation kinetic energy,

$$\boldsymbol{q}_k^{Fluct} \equiv \frac{\langle X_k\rho_0\boldsymbol{u}_k'e_k'\rangle}{\alpha_k} + \frac{\langle X_k\boldsymbol{T}_0 \cdot \boldsymbol{u}_k'\rangle}{\alpha_k} + \frac{1}{2}\frac{\langle X_k\rho_0\boldsymbol{u}_k'(\boldsymbol{u}_k' \cdot \boldsymbol{u}_k')\rangle}{\alpha_k} \tag{149}$$

is the fluctuation energy flux,

$$\varepsilon_k \equiv \frac{\langle X_k\rho_0\varepsilon_0\rangle}{\alpha_k\rho_k} \tag{150}$$

is the energy source,

$$\Omega_k^{energy} \equiv \langle \boldsymbol{q}_0 \cdot \nabla X_k\rangle \tag{151}$$

is the interfacial heat source, and

$$\boldsymbol{\Omega}_k^{mom} \cdot \boldsymbol{u}_k^{int} \equiv -\langle \boldsymbol{T}_0 \cdot \boldsymbol{u}_0 \cdot \nabla X_k\rangle \tag{152}$$

is the interfacial work term. The averaged interfacial energy balance constraint (interface jump condition) is

$$\sum_{k=1}^{no.\,of\,phases} \Omega_k^{energy} + \boldsymbol{\Omega}_k^{mom} \cdot \boldsymbol{u}_k^{int} + \Omega_k^{mass} \left(e_k^{int} + \frac{1}{2} \boldsymbol{u}_k^{int} \cdot \boldsymbol{u}_k^{int} \right) = \xi \tag{153}$$

where ξ is the interfacial energy source. The kinetic energy associated with the velocity fluctuations, e_k^{Fluct}, is a type of "turbulent" kinetic energy. Sometimes the sum $e_k + e_k^{Fluct}$ is interpreted as the effective internal energy per unit mass of phase k.

It is sometimes useful to have an expression for the balance of fluctuation kinetic energy, e_k^{Fluct}. Its evolutionary description is derived by introducing the partition $\boldsymbol{u}_k' = \boldsymbol{u}_0 - \boldsymbol{u}_k$ into the microscopic pure phase momentum balance, taking the dot product of this equation with $X_k \boldsymbol{u}_k'$, and then performing the statistical average over configurations (keeping in mind that $\langle X_k \rho_0 \boldsymbol{u}' \rangle$ vanishes) to obtain (details are left to the reader, see e.g. [42])

$$\begin{aligned} \alpha_k \rho_k \frac{\partial e_k^{Fluct}}{\partial t} + \alpha_k \rho_k \boldsymbol{u}_k \cdot \nabla e_k^{Fluct} &= \alpha_k \boldsymbol{T}_k^{Fluct} : \nabla \boldsymbol{u}_k \\ &\quad - \nabla \cdot \langle X_k \rho_0 \frac{\boldsymbol{u}_k' \cdot \boldsymbol{u}_k'}{2} \boldsymbol{u}_k' \rangle \\ &\quad + \langle X_k \boldsymbol{u}_k' \cdot (\nabla \cdot \boldsymbol{T}_0 + \rho_0 \boldsymbol{g}_0) \rangle \ . \end{aligned} \tag{154}$$

This equation exhibits some similarity to the equation of evolution of the fluctuational kinetic energy in a single-phase turbulent fluid [43]. The first term on the right side describes the influence of the gradient of $\boldsymbol{u}_k$ on the development of e_k^{Fluct}, the second term is expected to diffuse e_k^{Fluct}, and the last term represents the power developed by the stresses and external forces [36].

For most multiphase flows, including some very (conceptually) simple flows such as gas flow through a packed bed or through a pebble-bed nuclear reactor, the nature of e_k^{Fluct} is somewhat different than that of a turbulent single-phase flow. Contrary to a single-phase fluid in which the fluctuations disappear for slow flows, these fluctuations for a multiphase flow exist however slow the flow. For this reason, e_k^{Fluct} that is produced by hydrodynamic interactions between the phases has been called "pseudo-turbulence," for example by Lhuillier [36].

3.1.8 Entropy Inequality

The local form of the entropy inequality (112), sometimes called the "Second Law of Thermodynamics," is used to place restrictions on the constitutive relations used to give unique phase or material behaviors. With the assignment of

$$\psi_0 = s_0, \qquad J_0 = -\frac{\boldsymbol{q}_0}{\theta_0}, \qquad g_0 = \frac{\varepsilon_0}{\theta_0} \tag{155}$$

to the variables of the generic averaged balance relationship, the averaged entropy inequality results,

$$\begin{aligned}\frac{\partial \alpha_k \rho_k s_k}{\partial t} + \nabla \cdot \alpha_k \rho_k s_k \boldsymbol{u}_k \geq \nabla \cdot \alpha_k (\boldsymbol{\Phi}_k + \boldsymbol{\Phi}_k^{Fluct}) \\ + \alpha_k \rho_k S_k + \Omega_k^{entropy} + \Omega_k^{mass} s_k^{int}\end{aligned} \tag{156}$$

where

$$\boldsymbol{\Phi}_k \equiv -\frac{\left\langle X_k \frac{\boldsymbol{q}_0}{\theta_0} \right\rangle}{\alpha_k} \tag{157}$$

is the entropy flux,

$$\boldsymbol{\Phi}_k^{Fluct} \equiv -\frac{\langle X_k \rho_0 s_k' \boldsymbol{u}_k' \rangle}{\alpha_k} \tag{158}$$

is the fluctuation entropy flux,

$$S_k \equiv \frac{\left\langle X_k \frac{\rho_0 \varepsilon_0}{\theta_0} \right\rangle}{\alpha_k \rho_k} \tag{159}$$

is a volumetric entropy source, and

$$\Omega_k^{entropy} \equiv \left\langle \frac{\boldsymbol{q}_0}{\theta_0} \cdot \nabla X_k \right\rangle \tag{160}$$

is an interfacial entropy source. This entropy inequality corresponds to what Drew and Passman [34] call the *microscopic entropy inequality*. A *macroscopic entropy inequality* can be obtained by summing inequalities (156) over all of the phases or materials present in the mixture (for details, see Truesdell [44] and the other authors contained therein). The macroscopic entropy inequality is useful for placing restrictions on the phasic or material interaction constitutive relations. The averaged interfacial entropy inequality (interfacial jump condition) is

$$\sum_{k=1}^{no.\,of\,phases} \Omega_k^{entropy} + \Omega_k^{mass} s_k^{int} \geq 0 \ . \tag{161}$$

3.1.9 Volume Fraction Propagation Equation

There remains one very important relationship to derive, a dynamic relationship that effectively reflects boundary conditions at the microscale. It accounts for the fact that the constituent volume fractions may change without affecting the gross motion and, in a sense, models the microstructural force systems operating within the multiphase mixture. Beginning with the previous Lagrangian interface material derivative relationship for X_k,

$$\frac{\partial X_k}{\partial t} + \boldsymbol{u}_{int} \cdot \nabla X_k = 0 \tag{162}$$

this equation is averaged to give

$$\left\langle \frac{\partial X_k}{\partial t} + \boldsymbol{u}_{int} \cdot \nabla X_k \right\rangle = \frac{\partial \alpha_k}{\partial t} + \langle \boldsymbol{u}_{int} \cdot \nabla X_k \rangle = 0 \ . \tag{163}$$

Introducing the fluctuating interface velocity $\boldsymbol{u}'_I = \boldsymbol{u}_{int} - \boldsymbol{u}_I$, where $\boldsymbol{u}_I$ is the *average interface velocity*, into this equation yields

$$\begin{aligned} \frac{\partial \alpha_k}{\partial t} + \langle \boldsymbol{u}_{int} \cdot \nabla X_k \rangle &= \frac{\partial \alpha_k}{\partial t} + \langle (\boldsymbol{u}_I + \boldsymbol{u}'_I) \cdot \nabla X_k \rangle \\ &= \frac{\partial \alpha_k}{\partial t} + \boldsymbol{u}_I \cdot \nabla \alpha_k + \langle \boldsymbol{u}'_I \cdot \nabla X_k \rangle \\ &= \frac{\partial \alpha_k}{\partial t} + \boldsymbol{u}_I \cdot \nabla \alpha_k - \Omega_k^{vol} \\ &= 0 \end{aligned} \tag{164}$$

where Ω_k^{vol} (for which a constitutive description will be needed) is the driving function for the change of volume fraction α_k with time. In summary, the volume fraction propagation, or *volume fraction evolution equation* is written as

$$\frac{\partial \alpha_k}{\partial t} + \boldsymbol{u}_I \cdot \nabla \alpha_k = \Omega_k^{vol} \ . \tag{165}$$

The volume fraction evolution equation plays a central role in modern, well-posed two-phase models with correct wave dynamics.

Even before 2000, past researchers had proposed and utilized various forms of the independent volume fraction evolution equation: second order (in time) with "microinertia" effects [30, 45], first order (in time) as above with "viscous damping" effects [46–51], and zeroth order (in time) which amounts to a steady-state version balancing the microstructural forces operating within the Ω_k^{vol} function [34, 52–57]. Most of these were also used

in conjunction with so called "two pressure" two-phase flow models, which will be examined next. Since 2000, the literature has become much more voluminous, documenting the variations of models utilizing an independent volume fraction evolution equation, usually with independent phasic pressures. Most are from Europe. It is not the intent here to provide a review of such.

To gain closure for this set of generic material (fluid) balance equations, additional relations must be specified which will restore information that was lost during the averaging process, and render the model material specific. All of these relations are collectively referred to as constitutive relations. Those that are pertinent to the RELAP-7 equation system will be discussed in the following section on constitutive equations, but it is easier to discuss the microstructural force model, which is an important part of Ω_k^{vol}, in the volume fraction evolution equation now, before reducing the multi-dimensional model above (which will be applied in other INL MOOSE-based applications) to the 1-D variable cross-sectional area equation system employed in RELAP-7.

The need for, and form of, a dynamic volume fraction evolution equation is presented next with deliberate choice of an "intuitive" engineering approach over, perhaps, a "rigorously theoretical" approach. Consider a cell mixture physics model for two-phase flow in which a fixed volume V is instantaneously filled with two immiscible constituents or phases (e.g. from a computational fluid dynamical modeling point of view, these two constituents may have been advected into a mixed cell control volume). These two constituents have masses m_1 and m_2 occupying volumes V_1 and V_2, respectively, such that

$$V_1 + V_2 = V \ . \tag{166}$$

The constituent phases have material density ρ_1 and ρ_2, respectively, so

$$\begin{aligned} V &= V_1 + V_2 \\ &= \frac{m_1}{\rho_1} + \frac{m_2}{\rho_2} \end{aligned} \tag{167}$$

or

$$\begin{aligned} 1 &= \frac{V_1}{V} + \frac{V_2}{V} \\ &= \alpha_1 + \alpha_2 \\ &= \frac{m_1}{V\rho_1} + \frac{m_2}{V\rho_2} \end{aligned} \tag{168}$$

where $\alpha_1 = \frac{V_1}{V}$ and $\alpha_2 = \frac{V_2}{V}$ are now volume fractions (or with ensemble averaging, expected phasic presence) of each constituent or phase. For each phase, because $\rho_1 = \frac{m_1}{V_1}$

and $\rho_2 = \frac{m_2}{V_2}$, using a generic equation of state gives

$$\begin{aligned} p_1 &= f_1(\rho_1, e_1) \\ &= f_1\left(\frac{m_1}{V_1}, e_1\right) \end{aligned} \tag{169}$$

$$\begin{aligned} p_2 &= f_2(\rho_2, e_2) \\ &= f_2\left(\frac{m_2}{V_2}, e_2\right) . \end{aligned} \tag{170}$$

Generally the pressures p_1 and p_2 of the two phases are not equal. In fact, if V_1 and V_2 are adjusted (subject to the $V_1^* + V_2^* = V$ constraint) until the two phase pressures are equal to the "equilibration" or "equilibrium pressure" or "relaxed pressure", p, then

$$p = f_1\left(\frac{m_1}{V_1^*},\, e_1\right) = f_2\left(\frac{m_2}{V_2^*},\, e_2\right) . \tag{171}$$

At this equilibrium pressure the corresponding phase volumes yield the equilibrium volume fractions

$$\alpha_1^e = \frac{V_1^*}{V}, \qquad \alpha_2^e = \frac{V_2^*}{V} . \tag{172}$$

Alternatively, equations (169) and (170) can be rewritten as

$$\begin{aligned} p_1 &= f_1(\rho_1,\, e_1) \\ &= f_1\left(\frac{m_1}{\alpha_1 V},\, e_1\right) \end{aligned} \tag{173}$$

$$\begin{aligned} p_2 &= f_2(\rho_2,\, e_2) \\ &= f_2\left(\frac{m_2}{\alpha_2 V},\, e_2\right) = f_2\left(\frac{m_2}{(1-\alpha_1)V},\, e_2\right) \end{aligned} \tag{174}$$

and equivalently, α_1 can be varied until the equilibrium pressure is obtained along with the corresponding equilibrium volume fraction(s). Note also that, for two phases $\alpha_1 + \alpha_2 = 1$ and consequently $\frac{d\alpha_1}{dt} = -\frac{d\alpha_2}{dt}$ and $\frac{d^2\alpha_1}{dt^2} = -\frac{d^2\alpha_2}{dt^2}$. Intuitively, this can be accomplished in a dynamical manner with

$$\frac{d\alpha_1}{dt} = \frac{p_1 - p_2}{\tau} . \tag{175}$$

If α_1 is compressed too much (such that $p_1 > p_2$) then α_1 will increase with time (i.e. relax) letting p_1 reduce while α_2 decreases, thereby letting p_2 increase. This process will continue until $p_1 = p_2 = p$ and thus $\frac{d\alpha_1}{dt} = 0$. The relaxation rate, τ, controls the rate at which the phases (pressures) equilibrate or relax.

With analogy to the classical dynamics of simple mass-dashpot systems, a more general dynamical description of volume fractions could even be considered, wherein

$$\frac{d}{dt}\left[(\text{microinertia}) \times \frac{d\alpha_1}{dt}\right] + (\text{compression viscosity}) \times \frac{d\alpha_1}{dt} = (\text{microstructural forces}) = F \ . \tag{176}$$

The microstructural force F is a relaxation term that is intended to model the driving force or resistance exhibited by the mixture to changes in its configuration (volume fractions). Playing further upon this simple abstraction (analogy), the "microinertia" function is analogous to "mass" and the "compression viscosity" function is analogous to the viscous damping coefficient. As a simple example from mechanics, consider the compaction of a gas-solid particle bed [58] with

$$F = \begin{cases} \alpha_s \alpha_g (p_s - p_g - \beta_s) & , \quad p_s - \beta_s > 0 \\ -\alpha_s \alpha_g p_g & , \quad p_s - \beta_s \leq 0 \end{cases} \tag{177}$$

in accordance with the view of compaction as an irreversible process. β_s is the "configuration pressure" of the bed. If the microinertia and the configuration pressure are set to zero, then

$$\frac{d\alpha_1}{dt} = \frac{\alpha_1 \alpha_2 (p_1 - p_2)}{\mu} \tag{178}$$

where for this example μ could be referred to as the "compaction viscosity". Note the multiplicative coefficient $\alpha_1 \alpha_2$ in the driving force F. This term is included for two reasons. First, $\alpha_1 \alpha_2$ is roughly proportional to the interfacial area per unit volume, $\frac{A_i}{V}$. Second, better behavior results in the single-phase limit, i.e. $\alpha_1 \to 0$, $(\alpha_2 \to 1)$ or $\alpha_2 \to 0$, $(\alpha_1 \to 1)$. This concept will be further refined for the two-phase flow model of RELAP-7.

3.1.10 Multi-dimensional Two-Phase Governing Equations

Before moving on to the 1-D variable cross-sectional area form of the 7-equation two-phase model (next section), it is useful to collect a simplified multi-dimensional version of the mass, momentum, and energy balance equations, equations (121), (140), and (147) respectively, as well as the volume fraction evolution equation (165) with simple pressure driving force. For the liquid ("liq" subscript) and vapor ("vap" subscript) phases, we have

$$\frac{\partial\left(\alpha\rho\right)_{liq}}{\partial t}+\nabla\cdot\left(\alpha\rho\boldsymbol{u}\right)_{liq}=\Omega_{liq}^{mass} \tag{179}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho\boldsymbol{u}\right)_{liq}}{\partial t}+\nabla\cdot\left(\alpha\rho\boldsymbol{u}\otimes\boldsymbol{u}+\alpha p\boldsymbol{I}\right)_{liq}&=p_{int}\nabla\alpha_{liq}+\lambda(\boldsymbol{u}_{vap}-\boldsymbol{u}_{liq})\\&\quad+\left(\alpha\rho\right)_{liq}\boldsymbol{g}+\boldsymbol{u}_{int}\Omega_{liq}^{mass}\end{aligned} \tag{180}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{liq}}{\partial t}+\nabla\cdot\left[\alpha(\rho E+p)\boldsymbol{u}\right]_{liq}&=p_{int}\boldsymbol{u}_{int}\cdot\nabla\alpha_{liq}+\lambda\boldsymbol{u}_{int}'(\boldsymbol{u}_{vap}-\boldsymbol{u}_{liq})\\&\quad-\mu p_{int}'(p_{liq}-p_{vap})+E_{int}\Omega_{liq}^{mass}+Q_{liq}\end{aligned} \tag{181}$$

$$\frac{\partial\alpha_{liq}}{\partial t}+\nabla\alpha_{liq}\cdot\boldsymbol{u}_{int}=\mu(p_{liq}-p_{vap})+\frac{\Omega_{liq}^{mass}}{\rho_{int}} \tag{182}$$

$$\frac{\partial\left(\alpha\rho\right)_{vap}}{\partial t}+\nabla\cdot\left(\alpha\rho\boldsymbol{u}\right)_{vap}=-\Omega_{liq}^{mass} \tag{183}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho\boldsymbol{u}\right)_{vap}}{\partial t}+\nabla\cdot\left(\alpha\rho\boldsymbol{u}\otimes\boldsymbol{u}+\alpha p\boldsymbol{I}\right)_{vap}&=p_{int}\nabla\alpha_{vap}-\lambda(\boldsymbol{u}_{vap}-\boldsymbol{u}_{liq})\\&\quad+\left(\alpha\rho\right)_{vap}\boldsymbol{g}-\boldsymbol{u}_{int}\Omega_{liq}^{mass}\end{aligned} \tag{184}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{vap}}{\partial t}+\nabla\cdot\left[\alpha(\rho E+p)\boldsymbol{u}\right]_{vap}&=p_{int}\boldsymbol{u}_{int}\cdot\nabla\alpha_{vap}-\lambda\boldsymbol{u}_{int}'(\boldsymbol{u}_{vap}-\boldsymbol{u}_{liq})\\&\quad+\mu p_{int}'(p_{liq}-p_{vap})-E_{int}\Omega_{liq}^{mass}-Q_{liq}\end{aligned} \tag{185}$$

$$\frac{\partial\alpha_{vap}}{\partial t}+\nabla\alpha_{vap}\cdot\boldsymbol{u}_{int}=-\mu(p_{liq}-p_{vap})-\frac{\Omega_{liq}^{mass}}{\rho_{int}} \tag{186}$$

where $\boldsymbol{u}_{int}$ is the interface velocity inside the two-phase control volume and $\boldsymbol{u}_{int}'$ is the average interfacial velocity. The pressure exerted on the interfacial surface inside the two-phase control volume, interface pressure, is denoted p_{int} and the average interfacial pressure by p_{int}'. In these equations Q_{liq} denotes the direct energy transfer from the vapor

phase to the liquid phase not due to interphase mass transfer, and $E_k = e_k + \frac{1}{2}\boldsymbol{u}_k \cdot \boldsymbol{u}_k + \boldsymbol{g}h_{k,datum}$ $(k = liq,\, vap)$ represents the phasic total energy. Note that in a two-phase system, the saturation constraint allows either (182) or (186) to be replaced by the algebraic relation

$$\alpha_{vap} = 1 - \alpha_{liq} \ . \tag{187}$$

In this relaxation model, μ has been redefined as the reciprocal of that used above to intuitively describe the volume fraction evolution equation (where it was referred to, in a narrow context, as a "compaction viscosity"; before that it was referred to as compression viscosity and as a relaxation rate time constant τ). Now in this new form, μ will be more generally called the *pressure relaxation coefficient* or function and similarly λ is the *velocity relaxation coefficient* or function. Relaxation models play a key role in the modern theory of hyperbolic partial differential equations – physically, analytically, and numerically (see Leveque [59] for an introduction).

3.1.11 One-dimensional, Variable Cross-sectional Area, Seven Equation Two-phase Model

Because it is not economical to solve the entire two-phase flow field with highly resolved three-dimensional computational fluid dynamics for an entire light water reactor coolant system, it is necessary to construct a one-dimensional model for flow in pipes, nozzles, and other components. The one-dimensional model is constructed from the multi-dimensional model, following the approach developed in the one-dimensional Single-Phase Flow Model Section 2.1, to allow the representation of continuously variable cross-sectional area.

Consider flow through a duct with local cross-sectional area $A = A(x,t)$. Actually, most of the time we consider local cross-sectional area to depend upon position coordinate x only, for which a time rate of change of cross-sectional area is not necessary because for this case $\frac{\partial A}{\partial t} = 0$. However, $A(x,t)$ is left inside the time derivative terms for generality and possible future use. Applying the methods developed in the Single-Phase Flow Model Section 2.1 to the 7-equation model in Section 3.1.10 results in:

$$\frac{\partial \left(\alpha\rho\right)_{liq} A}{\partial t} + \frac{\partial \left(\alpha\rho u\right)_{liq} A}{\partial x} = -\Gamma^{\text{int}}_{\ell\to v} A_{int} A - \Gamma^{\text{wall}}_{\ell\to v} P_{\text{hf}} \tag{188}$$

$$\frac{\partial \left(\alpha\rho u\right)_{liq} A}{\partial t} + \frac{\partial \alpha_{liq} A \left(\rho u^2 + p\right)_{liq}}{\partial x} = p_{int} A \frac{\partial \alpha_{liq}}{\partial x} + p_{liq} \alpha_{liq} \frac{\partial A}{\partial x}$$
$$+ A\lambda(u_{vap} - u_{liq})$$
$$- \Gamma^{\text{int}}_{\ell \to v} A_{int} u_{int} A$$
$$- \Gamma^{\text{wall}}_{\ell \to v} u_{\text{int}} P_{\text{hf}}$$
$$- F^{\text{friction}}_{\text{wall},\ell} - F_{\text{friction},vap} - F^{\text{form}}_{\ell}$$
$$+ \left(\alpha\rho\right)_{liq} A \boldsymbol{g} \cdot \hat{\boldsymbol{n}}_{axis} \tag{189}$$

$$\frac{\partial \left(\alpha\rho E\right)_{liq} A}{\partial t} + \frac{\partial \alpha_{liq} u_{liq} A \left(\rho E + p\right)_{liq}}{\partial x} = p_{int} u_{int} A \frac{\partial \alpha_{liq}}{\partial x} - \bar{p}_{int} A \mu (p_{liq} - p_{vap})$$
$$+ \bar{u}_{int} A \lambda (u_{vap} - u_{liq})$$
$$+ \Gamma^{\text{int}}_{\ell \to v} A_{int} \left(\frac{p_{int}}{\rho_{int}} - H_{liq,int} \right) A$$
$$- \Gamma^{\text{wall}}_{\ell \to v} H_{\ell} P_{\text{hf}}$$
$$+ Q_{int,liq} + Q^{\text{wall,conv}}_{\ell} \tag{190}$$
$$+ \left(\alpha\rho u\right)_{liq} A \boldsymbol{g} \cdot \hat{\boldsymbol{n}}_{axis} \tag{191}$$

$$\frac{\partial \alpha_{liq} A}{\partial t} + u_{int} A \frac{\partial \alpha_{liq}}{\partial x} = A\mu(p_{liq} - p_{vap}) \tag{192}$$
$$- \frac{\Gamma^{\text{int}}_{\ell \to v} A_{int} A}{\rho_{int}} - \frac{\Gamma^{\text{wall}}_{\ell \to v} P_{\text{hf}}}{\rho_{\text{int}}}$$

for the liquid phase, and

$$\frac{\partial\left(\alpha\rho\right)_{vap}A}{\partial t}+\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial x}=\Gamma^{\text{int}}_{\ell\rightarrow v}A_{int}A+\Gamma^{\text{wall}}_{\ell\rightarrow v}P_{\text{hf}} \tag{193}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}A\left(\rho u^2+p\right)_{vap}}{\partial x}&=p_{int}A\frac{\partial\alpha_{vap}}{\partial x}+p_{vap}\alpha_{vap}\frac{\partial A}{\partial x}\\&+A\lambda(u_{liq}-u_{vap})\\&+\Gamma^{\text{int}}_{\ell\rightarrow v}A_{int}u_{int}A\\&+\Gamma^{\text{wall}}_{\ell\rightarrow v}u_{\text{int}}P_{\text{hf}}\\&-F^{\text{friction}}_{\text{wall},v}-F_{\text{friction},liq}-F^{\text{form}}_{v}\\&+\left(\alpha\rho\right)_{vap}A\boldsymbol{g}\cdot\hat{\boldsymbol{n}}_{axis}\end{aligned} \tag{194}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}u_{vap}A\left(\rho E+p\right)_{vap}}{\partial x}&=p_{int}u_{int}A\frac{\partial\alpha_{vap}}{\partial x}-\bar{p}_{int}A\mu(p_{vap}-p_{liq})\\&+\bar{u}_{int}A\lambda(u_{liq}-u_{vap})\\&-\Gamma^{\text{int}}_{\ell\rightarrow v}A_{int}\left(\frac{p_{int}}{\rho_{int}}-H_{vap,int}\right)A\\&+\Gamma^{\text{wall}}_{\ell\rightarrow v}H_{\ell}P_{\text{hf}}\\&+Q_{int,vap}+Q^{\text{wall}}_{v}+Q^{\text{wall,boil}}_{\ell} \qquad (195)\\&+\left(\alpha\rho u\right)_{vap}A\boldsymbol{g}\cdot\hat{\boldsymbol{n}}_{axis} \qquad (196)\end{aligned}$$

$$\begin{aligned}\frac{\partial\alpha_{vap}A}{\partial t}+u_{int}A\frac{\partial\alpha_{vap}}{\partial x}&=A\mu(p_{vap}-p_{liq})\\&+\frac{\Gamma^{\text{int}}_{\ell\rightarrow v}A_{int}A}{\rho_{int}}+\frac{\Gamma^{\text{wall}}_{\ell\rightarrow v}P_{\text{hf}}}{\rho_{\text{int}}}\end{aligned} \tag{197}$$

for the vapor phase. As before, it is noted that for two-phase flow, either of the differential relations (192) or (197) may be replaced with the algebraic relation

$$\alpha_{vap}=1-\alpha_{liq} \tag{198}$$

throughout, reducing the total number of equations to be solved to seven.

In equations (188)–(197), $\Gamma^{\text{int}}_{\ell\rightarrow v}$ is the net mass transfer per unit interfacial area from the liquid to the vapor phase and A_{int} is the interfacial area per unit volume of mixture.

Also, $H_{liq,int}$ and $H_{vap,int}$ are the liquid and gas total enthalpies at the interface, respectively. The nomenclature has also been modified so that now u_{int} and $\bar{u}_{int}$ are, respectively, the interfacial velocity and average interfacial velocity; and p_{int} and $\bar{p}_{int}$ are, respectively, the interfacial pressure and average interfacial pressure. In the momentum balance equations $\hat{\boldsymbol{n}}_{axis}$ is the unit vector directly along the axis of the duct, which is also the $\pm$ flow direction. Of course $F^{\text{friction}}_{\text{wall},k}$ is the frictional force due to the wall acting on phase k, and $F_{\text{friction},k'}$ is the frictional force acting on phase k due to the presence of the other phase k'. F^{form}_{k} is the linear form-loss force acting on phase k. Similarly, $Q_{int,k}$ is the direct heat transfer from the interface to phase k, Q^{wall}_{v} is the direct heat transfer rate per unit length from the wall to the vapor phase, and $Q^{\text{wall,conv}}_{\ell}$ and $Q^{\text{wall,boil}}_{\ell}$ are the convective and boiling components of wall heat transfer rate per unit length to liquid, respectively. Note that the boiling component $Q^{\text{wall,boil}}_{\ell}$ is used in vaporization and thus ultimately ends up in the vapor phase. The wall mass flux from liquid to vapor is denoted by $\Gamma^{\text{wall}}_{\ell\rightarrow v}$, and the heated perimeter is denoted by P_{hf}.

Equation system (188)–(197) is the basic system solved with RELAP-7. The system was implemented within the MOOSE computational framework following a series of logically-complete steps [60] designed to confidently allow physically- and mathematically-meaningful benchmark testing at each step of increased complexity. This 7-equation two-phase model allows both phases to be compressible. Because p_{vap} is not, in many practical problems, very different from p_{liq} (with the exception of surface tension effects), most traditional two-phase models assume $p_{vap} = p_{liq}$ which allows the elimination of one dependent variable and serves as a substitute for the volume fraction evolution equation. However, $p_{vap} \approxeq p_{liq}$ does not entail the same property for their partial derivatives [53]. Therefore the assumption of $p_{vap} = p_{liq}$ is very restrictive when derivatives are involved. As pointed out by Boure and Delhaye [61], it requires that pressure disturbances have the same average effect on the two phases and, in particular, that they propagate at the same velocity within the phases. While the assumption $p_{vap} = p_{liq}$ has proved useful in many cases, it is definitely too restrictive when propagation phenomena are important[2]. The RELAP-7 approach forgoes this assumption and retains the 7-equation model as its basis.

More importantly, the 7-equation model allows for complete mechanical and thermodynamic non-equilibrium between the phases and it is hyperbolic and well-posed in the

[2]With the *complex characteristics* that can occur with the classical 6-equation model, it is not clear how to set the boundary conditions, and high wave number instabilities occur during convergence testing. It has been argued that equation sets with complex characteristics may still model a range of phenomena quite adequately if the numerical method introduces sufficient dissipation to damp the high frequency instabilities. There are obviously real physical effects that do this but are left out of the equations. As pointed out in [62] one does not always know whether these effects are important and under what conditions they are important.

sense of Hadamard[3]. The hyperbolicity (and thus well-posedness) of this model is a direct result of incorporation of both phases' compressibilities, and not of a manipulation of interfacial variables as is done in CATHARE [65]. The system has symmetrically occurring eigenvalues and eigenvectors with respect to the two-phases; its wave speeds (eigenvalues) are $(u \pm c)_{liq}$ and $(u \pm c)_{vap}$ for the genuinely nonlinear fields, and u_{liq}, u_{vap}, and u_{int} (multiplicity 2) for the linear degenerate fields. This 7-equation two-phase model is a relaxation model and it has the very desirable feature of naturally devolving to simpler, even classical, models upon mechanical and/or thermodynamical relaxation [66]. Thus, this model can readily couple to simpler models via a natural transition from the 7-equation model to a classical 6-equation (ill-posed) model, a 5-equation Kapila model [67, 68], a 4-equation homogeneous relaxation model (HRM), or a 3-equation homogeneous equilibrium model (HEM). It is noted that, because of this feature, experience shows that some physically and mathematically realistic solutions may, upon first examination, appear counter-intuitive to the inexperienced modeler. More will be said about this later.

3.2 Constitutive Models

Without additional closure equations the balance relations derived above are generic, i.e. they apply to all materials (fluids). They must made to apply to the unique material (fluid) being considered – material specific. Also, though averaging the microlevel balance equations led to a "simplified" or perhaps more tractable model, this simplification (averaging) led to a loss of information, and some additional relations must also be specified to supply (or restore) at least some information that was lost in this process[4]. Collectively, any additional relations, or sub-models, that must be specified to render mathematical closure (allowing a solution to be obtainable) to the generic balance equations are known as "constitutive relations". Familiar examples of constitutive relations from single-phase flow include ideal gas equation of state, Newtonian fluid stress-rate of strain laws, Fourier's law for heat conduction, k-ϵ turbulence model.

Because the 7-equation two-phase model's most unique features are reflected in the

[3]The mathematician Jacques Hadamard [63] espoused that a "well-posed" mathematical model of physical phenomena should have the properties that (1) a solution exists, (2) the solution is unique, and (3) the solution's behavior depends continuously upon the initial conditions. Problems that are not well-posed are said to be "ill-posed." Early researchers in two-phase flow knew that, if due diligence was not exercised, an ill-posed formulation could result; and they understood the need for a well-posed model, as summarized in Hughes et al. [64].

[4]The process of averaging the balance equations produced a system with more unknowns than equations; thus postulates or empirical correlations are required to resolve this deficiency.

presence of a volume fraction evolution equation, interfacial pressure and velocity, and mechanical relaxation terms involving pressure and velocity relaxation, it is natural to begin with their constitutive relations. Constitutive ideas associated with the volume fraction evolution equation were discussed previously for pedagogical reasons. Thermodynamical relaxation will be discussed subsequently, followed by other closures.

3.2.1 Interface Pressure and Velocity, Mechanical Relaxation Coefficients

In the original 7-equation model of Baer and Nunziato [50], p_{int} was chosen to be equal to the phasic pressure of the phase with the largest acoustic impedance which for two-phase liquid-vapor flow corresponds to that of the liquid, i.e. $p_{int} = p_{liq}$. On the other hand, they took the interface velocity u_{int} to be that of the phase with the smallest acoustic impedance, which for liquid-vapor flows corresponds to that of the vapor phase, or $u_{int} = u_{vap}$. Later, Saurel and others chose the following interfacial values

$$p_{int} = \sum_{k=1,2} \alpha_k p_k \tag{199}$$

$$u_{int} = \frac{\sum_{k=1,2} \alpha_k \rho_k u_k}{\sum_{k=1,2} \alpha_k \rho_k} \ . \tag{200}$$

In this early research, mechanical relaxation parameters μ and λ were also specified in a, more or less, ad hoc manner. Abgrall and Saurel [69] introduced a clever generalization to the development of the 7-equation model, the discrete equation method (DEM), which permits some interesting closure capability. In reviewing the traditional approach presented above, the microscopic level, single-phase balance equations (PDEs) are first averaged to obtain macroscopic averaged balance equations (again PDEs). Then appropriate simplifying assumptions, including constitutive relations, are applied to this macroscopic system giving a simplified averaged balance equation system. Finally, the simplified averaged PDE system is discretized numerically using finite difference, finite volume, or finite element methods and the numerical solution is obtained.

With the DEM approach, a generic phase distribution topology is first assumed, then a discretized solution is developed within the computational cell employing Riemann or approximate Riemann methods. Then finally, this discrete local solution is effectively averaged over the cell volume and time to obtain a meaningful macroscopic solution. The DEM method carries a pressure and velocity for each phase and, because it effectively only solves Euler equations locally, is hyperbolic and well-posed and gives correct wave dynamics. But this new homogenization method offers an additional bonus; the DEM can

be used not only to obtain the 7-equation model above, but also explicit closure formulas for p_{int}, u_{int}, μ, and λ that are symmetric, compatible with the second law of thermodynamics, and responsible for the fulfillment of interface conditions when dealing with contact/interface problems! In the continuous limit of small mesh spacing and time steps along with employment of the Godunov weak wave limit, the finite closure relations converge [15, 70] to

$$p_{int} = \bar{p}_{int} + \frac{Z_{liq}Z_{vap}}{Z_{liq}+Z_{vap}}\text{sgn}\left(\frac{\partial \alpha_{liq}}{\partial x}\right)(u_{vap}-u_{liq}) \tag{201}$$

$$\bar{p}_{int} = \frac{Z_{vap}p_{liq}+Z_{liq}p_{vap}}{Z_{liq}+Z_{vap}} \tag{202}$$

$$u_{int} = \bar{u}_{int} + \text{sgn}\left(\frac{\partial \alpha_{liq}}{\partial x}\right)\frac{p_{vap}-p_{liq}}{Z_{liq}+Z_{vap}} \tag{203}$$

$$\bar{u}_{int} = \frac{Z_{liq}u_{liq}+Z_{vap}u_{vap}}{Z_{liq}+Z_{vap}} \tag{204}$$

$$\lambda = \frac{1}{2}\mu Z_{liq}Z_{vap} \tag{205}$$

$$\mu = \frac{A_{int}}{Z_{liq}+Z_{vap}} \tag{206}$$

where λ is the velocity relaxation coefficient function, μ is the pressure relaxation coefficient function, $Z_k = \rho_k w_k$, $(k = liq, vap)$, is the phasic acoustic impedance and A_{int} is the specific interfacial area (i.e. the interfacial surface area per unit volume of two-phase mixture) which must be specified from some type of flow regime map or function. The DEM model for two-phase flow of water and its vapor in a one dimensional duct of spatially varying cross-section was derived and demonstrated with these closures by Berry et al. [13].

Remark (1): From this specification of λ and μ it is clear that special coupling is rendered. To relax the 7-equation model to the ill-posed classical 6-equation model, the pressures should be relaxed toward a single pressure for both phases. This is accomplished by specifying the pressure relaxation coefficient to be very large, i.e. letting it approach infinity. But if the pressure relaxation coefficient goes to infinity, so does the velocity relaxation rate also approach infinity. This then relaxes the 7-equation model not to the classical 6-equation model, but to the mechanical equilibrium 5-equation model of Kapila. This reduced 5-equation model is also hyperbolic and well-posed. The 5-equation model provides

a very useful starting point for constructing multi-dimensional interface resolving methods which dynamically captures evolving, and even spontaneously generating, interfaces [71]. Thus the 7-equation model of RELAP-7 can be relaxed locally to couple seamlessly with such a multi-dimensional, interface resolving code.

Remark (2): Numerically, the mechanical relaxation coefficients μ (pressure) and λ (velocity) can be relaxed independently to yield solutions to useful, reduced models (as explained previously). It is noted, however, that relaxation of pressure only by making μ large without relaxing velocity will indeed give ill-posed and unstable numerical solutions, just as the classical 6-equation two-phase model does, with sufficiently fine spatial resolution, as confirmed in [13, 72].

Remark (3): Even though the implementation of the 7-equation two-phase model within RELAP-7 (or any other code for that matter) does not use the generalized approach of DEM, the interfacial pressure and velocity closures as well as the pressure and velocity relaxation coefficients of Equations (201) to (206) are utilized.

3.2.2 Wall and Interface Direct Heat Transfer

Without wall boiling, the direct, convective heat transfer from the wall to fluid phase k will be the same as that of a single-phase except the duct wall area over which this heat transfer can occur is weighted by the wetted fraction of the phase, κ_k. Thus the wall heat flux to phase k is

$$q_k^{\text{wall}} = \kappa_k \text{h}_k^{\text{wall}} \left(T_{\text{wall}} - T_k \right) . \tag{207}$$

The total heat flux is then $q^{\text{wall}} = q_\ell^{\text{wall}} + q_v^{\text{wall}}$. The wall heat power per unit length to phase k is then

$$Q_k^{\text{wall}} = q_k^{\text{wall}} P_{\text{hf}} , \tag{208}$$

where P_{hf} is perimeter over which wall heat transfer occurs. Similarly, the direct heat transfer from/to the interface to/from the phase k, which will also be used to determine the mass transfer between the phases, is

$$Q_{int,k} = h_{T,k} \left(T_{int} - T_k \right) A_{int} A \tag{209}$$

with $h_{T,k}$ denoting the convective heat transfer coefficient between the interface and phase k. The phasic bulk temperature T_k is determined from the respective phase's equation of state.

For wall heat flux sufficiently large to cause wall boiling a more elaborate model, which accounts for interphase mass transfer due to this wall heat transfer, must be utilized. Such a wall boiling model is detailed in the latter portion of the following section.

3.2.3 Interphase Mass Transfer and Wall Boiling

For a vapor to be formed from the liquid phase (vaporization) energy must be added to the liquid to produce vapor at nucleation sites; whether the liquid is heated directly or decompressed below its saturation pressure. A liquid to vapor phase change may occur based on two main mechanisms. The first is related to vaporization induced by external heating or heat transfer in a nearly constant pressure environment which is called heterogeneous boiling, or simply boiling. This heat input can occur through a solid/liquid interface with the solid typically hotter than the liquid, or through a liquid/gas interface with the gas being hotter than the liquid.

The second case corresponds to "flashing" vaporization such as cavitation induced by strong and rapid depressurization of the liquid phase (this is sometimes referred to as homogeneous boiling). In this relaxation process no extra energy is needed for the phase change; the necessary energy is already contained in the liquid phase in the form of internal energy. The process of phase change from vapor to liquid is known as condensation. The vapor condenses when it loses energy by heat transfer to a cool surface, but decompression of a saturated vapor also causes condensation at nucleation sites in the vapor. Nucleation sites are small particles or impurities in a fluid, or cavities or protrusions on a surface from which bubbles or droplets can grow during a change of phase. The phase change by condensation is similar to the first mechanism discussed and will be treated in the same manner.

To examine the mass flow rate between phases, local mechanisms of the vaporization (condensation) process are considered between the liquid phase and its associated vapor in the presence of temperature gradients. The mechanisms of interest here are dominated by heat diffusion at the interface. The pertinent local equations to consider are the mass and energy equations. As a vaporization front propagates slowly (on the order of 1 mm/s to 1 m/s) compared to acoustic waves present in the medium (which propagate with speeds of the order 1 km/s), acoustic propagation results in quasi-isobaric pressure evolution through vaporization fronts. The momentum equation is therefore not needed – because the quasi-isobaric assumption (neglecting the pressure and kinetic energy variations in the total energy equation) is made. The mass and energy balance equations are

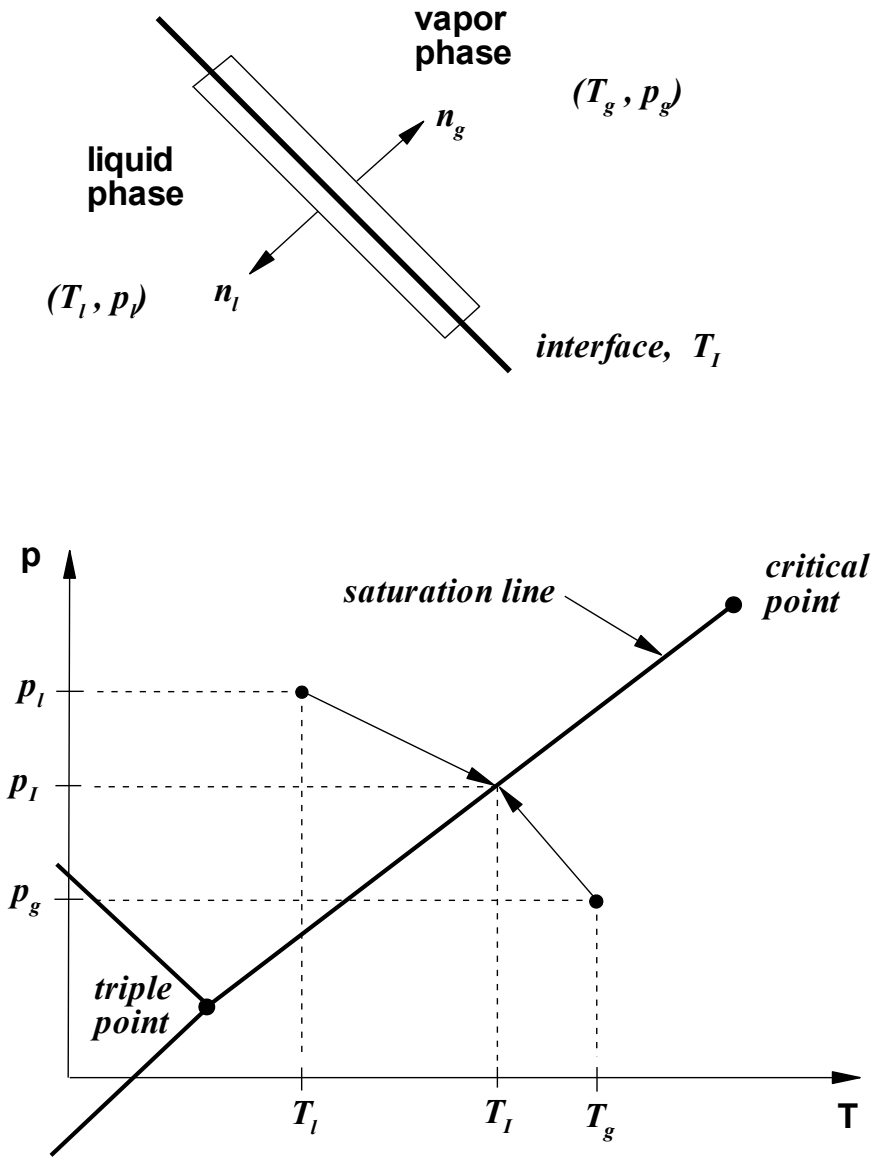

Figure 2. Interface control volume (top); T-p state space around saturation line, $T_{liq} < T_{vap}$, (bottom).

integrated over a "pill-box" control volume containing an interface (see e.g. Kuo [73]), as shown in the upper graphic of Figure 2, to obtain the algebraic "jump conditions"

$$\rho_{liq,int}\boldsymbol{u}_{liq,int}\cdot\hat{\boldsymbol{n}}_{liq}+\rho_{vap,int}\boldsymbol{u}_{vap,int}\cdot\hat{\boldsymbol{n}}_{vap}=0 \tag{210}$$

or

$$\Gamma_{liq}+\Gamma_{vap}=0 \tag{211}$$

for mass, and

$$\Gamma_{liq}h_{liq,int}+\boldsymbol{q}_{liq,int}\cdot\hat{\boldsymbol{n}}_{liq}+\Gamma_{vap}h_{vap,int}+\boldsymbol{q}_{vap,int}\cdot\hat{\boldsymbol{n}}_{vap}=0 \tag{212}$$

for energy, where the subscript int denotes the interface location. For convection dominated heat flux at the interface, and using local equilibrium conditions between phases at the local interface (equality of pressure and temperature), the heat fluxes can be defined as

$$\begin{aligned}\boldsymbol{q}_{k,int}\cdot\hat{\boldsymbol{n}}_k&=-k_k\nabla T_{k,int}\cdot\hat{\boldsymbol{n}}_k\\&=h_{T,k}\left(T_{int}-T_k\right)\end{aligned} \tag{213}$$

where T_{int} is the common interface temperature of phases. Combining these relations gives a simple expression for the interphase mass flow rate

$$\begin{aligned}\Gamma=\Gamma_{vap}&=\frac{h_{T,liq}\left(T_{liq}-T_{int}\right)+h_{T,vap}\left(T_{vap}-T_{int}\right)}{h_{vap,int}-h_{liq,int}}\\&=\frac{h_{T,liq}\left(T_{liq}-T_{int}\right)+h_{T,vap}\left(T_{vap}-T_{int}\right)}{L_v\left(T_{int}\right)}\end{aligned} \tag{214}$$

where $L_v\left(T_{int}\right)=h_{vap,int}-h_{liq,int}$ represents the latent heat of vaporization. The interface temperature is determined by the saturation constraint $T_{int}=T_{sat}(p)$ with the appropriate pressure $p=\bar{p}_{int}$ determined above, the interphase mass flow rate is thus determined. The lower graphic of Figure 2, schematically shows the p-T state space in the vicinity of the saturation line (shown for the case with $T_{liq}<T_{vap}$).

To better illustrate the model for vaporization or condensation, Figure 3 shows pure liquid and pure vapor regions separated by an interface. Representative temperature profiles are shown for heat transfer from vapor to liquid or liquid to vapor. As discussed by Moody [1], either vaporization or condensation can occur for both temperature profiles. The interphase mass transfer is determined by the net interfacial heat transfer: if net heat transfer is toward the interface, vapor will form; conversely, if net heat transfer is away from the interface, liquid will condense. Figure 3 shows heat transfer rates q_{vap} and q_{liq}

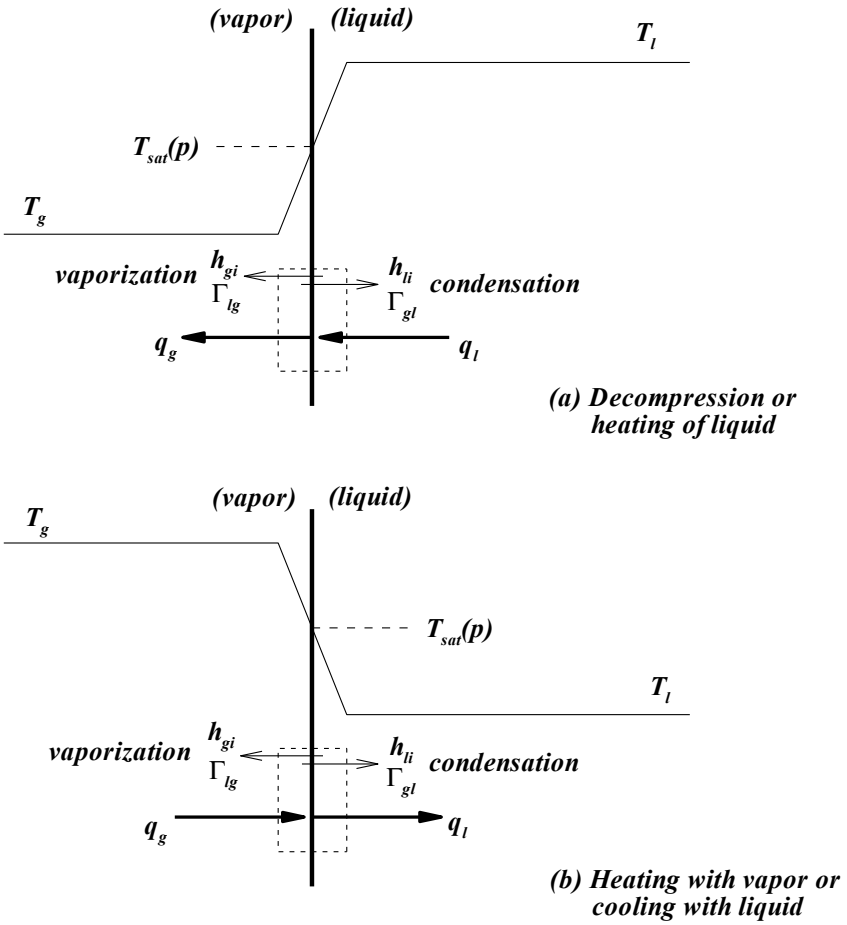

Figure 3. Vaporization and condensation at a liquid-vapor interface (after Moody [1]).

from the vapor and liquid sides of the interface. For bidirectional phase change (vaporization and condensation), mass transfer based on heat balance at the interface is adopted. When vaporization occurs, vapor is assumed to form at a saturated interface temperature $T_{int} = T_{sat}(\bar{p}_{int})$. If condensation occurs, liquid is assumed to form also at a saturated interface temperature $T_{int} = T_{sat}(\bar{p}_{int})$. The interfacial total enthalpies correspond to the saturated values in order that the interphase mass transfer rate and conservation of total energy be compatible:

$$H_{k,int} = h_{k,int} + \frac{1}{2}u_{int}^2 \tag{215}$$

for phase $k = (liq, vap)$, where $h_{k,int}$ is the phase k specific enthalpy evaluated at the interface condition. Phasic specific enthalpy depends upon the equation of state used and will be discussed with the equations of state. The interfacial density corresponds to the liquid saturated density $\rho_{int} = \rho_{liq,sat}(p_{int})$.

To summarize, the total saturated phasic enthalpies are constructed as

$$H_{liq,sat} = h_{liq,sat} + 0.5v_{int}^2 \tag{216}$$

$$H_{vap,sat} = h_{vap,sat} + 0.5v_{int}^2 \tag{217}$$

along with the total heat of vaporization at $T_{int} = T_{sat}$

$$L_{tot}(T_{sat}) = H_{vap,sat} - H_{liq,sat}. \tag{218}$$

Notice that this step was not really necessary in this case $L_{tot}(T_{sat})$ and $L_{vap}(T_{sat})$, which was obtained from the equations of state (see above), are identical. The interphase mass transfer rate (per unit interfacial area) per unit volume coming from the liquid phase across the interfacial area can now be determined from

$$\Gamma_{int,vap} = \frac{h_{conv,liq}(T_{liq} - T_{int}) + h_{conv,vap}(T_{vap} - T_{int})}{L_{vap}(T_{sat})}. \tag{219}$$

At this point, all information necessary to compute the interface energy transfer due to mass transfer as well as the direct energy transfer has been described, i.e.

Liquid energy equation terms

$$+ \Gamma_{int,vap}(A_{int})(\frac{p_{int}}{\rho_{int}} - H_{liq,sat})A + (A_{int})h_{conv,liq}(T_{int} - T_{liq})A \tag{220}$$

Vapor energy equation terms

$$- \Gamma_{int,vap}(A_{int})(\frac{p_{int}}{\rho_{int}} - H_{vap,sat})A + (A_{int})h_{conv,vap}(T_{int} - T_{vap})A. \tag{221}$$

Because our two-phase model is cast as a 1-D variable cross-sectional area model, in order to capture realistic multidimensional physical phenomena such as boiling and frictional shear stress that occur at the wall, additional mechanistic terms must be added. Simple forms of these terms will be described in the balance of this section and in the next section.

Thus, it is first noted that additional vapor will be generated at the wall, $\Gamma^{\text{wall}}_{\ell\to v}$, due to local wall boiling such that $\Gamma_{\ell\to v}A = \Gamma^{\text{int}}_{\ell\to v}A_{int}A + \Gamma^{\text{wall}}_{\ell\to v}P_{\text{hf}}$.

To describe this additional wall mass transfer term, $\Gamma^{\text{wall}}_{\ell\to v}$, a wall-boiling model has been incorporated into RELAP-7 in which the wall heat flux is first partitioned into a portion which may go directly to convective heat transfer to the vapor phase and a portion which is available to both convectively heat the liquid phase and generate vapor via wall boiling. This partitioning is specified with a simple function of the liquid volume fraction $\kappa(\alpha_l)$. The portion of the wall heat flux available to convectively heat the liquid phase and generate vapor is further partitioned into a portion which may convectively heat the liquid phase and a portion which goes toward generation of vapor by first bringing a portion of the liquid to the saturation condition then bringing it to the saturated vapor condition. This partitioning fraction f_{boil} depends upon the wall temperature and the saturated liquid temperature, i.e. $f_{\text{boil}}(T_{wall}, T_{sat,liq})$. Rendering this into equation form,

$$\begin{aligned} Q_{wall,total} &= Q_{wall,vap} + Q_{wall,liq} \\ &= Q_{wall,vap} + Q_{wall,liq,conv} + Q_{wall,liq,boil} \end{aligned} \tag{222}$$

where for a simple model in RELAP-7

$$\begin{aligned} Q_{wall,vap} &= h_{wall,vap}(T_{wall} - T_{vap})(1-\kappa)a_{wall}A \\ &= h_{wall,vap}(T_{wall} - T_{vap})(1-\kappa)P_{hf} \end{aligned} \tag{223}$$

$$\begin{aligned} Q_{wall,liq} &= h_{wall,liq}(T_{wall} - T_{liq})\,\kappa\,P_{hf} \\ &= Q_{wall,liq,conv} + Q_{wall,liq,boil} \\ &= (1-f_{\text{boil}})Q_{wall,liq} + f_{\text{boil}}Q_{wall,liq} \end{aligned} \tag{224}$$

so

$$Q_{wall,liq,conv} = (1-f_{\text{boil}})h_{wall,liq}(T_{wall} - T_{liq})\,\kappa\,P_{hf} \tag{225}$$

$$Q_{wall,liq,boil} = f_{\text{boil}}h_{wall,liq}(T_{wall} - T_{liq})\,\kappa\,P_{hf} \tag{226}$$

It is emphasized that additional, perhaps more sophisticated or realistic, models will be described in subsequent sections. The rate of vapor generated by boiling at the wall due to wall heat flux is then

$$\Gamma^{\text{wall}}_{\ell\to v} P_{\text{hf}} = \frac{Q_{wall,liq,boil}}{h^{\text{int}}_v - h_\ell}, \tag{227}$$

where $h^{\text{int}}_v - h_\ell$ represents the change in specific energy required for vaporization, and the interfacial vapor specific enthalpy is defined as

$$h^{\text{int}}_v \equiv h_{\text{sat}}(\bar{p}_{\text{int}}) = h_v(T_{\text{sat}}(\bar{p}_{\text{int}}), \bar{p}_{\text{int}}) = h_v(T_{\text{int}}, \bar{p}_{\text{int}}). \tag{228}$$

The total vapor production is the sum of the vapor transferred from the liquid phase directly via the interfacial area in the bulk flow and the vapor produced at the wall:

$$\Gamma_{\ell\to v} A = \Gamma^{\text{int}}_{\ell\to v} A_{int} A + \Gamma^{\text{wall}}_{\ell\to v} P_{\text{hf}}.$$

Again it is recalled that the interface saturation temperature corresponds to the interface pressure p_{int}.

Here, for the simple RELAP-7 model, f_{boil} is zero while the wall temperature is less than the liquid saturation temperature corresponding to the liquid pressure and drops exponentially (90% variation over 9.2 degrees) for wall temperatures greater than the liquid saturation temperature, i.e.

$$\begin{aligned} T_{wall} \le T_{sat,liq} \quad &\Rightarrow \quad f_{\text{boil}} = 0 \\ T_{wall} > T_{sat,liq} \quad &\Rightarrow \quad f_{\text{boil}} = 1 - exp[-0.25(T_{solid,wall} - T_{sat,liq})]. \end{aligned} \tag{229}$$

In the above κ is defined to be zero for $\alpha_{liq} < 0.01$, ramp up linearly to a value of 1.0 at $\alpha_{liq} = 0.1$, then remain constant at a value 1.0 for $0.01 < \alpha_{liq} < 1.0$. Again, more realistic models will be described subsequently.

3.2.4 Wall and Interphase Friction

A simple wall friction model results from making the same assumptions as for single-phase duct flow with the exception that the duct wall area over which the shear stress acts

is multiplied by the fraction of the wall area which the phase k occupies, χ_k. Thus the linear friction force for two-phase flow is the following:

$$F^{\text{friction}}_{\text{wall},k} = \frac{\phi^{\text{friction}}_k f_{wall,k}}{d_h} \left(\frac{1}{2} \rho_k u_k \left| u_k \right| \right) \chi_k A \tag{230}$$

for phases $k = (liq, vap)$, where $f_{wall,k}$ is the single-phase wall friction factor associated with phase k and ϕ^{friction}_k is the two-phase multiplier for the friction factor for phase k. For a simple approximation of χ_k, one may assume that it is equal to the volume fraction α_k; however, in general, this is dependent on flow regime.

The frictional pressure drop in each phase will be different in general due the different velocities of the two phases. However, because of the tendency toward pressure equilibrium between the phases an effective pressure drop will be realized.

The friction force, or viscous drag, acting between the two phases due to their relative motion is also given in analogy to that of single-phase duct flow:

$$F_{\text{friction},k'} = f_{k,\,k'} \frac{1}{2} \rho_k (u_k - u_{int}) \left| u_k - u_{int} \right| A_{int} A \tag{231}$$

for $k = (liq, vap)$, $k' = (vap, liq)$, with $f_{k,k'}$ denoting the friction factor acting upon phase k due to the (relative) motion of the other phase k'. This equation is rewritten as

$$F_{friction,k'} = K_{k,k'} (u_{k'} - u_k) A. \tag{232}$$

For a simple model with bubbles and droplets on the ends of the phasic topological spectrum with an interpolation between these two for intermediate volume fractions, as was done for the interphase mass transfer above, the coefficient $K_{k,k'}$ is obtained after [74] by first determining effective bubble/droplet radius, r_0, as

if $\alpha_{vap} \leq \alpha_{vap,A}$: **Bubbles**

$$r_0 = r_{bub} \tag{233}$$

if $\alpha_{vap} \geq \alpha_{vap,B}$: **Droplets**

$$r_0 = r_{drop} \tag{234}$$

if $\alpha_{vap,A} < \alpha_{vap} < \alpha_{vap,B}$: **Linear Interpolation**

$$r_0 = r_{bub,A} + \frac{(r_{drop,B} - r_{bub,A})(\alpha_{vap} - \alpha_{vap,A})}{\alpha_{vap,B} - \alpha_{vap,A}}. \tag{235}$$

Then

$$K_{k,k'} = \frac{\rho A_{int}}{8}[C_D|u_k - u_{k'}| + \frac{12\hat{\nu}}{r_0}] \tag{236}$$

where

$$\rho = \alpha_{vap}\rho_{vap} + (1 - \alpha_{vap})\rho_{liq}$$
$$\hat{\nu} = \alpha_{vap}\hat{\nu}_{vap} + (1 - \alpha_{vap})\hat{\nu}_{liq} \ \ (kinematic\ viscosity) \tag{237}$$
$$C_D = 0.5. \tag{238}$$

As with the simple wall boiling model, in RELAP-7 this simple wall friction model can be replaced with more sophisticated and realist models which are described later.

3.2.5 Distributed Form Losses

To get the 2-phase distributed form loss term for phase k, one follows the same process as for single phase, given in Section 2.2.2, but pressure will no longer be with respect to the total cross-sectional area A, but with respect to the phasic cross-sectional area $\alpha_k A$. Additionally, in general, a two-phase multiplier ϕ_k^{form} is applied:

$$F_k^{\text{form}} = \phi_k^{\text{form}} K_k' \left(\frac{1}{2}\rho_k u_k \left|u_k\right|\right) \alpha_k A\,. \tag{239}$$

3.2.6 Nonequilibrium, Seven-Equation, Two-Phase Flow Model Summary

Combine the discussion from the previous sections with the conservation equations results in the phasic balance equations of mass, momentum, and total energy along with volume fraction evolution:

$$\frac{\partial \left(\alpha\rho\right)_{liq} A}{\partial t} + \frac{\partial \left(\alpha\rho u\right)_{liq} A}{\partial x} = -\Gamma^{\text{int}}_{\ell\rightarrow v} A_{int} A - \Gamma^{\text{wall}}_{\ell\rightarrow v} P_{\text{hf}} + \frac{\partial \mathfrak{f}_{liq}}{\partial x} \tag{240}$$

$$\begin{aligned}\frac{\partial \left(\alpha\rho u\right)_{liq} A}{\partial t} + \frac{\partial \alpha_{liq} A \left(\rho u^2 + p\right)_{liq}}{\partial x} &= p_{int} A \frac{\partial \alpha_{liq}}{\partial x} + p_{liq}\alpha_{liq}\frac{\partial A}{\partial x} \\ &+ A\lambda(u_{vap} - u_{liq}) \\ &- \Gamma^{\text{int}}_{\ell\rightarrow v} A_{int} u_{int} A - \Gamma^{\text{wall}}_{\ell\rightarrow v} u_{int} P_{\text{hf}} \\ &- F^{\text{friction}}_{\text{wall},\ell} - F_{\text{friction},vap} - F^{\text{form}}_{\ell} \\ &+ \left(\alpha\rho\right)_{liq} A\boldsymbol{g}\cdot\hat{n}_{axis} \\ &+ \frac{\partial \mathfrak{g}_{liq}}{\partial x}\end{aligned} \tag{241}$$

$$\begin{aligned}\frac{\partial \left(\alpha\rho E\right)_{liq} A}{\partial t} + \frac{\partial \alpha_{liq} u_{liq} A \left(\rho E + p\right)_{liq}}{\partial x} &= p_{int} u_{int} A \frac{\partial \alpha_{liq}}{\partial x} - \bar{p}_{int} A\mu(p_{liq} - p_{vap}) \\ &+ \bar{u}_{int} A\lambda(u_{vap} - u_{liq}) \\ &+ \Gamma^{\text{int}}_{\ell\rightarrow v} A_{int}\left(\frac{p_{int}}{\rho_{int}} - H_{liq,int}\right) A \\ &+ A_{int} h_{conv,liq}(T_{int} - T_{liq}) A \\ &+ Q^{\text{wall,conv}}_{\ell} \\ &- \Gamma^{\text{wall}}_{\ell\rightarrow v} H_{\ell} P_{\text{hf}} \\ &+ \left(\alpha\rho u\right)_{liq} A\boldsymbol{g}\cdot\hat{n}_{axis} \\ &+ \frac{\partial \left(\mathfrak{h}_{liq} + u_{liq}\mathfrak{g}_{liq}\right)}{\partial x}\end{aligned} \tag{242}$$

$$\begin{aligned}\frac{\partial \alpha_{liq} A}{\partial t} + u_{int} A \frac{\partial \alpha_{liq}}{\partial x} &= A\mu(p_{liq} - p_{vap}) - \frac{\Gamma^{\text{int}}_{\ell\rightarrow v} A_{int} A}{\rho_{int}} - \frac{\Gamma^{\text{wall}}_{\ell\rightarrow v} P_{\text{hf}}}{\rho_{int}} \\ &+ \frac{\partial \mathfrak{l}_{liq}}{\partial x}\end{aligned} \tag{243}$$

for the liquid phase, and

$$\frac{\partial\left(\alpha\rho\right)_{vap}A}{\partial t}+\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial x}=\Gamma^{\text{int}}_{\ell\rightarrow v}A_{int}A+\Gamma^{\text{wall}}_{\ell\rightarrow v}P_{\text{hf}}+\frac{\partial\mathfrak{f}_{vap}}{\partial x} \tag{244}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}A\left(\rho u^2+p\right)_{vap}}{\partial x}&=p_{int}A\frac{\partial\alpha_{vap}}{\partial x}+p_{vap}\alpha_{vap}\frac{\partial A}{\partial x}\\&+A\lambda(u_{liq}-u_{vap})\\&+\Gamma^{\text{int}}_{\ell\rightarrow v}A_{int}u_{int}A+\Gamma^{\text{wall}}_{\ell\rightarrow v}u_{int}P_{\text{hf}}\\&-F^{\text{friction}}_{\text{wall},v}-F_{\text{friction},liq}-F^{\text{form}}_{v}\\&+\left(\alpha\rho\right)_{vap}A\boldsymbol{g}\cdot\hat{n}_{axis}\\&+\frac{\partial\mathfrak{g}_{vap}}{\partial x}\end{aligned} \tag{245}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}u_{vap}A\left(\rho E+p\right)_{vap}}{\partial x}&=p_{int}u_{int}A\frac{\partial\alpha_{vap}}{\partial x}-\bar{p}_{int}A\mu(p_{vap}-p_{liq})\\&+\bar{u}_{int}A\lambda(u_{liq}-u_{vap})\\&-\Gamma^{\text{int}}_{\ell\rightarrow v}A_{int}\left(\frac{p_{int}}{\rho_{int}}-H_{vap,int}\right)A\\&+A_{int}h_{conv,vap}(T_{int}-T_{vap})A\\&+Q^{\text{wall}}_{v}+Q^{\text{wall,boil}}_{\ell}\\&+\Gamma^{\text{wall}}_{\ell\rightarrow v}H_{\ell}P_{\text{hf}}\\&+\left(\alpha\rho u\right)_{vap}A\boldsymbol{g}\cdot\hat{n}_{axis}\\&+\frac{\partial\left(\mathfrak{h}_{vap}+u_{vap}\mathfrak{g}_{vap}\right)}{\partial x}\end{aligned} \tag{246}$$

$$\begin{aligned}\frac{\partial\alpha_{vap}A}{\partial t}+u_{int}A\frac{\partial\alpha_{vap}}{\partial x}&=A\mu(p_{vap}-p_{liq})+\frac{\Gamma^{\text{int}}_{\ell\rightarrow v}A_{int}A}{\rho_{int}}+\frac{\Gamma^{\text{wall}}_{\ell\rightarrow v}P_{\text{hf}}}{\rho_{int}}\\&+\frac{\partial\mathfrak{l}_{vap}}{\partial x}\end{aligned} \tag{247}$$

for the vapor phase. The terms shown in red are viscous regularizations added as part of the *entropy viscosity method*, *EVM*. The entropy viscosity method is an approach to regularization which is applied purely to stabilized and to insure compatibility of the entropy

inequality when capturing discontinuities (shocks) with this otherwise hyperbolic system of equations. More will be said about these terms as well as their correct formulation in the subsequent *Numerical Methods* chapter.

3.2.7 Stiffened Gas Equation of State for Two-phase Flows

With the 7-equation two-phase model each phase is compressible and behaves with its own convex equation of state (EOS). For initial development purposes it was decided to use a simple form capable of capturing the essential physics. For this purpose the stiffened gas equation of state (SGEOS) [4] was selected

$$p(\rho, e) = (\gamma - 1)\rho(e - q) - \gamma p_\infty \tag{248}$$

where p, ρ, e, and q are the pressure, density, internal energy, and the binding energy of the fluid considered. The parameters γ, q, and p_∞ are the constants (coefficients) of each fluid. The first term on the right hand side is a repulsive effect that is present for any state (gas, liquid, or solid), and is due to molecular vibrations. The second term on the right represents the attractive molecular effect that guarantees the cohesion of matter in the liquid or solid phases. The parameters used in this SGEOS are determined by using a reference curve, usually in the $\left(p, \frac{1}{\rho}\right)$ plane.

LeMetayer [4] uses the saturation curves as this reference curve to determine the stiffened gas parameters for liquid and vapor phases. The SGEOS is the simplest prototype that contains the main physical properties of pure fluids, repulsive and attractive molecular effects, thereby facilitating the handling of the essential physics and thermodynamics with a simple analytical formulation. Thus each fluid has its own thermodynamics. For each phase the thermodynamic state is determined by the SGEOS:

$$e(p, \rho) = \frac{p + \gamma p_\infty}{(\gamma - 1)\rho} + q \tag{249}$$

$$\rho(p, T) = \frac{p + p_\infty}{(\gamma - 1)c_v T} \tag{250}$$

$$h(T) = \gamma c_v T + q \tag{251}$$

$$g(p, T) = (\gamma c_v - q')\, T - c_v T \ln \frac{T^\gamma}{(p + p_\infty)^{\gamma - 1}} + q \tag{252}$$

where T, h, and g are the temperature, enthalpy, and Gibbs free enthalpy, respectively, of the phase considered. In addition to the three material constants mentioned above, two additional material constants have been introduced, the constant volume specific heat c_v and the parameter q'. The method to determine these parameters in liquid-vapor systems, and in particular the coupling of liquid and vapor parameters, is given in [4]. The values for water and its vapor from that reference are given in Table 2. These parameter values appear to yield reasonable approximations over a temperature range from 298 to 473K.

Unlike van der Waals type modeling where mass transfer is a thermodynamic path, with the 7-equation two-phase model the mass transfer modeling, which produces a relaxation toward thermodynamic equilibrium, is achieved by a kinetic process. Thus the 7-equation model preserves hyperbolicity during mass transfer. From equation (251) it is readily seen that the phase k specific enthalpy evaluated at the interface condition from equation (215) is

$$h_{k,int} = c_{p,k} T_{int} + q_k \tag{253}$$

because $c_{p,k} = \gamma_k c_{v,k}$.

The bulk interphase mass transfer from the liquid phase to the vapor phase Γ is due to their difference in Gibb's free energy. At saturated conditions the Gibb's energies of the two-phases are equal. It is necessary to determine the saturation temperature $T_{sat}(p)$ for given pressure $p = \bar{p}_{int}$ and the heat of vaporization $L_v\left(T_{sat}(\bar{p}_{int})\right)$ at this saturation temperature with the SGEOS for each phase. For this calculation the procedure of [4] is adopted. This procedure for the determination of SGEOS parameters can be made very accurate provided the two reference states are picked sufficiently close to represent the experimental saturation curves as locally quasi-linear. Restrictions occur near the critical point, but away from this point wide ranges of temperatures and pressures can be considered. At thermodynamic equilibrium at the interface, the two phasic Gibbs free enthalpies must be equal, $g_{vap} = g_{liq}$, so the use of equation (252) yields

$$\ln\left(p + p_{\infty,vap}\right) = A + \frac{B}{T} + C\ln(T) + D\ln\left(p + p_{\infty,liq}\right) \tag{254}$$

where

$$A = \frac{c_{p,liq} - c_{p,vap} + q'_{vap} - q'_{liq}}{c_{p,vap} - c_{v,vap}} \tag{255}$$

$$B = \frac{q_{liq} - q_{vap}}{c_{p,vap} - c_{v,vap}} \tag{256}$$

$$C = \frac{c_{p,vap} - c_{p,liq}}{c_{p,vap} - c_{v,vap}} \tag{257}$$

$$D = \frac{c_{p,liq} - c_{v,liq}}{c_{p,vap} - c_{v,vap}} \ . \tag{258}$$

Relation (254) is nonlinear, but can used to compute the theoretical curve $T_{sat}(p)$. A simple Newton iterative numerical procedure is used. With $T_{sat}(p)$ determined, the heat of vaporization is calculated as

$$\begin{aligned} L_v\left(T_{int}\right) &= h_{vap,int} - h_{liq,int} \\ &= h_{k,int} \\ &= \left(\gamma_{vap} c_{v,vap} T + q_{vap}\right) - \left(\gamma_{liq} c_{v,liq} T + q_{liq}\right) \ . \end{aligned} \tag{259}$$

3.2.8 Spline Based Table Look-up Method with IAPWS-95 Equation of State for Steam and Water

For the simulation of two-phase flows with RELAP-7 accurate equations of state must be used to obtain the properties of steam and water. Moreover, for CPU-intensive numerical simulations with this code, thermodynamic and transport properties of steam and water are calculated extremely often. Because the dependent variables of the two-phase model partial differential equations are mass-, momentum-, and total energy-densities the thermodynamically independent variables of the required property functions are specific volume and specific internal energy (v, e). These are readily computed from the phasic dependent variables as

$$v_k = \frac{1}{\rho_k} = \frac{\alpha_k}{\left(\alpha\rho\right)_k}, \qquad k = \{liq, vap\} \tag{260}$$

$$e_k = \frac{\left(\alpha\rho E\right)_k}{\left(\alpha\rho\right)_k} - \frac{1}{2}\frac{\left(\alpha\rho u\right)_k^2}{\left(\alpha\rho\right)_k^2}, \qquad k = \{liq, vap\} \ . \tag{261}$$

Then other phasic properties are functions of these two phasic thermodynamic properties, e.g. pressure $p_k = f(v_k, e_k)$.

Determining properties as a function of (v, e) from an accurate equation of state such as IAPWS-95 would normally require backward functions for calculations from pressure and specific volume (p, v) and specific internal energy and specific entropy (e, s). This requires an iterative solution that is very time-consuming and not computationally efficient. Therefore, in the original development of RELAP-7 property calculations were simplified through the use of the stiffened gas equation of state for each phase. These simplifications cause, depending on the range of state, inaccuracies in the results of the reactor system simulation. To provide fast and accurate property calculation algorithms, RELAP-7 was modified to employ the Spline-Based Table Look-up (SBTL) Method [75] which was developed in a project of the International Association for the Properties of Water and Steam (IAPWS). With this method properties from existing accurate equations of state, such as IAPWS-95 for steam and water, can be reproduced with high accuracy and significantly reduced computational times. Under INL direction, the SBTL method based on the IAPWS-95 properties for steam and water was extensively modified for RELAP-7, by Matthias Kunick at Zittau/Goerlitz University of Applied Sciences [76], to allow the calculation of not just the equilibrium properties for the homogeneous equilibrium model (HEM), but also to provide the metastable properties that are needed by the 7-equation, nonequilibrium, two-pressure model.

Table look-up methods can be well-suited for fast and accurate property calculations. A table is populated with discrete values of the required properties which are calculated from an available equation of state such as IAPWS-95. During the simulation process, properties are determined from this look-up table through the use of simple interpolation and approximation algorithms. The Spline-Based Table Look-up (SBTL) method [75] applies polynomial spline interpolation techniques to reproduce the results of the IAPWS-95 equation of state with high accuracy and low computing time. It employs specialized coordinate transformations and simplified search algorithms to minimize the computing time and to optimize the look-up table for the desired accuracy [77].

For the numerical process simulations here, the continuous, piecewise-defined spline functions need additionally to be only once continuously differentiable. Therefore the SBTL method utilizes a simple bi-quadratic spline polynomial which offers the additional advantage of being analytically solvable in terms of the independent variables. This latter property allows the calculation of the inverse spline functions, i.e. the numerically consistent backward functions. Because the bi-quadratic polynomial spline has a constant second derivative which precludes its capture of changing curvature, SBTL method allows the

transformation of the variables of the interpolated function in order to minimize the third derivative, i.e. the coordinates are transformed in such manner that the change in curvature of the underlying function is reduced. This allows the spline polynomial to reproduce the transformed property function more easily and with greater accuracy [77]. For the version of SBTL utilized for RELAP-7, the specific internal energy *e* is not transformed while the specific volume is transformed as $\bar{v} = ln(v)$.

For example, a two-dimensional spline-based property function, such as pressure, for the liquid phase would be written $p^L(\bar{v}, e)$ while the same property for the vapor (gas) phase would be written $p^G(\bar{v}, e)$. In the RELAP-7 nonequilibrium, 7-equation two-phase model the phasic specific internal energies and phasic transformed specific volumes are passed, respectively, to compute each corresponding phasic property function. It is important to point out that for the 7-equation two-phase model, these phasic property functions can be either normal (equilibrium) single phase values or metastable (nonequilibrium) single phase values.

For the SBTL Method the spline function is created in transformed coordinates $(\bar{v}, e)$ and interpolates values from a logically rectangular set of discrete data points called *nodes*. Locally defined spline polynomials are defined over a local rectangular cells having *nodes* at their centers and *knots* at their four corners. Four polynomial cells are connected at each *knot*, see Figure 4. The equidistant nodes (in transformed space) are distributed in a manner to insure the required accuracy of the spline function over the full range of validity. An efficient search algorithm is employed to rapidly determine the grid cell in which an arbitrary $(\bar{v}, e)$ is located. The locally defined polynomial must intersect the cell node, e.g. $p^L_{i,j}(\bar{v}_i, e_j)$, while its partial state derivatives with respect to $\bar{v}$ and e must match at the right and left edges (located midway between the nodes in the horizontal direction) and, respectively, the top and bottom edges (located midway between nodes in the vertical direction). At the cell corners, *knots*, the cross derivatives of all four contiguous cells must match. The equations representing these conditions, the composite of all of these nine-point stencil cells, form a system of equations that are solved globally to yield the local polynomial coefficients for each cell [75] [78]. As an illustration, the pressure of one of the phases (the phasic subscript is supressed here purely for clarity of exposition) would be determined from the expression

$$p_{\{i,j\}}((\bar{v}, e) = \sum_{k=1}^{3} \sum_{l=1}^{3} a_{ijkl} (\bar{v} - \bar{v}_i)^{k-1} (e - e_j)^{l-1}$$

where a_{ijkl} are the polynomial spline coefficients.

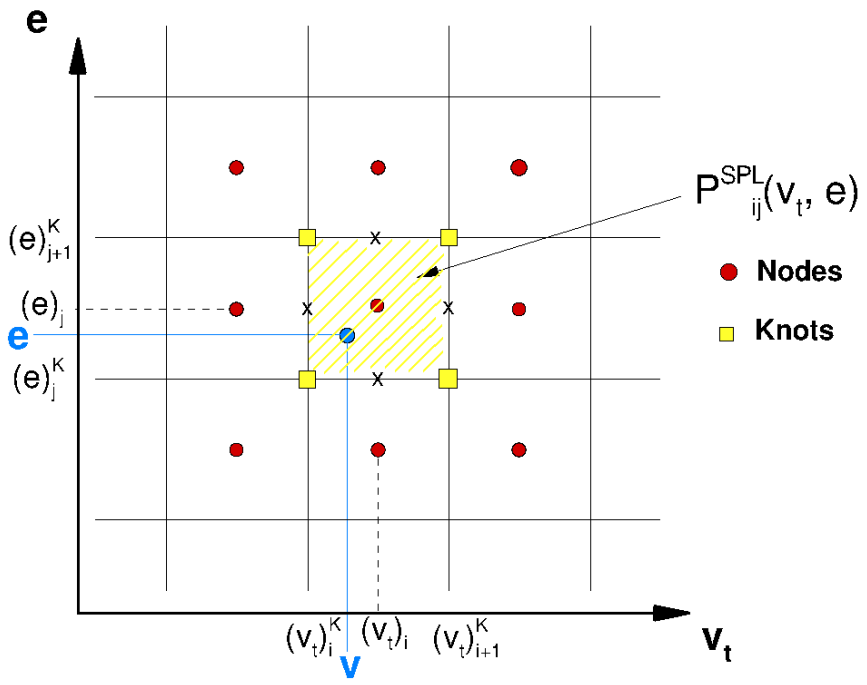

Figure 4. $(\bar{v}, e)$ state space spline polynomial cell $P^{SPL}_{ij}(\bar{v}, e)$ [note: $\bar{v}$ is denoted v_t], with node (center circle), knots (corner squares), and mid-points (edge x's) plus neighboring cells and nodes.

The SBTL method was applied to industrial formulation IAPWS-IF97 in [77] and tested in multidimensional CFD simulations of condensing steam in a turbine cascade. With this approach to obtaining real fluid properties the computing times were increased by a factor of only 1.4 over the same calculation using analytical ideal gas values, and these CFD simulations using the SBTL method were 6-10 times faster than using IAPWS-IF97 directly [77] (presumably in an iterative manner).

The following thermodynamic and transport properties are provided by the modified SBTL Package for equiliibrium mixture and for each phase (stable and metastable) as a function of respective phasic specific volume v and specific energy e, as well partial derivatives of the property with respect to v and e (it is noted that RELAP-7 employs implicit

temporal integration and needs a Jacobian which is based on these derivatives, which will be discussed in the following chapter describing the numerical methods it uses):

$p(v, e)$ – pressure
$T(v, e)$ – temperature
$w(v, e)$ – sound speed
$c_p(v, e)$ – isobaric specific heat
$c_v(v, e)$ – isochoric specific heat
$g(v, e)$ – Gibbs energy
$s(v, e)$ – specific entropy

$k(v, e)$ – thermal conductivity
$\nu(v, e)$ – dynamic viscosity
$\sigma(T(v, e))$ – surface tension.

For convenience the following functions are also provided:

$\rho(p, T)$ – mass density ($1/v$) as a function of pressure and temperature
$e(p, T)$ – specific internal energy as a function of pressure and temperature

along with their partial derivatives with respect to pressure p, and temperature T, and

$e(v, p)$ – specific internal energy as a function of specific volume and pressure
$e(s, p)$ – specific internal energy as a function of specific entropy and pressure
$v(s, p)$ – specific volume as function of specific entropy and pressure

along with their respective partial derivatives.

Lastly, the following functions of, and partial derivatives with respect to, pressure p and temperature T, at saturation condition are provided:

$T_{sat}(p)$ – saturation temperature as a function of pressure
$p_{sat}(T)$ – saturation pressure as function of temperature
δh_{vap} – heat of vaporization.

3.3 Homogeneous Equilibrium Model (HEM)

As remarked in Section 3.1.11 and Section 3.2.1, the 7-equation two-phase model naturally reduces to simpler models with appropriate relaxation. If mechanical relaxation, in which pressure and velocity are relaxed to a single value, is performed the 5-equation model of Kapila results. If, in addition, thermodynamic relaxation is performed, in which temperatures and Gibb's energies are relaxed to a single value, the 3-equation Homogeneous Equilibrium Model (HEM) is obtained. The HEM model is also know as the EVET (Equal Velocity, Equal Temperature) model, wherein it is implied that the pressures are equal and a saturated condition, which also implies that the Gibb's energies for the liquid and vapor phases are equal. The 3-equation HEM model is the simplest (at least from the balance equation viewpoint) and oldest of the two-phase model hierarchy [66], however some of its other properties, e.g. effective sound speed, are more difficult, and may even exhibit discontinuities in transitions from single- to two-phase.

For some applications where the HEM representation is physically appropriate, it may be more economical to begin with the 3-equation HEM model, rather than carrying the additional expense of a relaxed 7-equation model. The 3-equation HEM model is included also as a selectable model in RELAP-7. It is noted that a partially- or transitionally-relaxed 7-equation model will be very useful for coupling of the spatial regions where the unrelaxed 7-equation model is needed with spatial regions where the 3-equation HEM model may be used.

3.3.1 Field Equations

In the HEM model, the two phases in the mixture are assumed to be in thermodynamical and mechanical equilibrium and the pressure in the mixture is taken to be equal to the saturation pressure. Consequently, the two-phase mixture is effectively treated as a single (pseudo) fluid whose properties are suitable averages of the phasic properties of the individual phases. The balance equations for HEM are the same as those for the single-phase flow as shown in (20) through (23); but each primary variable now represents the state of a homogeneous mixture of two phases. Therefore, the primary variables are denoted with an overbar as $\bar{\rho}$, $\overline{\rho u}$, $\overline{\rho E}$ and $\overline{\rho s}$, where for example $\bar{\rho} = (1-\alpha)\rho_{liq,sat}(T) + \alpha\rho_{vap,sat}(T)$ is the mixture density and α is again the probability of presence (or volume fraction) of the vapor phase.

In the RELAP-7 solution of the HEM model, the primary variables are solved with

fully implicit time discretization and the vapor volume fraction is calculated with

$$\alpha = \frac{\bar{\rho} - \rho_{liq,sat}(T)}{\rho_{vap,sat}(T) - \rho_{liq,sat}(T)} \tag{262}$$

where $\rho_{liq,sat}(T)$ and $\rho_{vap,sat}(T)$ are the saturated density of liquid and vapor respectively for a given temperature T.

3.3.2 Constitutive Models

The same closure models are used for the HEM model as for the single-phase flow, such as wall friction coefficients and convective heat transfer coefficients, except that the following viscosity and thermal conductivity models are used:

$$\bar{\mu} = \mu_{liq}(1-\alpha) + \alpha\mu_{vap} \tag{263}$$

$$\bar{k} = k_{liq}(1-\alpha) + \alpha k_{vap}\ . \tag{264}$$

The stiffened gas equation of state discussed above for the single-phase and 7-equation two-phase model is used also for the HEM model.

4 Constitutive Models

4.1 Convective Heat Transfer

The wall heat source term Q^{wall} in Equation (22) (which is a linear power term having units [W/m]) in general can be composed of a number of sources:

$$Q^{\text{wall}} = \sum_i Q_i^{\text{wall}}\,. \tag{265}$$

In RELAP-7, a heat source Q_i^{wall} can be specified from one of the following:

- a known **wall temperature and wall heat transfer coefficient**:

$$Q_i^{\text{wall}} = \mathrm{h}_i^{\text{wall}}(T_i^{\text{wall}} - T)P_i\,, \tag{266}$$

where $\mathrm{h}_i^{\text{wall}}$ is the wall heat transfer coefficient associated with source i, T_i^{wall} is its wall temperature, and P_i is the perimeter of the wall surface bordering the flow channel.

- a known **wall heat flux**:

$$Q_i^{\text{wall}} = q_i^{\text{wall}} P_i \,, \tag{267}$$

where q_i^{wall} is the heat flux of source i.

Often various closures, such as determination of various categories of flow regime, require information of wall heat transfer quantities. Since these closures are not implemented on a per-heat-source basis, single values should be computed to represent the sum heat source for the flow channel. For instance, an aggregate heated perimeter is computed by summing the individual heated perimeters:

$$P_{\text{tot}} = \sum_i P_i \,. \tag{268}$$

A single wall heat transfer coefficient value is computed by weighting by the heated perimeter. Thus for single-phase flow,

$$\bar{\mathrm{h}}^{\text{wall}} = \frac{1}{P_{\text{tot}}} \sum_i \mathrm{h}_i^{\text{wall}} P_i \,, \tag{269}$$

and for two-phase flow,

$$\bar{\mathrm{h}}_k^{\text{wall}} = \frac{1}{P_{\text{tot}}} \sum_i \mathrm{h}_{k,i}^{\text{wall}} P_i \,. \tag{270}$$

RELAP-7 assumes that for a given flow channel, heat sources will come from source only of the temperature type or only of the heat flux type, since a mixture of specifications leads to some ambiguity in defining corresponding wall heat transfer coefficients. The following subsections detail how relevant averages are computed in each case.

4.1.1 Temperature-Specified

The average wall temperature is computed to preserve total wall heat flux. For 1-phase flow,

$$\bar{\mathrm{h}}^{\text{wall}} (\bar{T}^{\text{wall}} - T) P_{\text{tot}} = \sum_i \mathrm{h}_i^{\text{wall}} (T_i^{\text{wall}} - T) P_i \,, \tag{271}$$

and rearranging gives

$$\bar{T}^{\text{wall}} = T + \frac{\sum_i \mathrm{h}_i^{\text{wall}}(T_i^{\text{wall}} - T)P_i}{\bar{\mathrm{h}}^{\text{wall}} P_{\text{tot}}} \,. \tag{272}$$

For 2-phase flow,

$$\begin{aligned}\kappa_\ell \bar{\mathrm{h}}_\ell^{\text{wall}}(\bar{T}^{\text{wall}} - T_\ell)P_{\text{tot}} + \kappa_v \bar{\mathrm{h}}_v^{\text{wall}}(\bar{T}^{\text{wall}} - T_v)P_{\text{tot}} \\ = \sum_i \kappa_\ell \mathrm{h}_{\ell,i}^{\text{wall}}(T_i^{\text{wall}} - T_\ell)P_i + \sum_i \kappa_v \mathrm{h}_{v,i}^{\text{wall}}(T_i^{\text{wall}} - T_v)P_i \,,\end{aligned} \tag{273}$$

and rearranging gives

$$\bar{T}^{\text{wall}} = \frac{\sum_i \kappa_\ell \mathrm{h}_{\ell,i}^{\text{wall}}(T_i^{\text{wall}} - T_\ell)P_i + \sum_i \kappa_v \mathrm{h}_{v,i}^{\text{wall}}(T_i^{\text{wall}} - T_v)P_i + (\kappa_\ell \bar{\mathrm{h}}_\ell^{\text{wall}} T_\ell + \kappa_v \bar{\mathrm{h}}_v^{\text{wall}} T_v)P_{\text{tot}}}{(\kappa_\ell \bar{\mathrm{h}}_\ell^{\text{wall}} + \kappa_v \bar{\mathrm{h}}_v^{\text{wall}})P_{\text{tot}}} \,. \tag{274}$$

4.1.2 Heat-Flux-Specified

Similarly to the temperature case, the average heat flux is defined to preserve the sum of the heat source terms:

$$\bar{q}^{\text{wall}} P_{\text{tot}} = \sum_i q_i^{\text{wall}} P_i \,. \tag{275}$$

Rearranging gives the definition:

$$\bar{q}^{\text{wall}} = \frac{1}{P_{\text{tot}}} \sum_i q_i^{\text{wall}} P_i \,. \tag{276}$$

5 Numerical Methods

5.1 Spatial Discretization

5.1.1 Continuous Galerkin Finite Element Method

5.1.1.1 Formulation In this section, the continuous Galerkin finite element method used for spatial discretization method of the governing partial differential equations of the

7-equation two-phase model is briefly described. The temporal discretization and time integration used for this system will be described in the subsequent section. This discretized system of RELAP-7 is implemented through the MOOSE multiphysics framework [5]. For conciseness, the system of equations is recalled here in more compact form by considering for the two phases, denoted by the subscript k and j, the following form:

$$\partial_t \mathbf{U} + \boldsymbol{\nabla} \cdot \mathbf{F}\left(\mathbf{U}\right) = \mathbf{N}\left(\mathbf{U}\right) + \mathbf{R}\left(\mathbf{U}\right) + \mathbf{S}\left(\mathbf{U}\right) + \boldsymbol{\nabla} \cdot \mathbf{D}(\mathbf{U}) \boldsymbol{\nabla} \mathbf{U} \tag{277}$$

where $\mathbf{U} = [(\alpha A)_k,\, (\alpha \rho A)_k,\, (\alpha \rho \boldsymbol{u} A)_k,\, (\alpha \rho E A)_k, (\alpha \rho A)_j,\, (\alpha \rho \boldsymbol{u} A)_j,\, (\alpha \rho E A)_j]^T$ is the solution vector. The nomenclature has been abbreviated by using subscript k to denote the liquid phase and subscript j to denote the vapor phase. The inviscid *conservative Eulerian fluxes* are denoted $\mathbf{F}\left(\mathbf{U}\right)$. $\mathbf{N}\left(\mathbf{U}\right)$ contains the *non-conservative* differential terms (including duct wall pressure force and *Lagrangian fluxes*). $\mathbf{R}\left(\mathbf{U}\right)$ contains the *mechanical relaxation* terms. The source vector $\mathbf{S}\left(\mathbf{U}\right)$ contains the *thermodynamic relaxation* terms (interphase mass, energy, and heat transfer), as well as the interphase friction, duct wall friction, duct wall heat addition, body force (gravity) terms. These are given, respectively, by:

$$\mathbf{F} \equiv \begin{bmatrix} 0 \\ (\alpha \rho u A)_k \\ \left[\alpha \left(\rho u^2 + p\right) A\right]_k \\ \left[\alpha u (\rho E + p) A\right]_k \\ (\alpha \rho u A)_j \\ \left[\alpha \left(\rho u^2 + p\right) A\right]_j \\ \left[\alpha u (\rho E + p) A\right]_j \end{bmatrix}, \quad \mathbf{N} \equiv \begin{bmatrix} -A \boldsymbol{u}_{int} \cdot \boldsymbol{\nabla} \alpha_k \\ 0 \\ \alpha_k p_k \boldsymbol{\nabla} A + p_{int} A \boldsymbol{\nabla} \alpha_k \\ p_{int} A \boldsymbol{u}_{int} \cdot \boldsymbol{\nabla} \alpha_k \\ 0 \\ \alpha_j p_j \boldsymbol{\nabla} A + p_{int} A \boldsymbol{\nabla} \alpha_j \\ p_{int} A \boldsymbol{u}_{int} \cdot \boldsymbol{\nabla} \alpha_j \end{bmatrix},$$

$$\mathbf{R} \equiv \begin{bmatrix} A\mu \left(p_k - p_j\right) \\ 0 \\ A\lambda \left(\boldsymbol{u}_j - \boldsymbol{u}_k\right) \\ -\bar{p}_{int} A\mu \left(p_k - p_j\right) + \bar{u}_{int} A\lambda \left(\boldsymbol{u}_j - \boldsymbol{u}_k\right) \\ 0 \\ A\lambda \left(\boldsymbol{u}_k - \boldsymbol{u}_j\right) \\ -\bar{p}_{int} A\mu \left(p_j - p_k\right) + \bar{u}_{int} A\lambda \left(\boldsymbol{u}_k - \boldsymbol{u}_j\right) \end{bmatrix},$$

and $\mathbf{S}(\mathbf{U}) = [S_1(U),\, S_2(U),\, S_3(U),\, S_4(U),\, S_5(U),\, S_6(U),\, S_7(U)]^T$, where

$$
\begin{aligned}
S_1(U) =& -\frac{\Gamma_{int,j} A_{int} A}{\rho_{int}} - \frac{\Gamma_{wall,j}}{\rho_{int}} \\
S_2(U) =& -\Gamma_{int,j} A_{int} A - \Gamma_{wall,j} \\
S_3(U) =& -\Gamma_{int,j} A_{int} u_{int} A - \Gamma_{wall,j} u_{int} - F^{\text{friction}}_{\text{wall},k} - F_{\text{friction},j} - F^{\text{form}}_{k} + (\alpha\rho)_k A\boldsymbol{g}\cdot\hat{n}_{axis} \\
S_4(U) =& +\Gamma_{int,j} A_{int}\left(\frac{p_{int}}{\rho_{int}} - H_{k,int}\right) A + A_{int} h_{conv,k}(T_{int} - T_k) A + Q_{wall,k,conv} \\
& -\Gamma_{wall,j}\left(-\frac{p_{int}}{\rho_{int}} + h_{j,int} + \frac{u_{int}^2}{2}\right) + (\alpha\rho u)_k A\boldsymbol{g}\cdot\hat{n}_{axis} \\
S_5(U) =& +\Gamma_{int,j} A_{int} A + \Gamma_{wall,j} \\
S_6(U) =& +\Gamma_{int,j} A_{int} u_{int} A + \Gamma_{wall,j} u_{int} - F^{\text{friction}}_{\text{wall},j} - F_{\text{friction},k} - F^{\text{form}} j + (\alpha\rho)_j A\boldsymbol{g}\cdot\hat{n}_{axis} \\
S_7(U) =& -\Gamma_{int,j} A_{int}\left(\frac{p_{int}}{\rho_{int}} - H_{j,int}\right) A + A_{int} h_{conv,j}(T_{int} - T_j) A + Q_{wall,j} \\
& +\Gamma_{wall,j}\left(-\frac{p_{int}}{\rho_{int}} + h_{j,int} + \frac{u_{int}^2}{2}\right) + (\alpha\rho u)_j A\boldsymbol{g}\cdot\hat{n}_{axis}\,.
\end{aligned}
$$

The non-physical dissipative flux terms are given by $\boldsymbol{\nabla}\cdot D(\mathbf{U})\boldsymbol{\nabla}\mathbf{U}$. The viscous coefficients $\mathbf{D}(\mathbf{U})$ in these terms will be described later when discussing the *entropy viscosity method* for stabilization of this hyperbolic equation system. In the equations above, and in the finite element equations (278) to follow, a nomenclature (that is also useful for describing multidimensional systems) has been used wherein, for the quasi one-dimensional systems of RELAP-7, ∇ means $\frac{\partial}{\partial x}$.

To apply the continuous finite element method, Eq. (277) is multiplied by a test function $\mathbf{W}(\boldsymbol{r})$, integrated by parts and each integral is decomposed into a sum of spatial integrals over the domain Ω_e of each element e of the discrete mesh domain Ω. The following *weak form* is obtained:

$$
\begin{aligned}
\mathbb{R}(\mathbf{U}) \equiv \sum_e \int_{\Omega_e} \partial_t \mathbf{U}\,\mathbf{W} d\Omega_e - \sum_e \int_{\Omega_e} \mathbf{F}(\mathbf{U})\cdot\boldsymbol{\nabla}\mathbf{W} d\Omega_e + \int_{\partial\Omega} \mathbf{F}(\mathbf{U})\cdot\boldsymbol{n}\mathbf{W} \\
- \sum_e \int_{\Omega_e} \left(\mathbf{N}(\mathbf{U}) + \mathbf{R}(\mathbf{U}) + \mathbf{S}(\mathbf{U})\right)\mathbf{W} d\Omega_e \\
+ \sum_e \int_{\Omega_e} D(\mathbf{U})\boldsymbol{\nabla}\mathbf{U}\cdot\boldsymbol{\nabla}\mathbf{W} d\Omega_e - \int_{\partial\Omega} D(\mathbf{U})\boldsymbol{\nabla}\mathbf{U}\cdot\boldsymbol{n}\,\mathbf{W} = 0\,. \quad (278)
\end{aligned}
$$

The two boundary integral terms in (278), i.e. $\int_{\partial\Omega} \mathbf{F}(\mathbf{U}) \cdot \boldsymbol{n}\mathbf{W}$ and $\int_{\partial\Omega} D(\mathbf{U})\boldsymbol{\nabla}\mathbf{U} \cdot \boldsymbol{n}\,\mathbf{W}$ are evaluated at the boundaries of the domain and come from appropriate *boundary condition* approximations. Appropriate physical boundary conditions for finite elements, based upon, and consistent with, the *method of characteristics* will be discussed in the next chapter. The integrals over the elements Ω_e are evaluated using a numerical quadrature. The MOOSE framework provides a wide range of test functions and quadrature rules. Note that the test function $\boldsymbol{W}$ is not chosen arbitrarily. In particular, it is required that $\boldsymbol{W}$ come from the space of vector functions

$$\boldsymbol{W} \in \left\{ \begin{bmatrix} w \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ w \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ w \end{bmatrix} \right\} \tag{279}$$

where $w \in \mathcal{W}$ is a scalar test function. In the present work, and in general practice, the space $\mathcal{W}$ is taken to be (a subspace of) the Hilbert space $H^1(\Omega)$. This choice, for instance, guarantees enough smoothness that (278) makes sense. The approximate problem proceeds by selecting only test functions from a finite-dimensional subspace of $\mathcal{W}$, denoted by $\mathcal{W}^h$, and which is spanned by the basis $\{\phi_i\}$, $i = 1, \ldots, N$. Linear Lagrange polynomials are employed by RELAP-7 as test functions, from which second-order spatial convergence is obtained for smooth solutions.

Remark: As the dissipative terms are added to the basic balance equation for the 7-equation two-phase model purely for stabilization of the weak solution to the hyperbolic system, the last boundary integral term of (278) is neglected. The stabilization is needed only for the solution in the domain interior and not at the boundary points.

For the continuous Galerkin formulation of RELAP-7, the unknown functions of the solution vector, $\mathbf{U}$, are approximated in the same basis used for the test functions, i.e.

$$U_m^h(x,t) = \sum_{j=1}^{nnd} U_{m,j}(t)\phi_j(x) \quad for\ m = 1, 2, \cdots, 7\,. \tag{280}$$

where $j = 1$ is the first (terminal) node in the duct and $j = nnd$ (number of nodes) is the last (terminal) node in the duct. The coefficients $U_{m,j}(t)$ vary in time only, and comprise the solution vector, (at each iteration) for each dependent variable U_m and at each spatial node j, of the system of equations. The system of so-called "semi-discrete" equations resulting from the finite element procedure discussed above (they have been discretized in space, but the temporal derivatives remain in continuous form) are, in this incomplete

state, effectively a system of coupled ordinary differential equations (*ODEs*), which must be integrated in time to obtain the solution. In Section 5.2 the various time discretization methods employed in RELAP-7 to perform this temporal integration are discussed.

It is well-known that a continuous Galerkin discretization of this set of hyperbolic equations is equivalent to a central finite difference method for a certain choice of integration rule, and therefore will exhibit oscillatory instabilities unless some artificial dissipation is added, as state previously, to stabilize the method. In Section 5.1.1.2, the stabilization methods available in RELAP-7 are discussed, namely the *entropy viscosity method* (5.1.1.2.1) used for both single- and two-phase flows, and the *streamline upwind/Petrov-Galerkin method* (5.1.1.2.2) available only for single-phase flows in the present work.

5.1.1.2 Stabilization In review of solutions to nonlinear hyperbolic, initial-boundary value problems such as the single- and two-phase equation systems of RELAP-7, it is known that even with smooth initial data, the existence of a globally smooth solution may be violated because of the nonlinearity of the flux functions and other nonlinear terms. The concept of a weak solution is introduced to guarantee the existence of a global solution; however, the uniqueness of the solution(s) is lost because the problem may allow infinitely many weak solutions. An additional condition is usually imposed, which is called the "entropy condition," to select a unique solution from the infinitely many weak solutions. The unique solution is called the "entropy solution."

In the literature, although there are several different ways of defining the entropy condition, they are all equivalent in the sense that they select the same entropy solution. For numerical schemes, this entropy condition and solution is sought through utilization of so-called conservative formulations of the physically descriptive equations along with appropriate specification of an artificial viscosity, either added directly to the governing equations or implied by the discretization employed. That is, a discretization scheme is selected, or built, which is consistent with the entropy condition, thereby guaranteeing that the numerical computation faithfully captures the physically relevant solution.

It is not easy to satisfy the somewhat contradictory objectives of capturing singularities (like shocks or interfaces) without instability or numerical dispersion while also realizing better resolution where the solution is smooth. Consequently, a plethora of schemes fill the literature, all attempting to accomplish this, either better or more robustly. First order Godunov upwind schemes are overly dissipative while sophisticated higher order methods, which are typically a nonlinear combination of first order dissipative schemes and basic higher order schemes that are necessarily oscillatory, need to employ flux limiters to

prevent unphysical oscillations. Even linear hyperbolic equation systems can be problematic for numerical discretization schemes. For example, the well-known central difference method generally produces oscillations for simple linear advection.

It is well-known that the continuous Galerkin finite element method, as described in Section 5.1.1.1, is unstable when applied directly to hyperbolic systems of equations. It attempts to approximate potentially nonlinear discontinuous solutions with continuous, δ-mollified solutions as nearly as possible with the functional space selected and element spacing chosen [79]. For certain finite element spaces and integration rules, the central difference method and Galerkin finite element methods are equivalent. This spatial discretization is known to not produce sufficient entropy locally. To compensate, especially for equations in conservative form, the method attempts to achieve this through a train of entropy producing oscillations in the vicinity of the local entropy production deficit. For example, this discretization exhibits oscillations when applied to convection-dominated flows.

Remark: It is also pointed out that the first-order backward Euler time integration method (BDF) described above is known to inherently introduce an excessive $\mathcal{O}(\Delta t)$ artificial viscosity through its discretization error. Thus its use is **strongly discouraged** for simulation of transient flow phenomena.

Currently available options of solution stabilization for RELAP-7 application include the entropy viscosity method (EVM), streamline upwind/Petrov- Galerkin method (SUPG), Lapidus methodologies, and pressure-based stabilization methods. A selection of these schemes are described in the following sections.

5.1.1.2.1 Entropy Viscosity Method As an available option, RELAP-7 employs a new technique, introduced recently [80–83], which requires the explicit addition of artificial viscosity or dissipation terms to the equations while ensuring that the physical entropy minimum principle remains satisfied. Most modern solvers for hyperbolic equation systems now use Godunov methods employing Riemann or approximate Riemann solvers to capture the discontinuities, or shocks. However, both the methods of artificial viscosity (either explicitly included by the addition of dissipation terms or implicitly included through the inherent truncation error of the numerical scheme used) and Godunov methods are general shock capturing methods. The effect of either method is the introduction of an appropriate amount of entropy into the flow [84]. With the artificial viscosity methods, the entropy is added by the dissipation produced by the incorporated artificial viscosity. On the other hand, with Godunov methods the entropy is primarily added implicitly by the

presence of shock waves resulting from the Riemann problem. Actually, at least in those cases when it can be found explicitly, the shock Hugoniot curve (i.e. the shock pressure jump as a function of the shock velocity jump) closely resembles commonly used, early forms of explicitly added artificial shock viscosity [85].

Under INL direction, the viscous regularization for the 7-equation two-phase model of RELAP-7 was obtained by Delchini [86] at Texas A&M University, using the similar methodology to that for the Euler equations. The method consists of adding dissipative terms to the system of governing balance equations and in deriving an entropy equation for the regularized system. By adequately selecting these artificial viscous fluxes, the sign of the entropy production remains positive. Derivation of the viscous regularization for the 7-equation two-phase model can be achieved by considering either the phasic entropy equation or the total entropy equation. In the latter case, the minimum entropy principle can be established for the whole two-phase system but may not ensure positivity of the entropy equation for each phase. However, positivity of the total entropy equation can also be achieved by requiring that the minimum entropy principle holds for each phase. This stronger requirement has the advantage of ensuring consistency with the single-phase Euler equations when one of the phases disappears in the limit of phase disappearance. With the entropy viscosity method, the added viscous dissipation is controlled locally to be effective only where discontinuities or wiggles occur in the solution. When a shock is formed, entropy is produced, so this metric is utilized to locally increase the viscous dissipation. However, other discontinuities, such as contact surfaces or volume fraction discontinuities, do not produce entropy. These will be signaled to the artificial viscosity controller by additional metrics such as the jump or change in a solution variable gradient. Additional details regarding its application to the 7-equation two-phase model and to low Mach number flows are directly based upon INL-sponsored research of Delchini [87]. This entropy viscosity method is independent of the spatial discretization employed, so it can be used with the standard Galerkin, continuous Finite Element Method (FEM). Though shown below for the 7-equation two-phase model, the entropy viscosity method is available for use with single-phase flow systems as well.

The red terms in the balance equations for the non-equilibrium, 7-equation, two-phase model summarized in *Section 3.2.5*, $\mathfrak{f}_k, \mathfrak{g}_k, \mathfrak{h}_k$, and $\mathfrak{l}_k$, with $k = \{liq, vap\}$, are the added phasic viscous terms to be specified. The 7-equation model *without* the viscous regularization terms is, by design, entropy producing. The 7-equation model *with* the regularization terms must also be entropy producing. To verify the entropy production of the 7-equation with regularization terms we therefore need only consider the regularization terms. The phasic entropy equation with only the added regularization terms is

$$\alpha_k \rho_k A \frac{D_k s_k}{Dt} = \left[(\rho s_\rho)_k - (e s_e)_k\right] \frac{\partial \mathfrak{f}_k}{\partial x} - \rho_k^2 (s_\rho)_k \frac{\partial \mathfrak{l}_k}{\partial x} + (s_e)_k \frac{\partial \left(\mathfrak{h}_k + \frac{1}{2} u_k^2 \mathfrak{f}_k\right)}{\partial x} + (s_e)_k \left(\mathfrak{g}_k - \mathfrak{f}_k u_k\right) \frac{\partial u_k}{\partial x}. \quad (281)$$

where $\frac{D_k(\cdot)}{Dt}$ is the phase k material derivative. Because the right hand side of this equation must be greater than zero, by the minimum entropy principle, at a point where the entropy $s_k(\rho_k, e_k)$ reaches its minimum value, the gradient $\nabla_{\rho_k, e_k}(s_k)$ must be zero and the Laplacian $\Delta_{\rho_k, e_k}(s_k)$ must be positive; see e.g. [88]. It can be shown [89] that a way to ensure this principle is to require

$$\mathfrak{l}_k = \beta_k A \frac{\partial \alpha_k}{\partial x} \quad (282)$$

$$\mathfrak{f}_k = \alpha_k \kappa_k A \frac{\partial \rho_k}{\partial x} + \rho_k \mathfrak{l}_k \quad (283)$$

$$\mathfrak{g}_k = \alpha_k \mu_k \rho_k A \frac{\partial u_k}{\partial x} + \mathfrak{f}_k u_k \quad (284)$$

$$\mathfrak{h}_k = \alpha_k \kappa_k A \frac{\partial (\rho e)_k}{\partial x} - \frac{u_k^2}{2} \mathfrak{f}_k + (\rho e)_k \mathfrak{l}_k \quad (285)$$

where β_k, μ_k, and κ_k are positive coefficients to be specified (note: the phasic, subscripted parameter κ_k here is not to be confused with the un-subscripted variable κ appearing in the total energy balance equations).

Because two-phase flows may be found in a wide range of speeds, from extremely low-Mach subsonic (nearly incompressible) to supersonic, these three positive viscous coefficients are designed, from the scaled 7-equation model to ensure well-scaled dissipative terms over the entire range of Mach numbers of interest. When artificial viscosity techniques are used, sufficient artificial viscosity must be present in the shock and discontinuity regions to prevent spurious oscillations from forming in the numerical solution, but little

or no dissipation should be present where the solution is smooth. It is also imperative that viscosity coefficients scale properly to ensure recovery of the incompressible equations in the low-Mach asymptotic limit. Careful analysis has resulted in the following definitions for the viscous regularization coefficients:

$$\beta_k(x,t) = min\left(\beta_{k,e}(x,t), \beta_{k,max}(x,t)\right) \tag{286}$$
$$\mu_k(x,t) = min\left(\mu_{k,e}(x,t), \mu_{k,max}(x,t)\right) \tag{287}$$
$$\kappa_k(x,t) = min\left(\kappa_{k,e}(x,t), \kappa_{k,max}(x,t)\right) \tag{288}$$

where the definitions of the entropy viscosity coefficients with subscript e and the first-order viscosity coefficients (ceiling values) with subscript max are given, respectively, by

$$\beta_{k,e}(x,t) = h^2 \frac{max\left(|R_k^\alpha(x,t)|, J_k^\alpha\right)}{|s_k^\alpha - \bar{s}_k^\alpha|_\infty} \tag{289}$$

$$\mu_{k,e}(x,t) = h^2 \frac{max\left(|\tilde{R}_k(x,t)|, J_k\right)}{(1-\sigma(M_k))\,\rho_k w_k^2 + \sigma(M_k)\,\rho_k u_k^2} \tag{290}$$

$$\kappa_{k,e}(x,t) = h^2 \frac{max\left(|\tilde{R}_k(x,t)|, J_k\right)}{\rho_k w_k^2} \tag{291}$$

$$\beta_{k,max}(x,t) = \mu_{k,max}(x,t) = \kappa_{k,max}(x,t) = \frac{h}{2}\left(|u_k| + w_k\right) . \tag{292}$$

In the above

$$\tilde{R}_k\left(x,t\right) = \frac{D_k p_k\left(x,t\right)}{Dt} - w_k^2\left(x,t\right)\frac{D_k \rho_k\left(x,t\right)}{Dt} \tag{293}$$

$$R_k^\alpha\left(x,t\right) = \frac{\partial\left(A s_k\right)}{\partial t} + A u_{int}\frac{\partial s_k}{\partial x} \tag{294}$$

$$J_k = \left|u_k\right| \; max\left(J_e\left[\frac{\partial p_k}{\partial x}\right], w_k^2 J_e\left[\frac{\partial \rho_k}{\partial x}\right]\right) \tag{295}$$

$$J_k^\alpha = \left|u_{int}\right| \; J_e\left[\frac{\partial \alpha_k}{\partial x}\right] \tag{296}$$

and s_k^α denotes any *entropy function* of the volume fraction evolution equation, e.g. $s_k^\alpha = \frac{1}{2}\alpha_{liq}^2$. Also, $\bar{s}_k^\alpha$ denotes the average of s_k^α over the computational domain, i.e. $\bar{s}_k^\alpha$ is a function of time only. Note that s_k^α is not the same as the physical phasic entropies, s_k $k = \{liq,\, vap\}$. $J_e\left[\cdot\right]$ denotes the hybrid elemental jump in function $\left(\cdot\right)$,

$$J_e\left[\frac{\partial a}{\partial x}\right] = max\left\{\left|\left[\!\left[\frac{\partial a}{\partial x}\right]\!\right]_1\right|, \left|\left[\!\left[\frac{\partial a}{\partial x}\right]\!\right]_2\right|\right\} \tag{297}$$

where

$$\left[\!\left[\frac{\partial a}{\partial x}\right]\!\right]_1 = \left(\frac{\partial a}{\partial x}\right)_e - \left(\frac{\partial a}{\partial x}\right)_{e-1}$$

and

$$\left[\!\left[\frac{\partial a}{\partial x}\right]\!\right]_2 = \left(\frac{\partial a}{\partial x}\right)_{e+1} - \left(\frac{\partial a}{\partial x}\right)_e .$$

with subscripts 1 and 2 representing the two node points for the (linear) element e. Thus, $J_e\left[\frac{\partial a}{\partial x}\right]$ for generic variable a is constant over element e and has the same value for each quadrature point, q_p, in element e. In the equations above, h represents the element characteristic size (for example, when considering a cell of volume V belonging to a mesh of dimension r then $h = V^{\frac{1}{r}}$).

In the denominator of the equation for $\mu_{k,e}$ above, the parametric function $\sigma\left(M_k\right)$ is a weighting function designed to change the normalization, and thus the local dissipation, with varying flow Mach number; this parameter is important for the success of the entropy

viscosity method for all-speed flows. To produce a stabilization method valid for a wide range of Mach numbers, from very low-Mach to supersonic flows, the denominator of the equation for $\mu_{k,e}$ above should vary between $\rho_k u_k^2$ for non-isentropic flows and $\rho_k w_k^2$ for low-Mach flows. These two scalings (denominator terms) are combined via a smoothed, shifted Heaviside-type function, $\sigma\left(M_k\right)$, to give a smooth transition between these two states. One such function available in RELAP-7 (which varies smoothly between 0 and 1) is the following [86]:

$$\sigma(M) = \begin{cases} 0 & \text{if } M \leq M^{\text{thresh}} - a, \\ 1 & \text{if } M \geq M^{\text{thresh}} + a, \\ \frac{1}{2}\left(1 + \frac{M - M^{\text{thresh}}}{a} + \frac{1}{\pi}\sin\left(\frac{\pi(M - M^{\text{thresh}})}{a}\right)\right) & \text{otherwise,} \end{cases} \tag{298}$$

where M^{thresh} is a threshold Mach number value beyond which the flow is no longer considered to be low-Mach (default value is $M^{\text{thresh}} = 0.05$), M is the local Mach number, and the scalar a determines how rapidly the function $\sigma(M)$ changes in the vicinity of M^{thresh} (default, $a = 0.005$). Both M^{thresh} and a are, however, user specified inputs in RELAP-7.

This definition of the phasic viscosity coefficients takes advantage of the properties of the entropy residual that is peaked in the vicinity of the shock, whereby the high-order viscosity coefficient will saturate to the first-order viscosity coefficient that is known to be over-dissipative. Moreover, in regions where the numerical solution is smooth, the phasic viscosity coefficient will be equal to the high-order viscosity coefficient that will ensure higher order accuracy and also the correct low-Mach asymptotic limit.

5.1.1.2.2 Streamline Upwind/Petrov-Galerkin Method The Streamline Upwind/Petrov-Galerkin (SUPG) method is available in RELAP-7 for use with single-phase flows only. The SUPG method is introduced by first writing (20)–(22) from Section 2.1 in system notation as

$$\boldsymbol{R}(\boldsymbol{V}) \equiv \frac{\partial \boldsymbol{V}}{\partial t} + \frac{\partial \boldsymbol{G}}{\partial x} - \boldsymbol{S} = \boldsymbol{0} \tag{299}$$

where

$$\boldsymbol{V} \equiv \begin{bmatrix} \rho A \\ \rho u A \\ \rho E A \end{bmatrix} \qquad \boldsymbol{G} \equiv \begin{bmatrix} \rho u A \\ \left(\rho u^2 + p\right) A \\ \rho u H A \end{bmatrix} \tag{300}$$

and $\boldsymbol{S}$ comprises the remaining source (gravity, wall-heating, friction) terms. Note that a slightly different notation for the area conserved variables, $\boldsymbol{V}$, and flux, $\boldsymbol{G}$, has been

utilized because it will prove useful to refer to the non-area conserved variables in the discussion which follows. As in Section 5.1.1.1, the weak form proceeds by dotting (299) with an admissible test function $\boldsymbol{W}$, integrating over the domain Ω, and applying the divergence theorem. We then define

$$a(\boldsymbol{V}, \boldsymbol{W}) \equiv \int_{\Omega} \left(\frac{\partial \boldsymbol{V}}{\partial t} \cdot \boldsymbol{W} - \boldsymbol{G} \cdot \frac{\partial \boldsymbol{W}}{\partial x} - \boldsymbol{S} \cdot \boldsymbol{W} \right) \mathrm{d}\Omega + \int_{\Gamma} (\boldsymbol{G} \cdot \boldsymbol{W}) \, \hat{n}_x \, \mathrm{d}\Gamma \tag{301}$$

for subsequent use. To introduce the SUPG method, we begin by defining the non-area conserved variable and flux vectors

$$\boldsymbol{U} \equiv \begin{bmatrix} \rho \\ \rho u \\ \rho E \end{bmatrix} \qquad \boldsymbol{F} \equiv \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u H \end{bmatrix} . \tag{302}$$

In particular, note that $\boldsymbol{V} = A\boldsymbol{U}$ and $\boldsymbol{G} = A\boldsymbol{F}$. If $\boldsymbol{F}$ and $\boldsymbol{U}$ are continuous, the chain rule can be used to write

$$\frac{\partial \boldsymbol{F}}{\partial x} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}} \frac{\partial \boldsymbol{U}}{\partial x} \equiv \boldsymbol{A} \frac{\partial \boldsymbol{U}}{\partial x} . \tag{303}$$

The matrix $\boldsymbol{A}$ is known as the "flux Jacobian" matrix. The identities

$$\frac{\partial \boldsymbol{G}}{\partial x} = A \frac{\partial \boldsymbol{F}}{\partial x} + \frac{\partial A}{\partial x} \boldsymbol{F} \tag{304}$$

$$\boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} = \boldsymbol{A} \left(A \frac{\partial \boldsymbol{U}}{\partial x} + \frac{\partial A}{\partial x} \boldsymbol{U} \right) = A \frac{\partial \boldsymbol{F}}{\partial x} + \frac{\partial A}{\partial x} \boldsymbol{A}\boldsymbol{U} \tag{305}$$

can be combined to eliminate the $A\frac{\partial \boldsymbol{F}}{\partial x}$ terms and obtain

$$\frac{\partial \boldsymbol{G}}{\partial x} = \boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} + (\boldsymbol{F} - \boldsymbol{A}\boldsymbol{U}) \frac{\partial A}{\partial x} . \tag{306}$$

Substituting (306) into (299) then gives

$$\widetilde{\boldsymbol{R}}(\boldsymbol{V}) \equiv \frac{\partial \boldsymbol{V}}{\partial t} + \boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} + (\boldsymbol{F} - \boldsymbol{A}\boldsymbol{U}) \frac{\partial A}{\partial x} - \boldsymbol{S} = \boldsymbol{0} \tag{307}$$

which is the so-called "quasi-linear" form of (299).

A few remarks about (307) are warranted. First, in the special case where $\boldsymbol{F}$ is a "homogeneous function of degree 1," $\boldsymbol{F} = \boldsymbol{A}\boldsymbol{U}$, and the term in (307) which is proportional

to $\frac{\partial A}{\partial x}$ vanishes. The flux $\boldsymbol{F}$ is a homogeneous function of degree 1 for the ideal gas equation of state, but not for equations of state in general. It is relatively straightforward to show that

$$\boldsymbol{F} - \boldsymbol{A}\boldsymbol{U} = \begin{bmatrix} 0 \\ \hat{p} \\ u\hat{p} \end{bmatrix} \tag{308}$$

where

$$\hat{p} \equiv p - p_{,0}\rho - p_{,1}\rho u - p_{,2}\rho E \ , \tag{309}$$

and, for the one-dimensional Euler equations with a generic equation of state $p = p(U_0, U_1, U_2)$, the partial derivatives are denoted $p_{,i} \equiv \frac{\partial p}{\partial U_i}$, $i = 0, 1, 2$. For the stiffened gas equation of state, we can use the partial derivatives discussed in Section 2.2.4.4 to compute $\hat{p} = -\gamma p_\infty$. Finally, we note that the two forms of the residual, $\boldsymbol{R}$ and $\widetilde{\boldsymbol{R}}$, coincide if the exact solution $\boldsymbol{V}$ is smooth. Some solutions, e.g. with shocks, violate this assumption, but the SUPG method is nevertheless still applicable in such situations. The SUPG method may now be stated succinctly as: find $\boldsymbol{V}$ such that

$$a(\boldsymbol{V}, \boldsymbol{W}) + \sum_K \int_{\Omega_K} \boldsymbol{A}^T \frac{\partial \boldsymbol{W}}{\partial x} \cdot \boldsymbol{\tau}_{\text{SUPG}} \widetilde{\boldsymbol{R}}(\boldsymbol{V}) \, \mathrm{d}\Omega_K = 0 \tag{310}$$

for all admissible $\boldsymbol{W}$. In (310), $\boldsymbol{A}^T$ is the transpose of the flux Jacobian matrix, $\boldsymbol{\tau}_{\text{SUPG}}$ is in general a 3×3 matrix of solution-dependent stabilization parameters, and the second term of (310) is traditionally written as a sum of integrals over the finite elements Ω_K because of the possibility of higher-order derivatives in $\widetilde{\boldsymbol{R}}$, although there are no such higher derivatives present in the current work. The method (310) is said to be "consistent" in the following sense: if the true solution $\boldsymbol{V}$ (which satisfies (299) pointwise and the weak form (301)) is smooth, then it also satisfies (307), and therefore the additional stabilizing term is zero.

The "stabilizing" effects of (310) come specifically from the inviscid flux terms of the quasi-linear residual (307), i.e.

$$\begin{aligned} \int_{\Omega_K} \boldsymbol{A}^T \frac{\partial \boldsymbol{W}}{\partial x} \cdot \boldsymbol{\tau}_{\text{SUPG}} \widetilde{\boldsymbol{R}}(\boldsymbol{V}) \, \mathrm{d}\Omega_K &= \int_{\Omega_K} \boldsymbol{A}^T \frac{\partial \boldsymbol{W}}{\partial x} \cdot \boldsymbol{\tau}_{\text{SUPG}} \left(\ldots + \boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} + \ldots \right) \mathrm{d}\Omega_K \\ &= \int_{\Omega_K} \frac{\partial \boldsymbol{W}}{\partial x} \cdot \left(\boldsymbol{A} \boldsymbol{\tau}_{\text{SUPG}} \boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} \right) \mathrm{d}\Omega_K + \ldots \end{aligned} \tag{311}$$

where the ellipsis are used to represent other terms in the quasi-linear residual which do not lead to stabilization, but are nevertheless required for consistency. The matrix $\boldsymbol{M} \equiv \boldsymbol{A}\boldsymbol{\tau}_{\text{SUPG}}\boldsymbol{A}$ can be thought of as the "artificial diffusivity" tensor associated with the method. Thus, a major design goal of the SUPG method is to pick $\boldsymbol{\tau}_{\text{SUPG}}$ in such a way that $\boldsymbol{M}$ is:

1. $\mathcal{O}(h)$ in size, so the scheme retains the Galerkin method's order of accuracy.
2. Positive-definite, to mimic a physical diffusion tensor.

Most of the effort and "art" in implementing the SUPG method is therefore concerned with choosing $\boldsymbol{\tau}_{\text{SUPG}}$ appropriately. For advection-dominated one-dimensional systems of conservation equations, Hughes et. al [90] have shown that a possible form for the stabilization operator $\boldsymbol{\tau}_{\text{SUPG}}$ is

$$\boldsymbol{\tau}_{\text{SUPG}} = \frac{h}{2}|\boldsymbol{A}|^{-1} \tag{312}$$

where h is element length, and the absolute value of a $\boldsymbol{A}$ is defined as

$$|\boldsymbol{A}| \equiv \boldsymbol{P}|\boldsymbol{D}|\boldsymbol{P}^{-1} \tag{313}$$

where $\boldsymbol{D}$ is a diagonal matrix of eigenvalues of $\boldsymbol{A}$ and $\boldsymbol{P}$ is a matrix whose columns are $\boldsymbol{A}$'s eigenvectors. The absolute value of a diagonal matrix $\boldsymbol{D}$ is defined simply by taking the absolute value of each of the entries on the diagonal. For the one-dimensional Euler equations with a generic equation of state $p = p(U_0, U_1, U_2)$ having partial derivatives $p_{,i} \equiv \frac{\partial p}{\partial U_i}$, $i = 0, 1, 2$, we have:

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 \\ p_{,0} - u^2 & p_{,1} + 2u & p_{,2} \\ u\left(p_{,0} - H\right) & up_{,1} + H & u\left(1 + p_{,2}\right) \end{bmatrix}. \tag{314}$$

The eigenvalues of the matrix defined in (314) are given by

$$\lambda_1 = u \tag{315}$$

$$\lambda_{2,3} = u + \frac{p_{,1} + up_{,2}}{2} \pm \frac{\left[4\left(p_{,0} + up_{,1} + Hp_{,2}\right) + \left(p_{,1} + up_{,2}\right)^2\right]^{1/2}}{2} \tag{316}$$

The eigenvalues (316) will be real (and hence the system will be hyperbolic) only if the term under the square root sign is ≥ 0. It may be readily verified that, for a given equation of state, (316) reduces to $\lambda_{2,3} = u \pm c$, where c is the local sound speed. In general, the

form (316) is preferred because it explicitly demonstrates the intrinsic role of the equation of state in determining the eigenvalues of $\boldsymbol{A}$.

The matrix of eigenvectors of $\boldsymbol{A}$ is given by

$$\boldsymbol{P} \equiv \begin{bmatrix} c_1 & c_3 & c_2 \\ \lambda_1 c_1 & \lambda_2 c_3 & \lambda_3 c_2 \\ 1 & 1 & 1 \end{bmatrix} \tag{317}$$

where

$$c_1 \equiv \frac{-p_{,2}}{p_{,0} + \lambda_1 p_{,1}} \tag{318}$$

$$c_j \equiv \frac{-\lambda_j}{d_j} \quad , \quad j = 2, 3 \tag{319}$$

$$d_j \equiv (H - u^2)(u p_{,2} - \lambda_j) + u(p_{,0} - u^2) \quad , \quad j = 2, 3 \ . \tag{320}$$

Its inverse is

$$\boldsymbol{P}^{-1} \equiv \frac{1}{\det \boldsymbol{P}} \begin{bmatrix} \lambda_2 c_3 - \lambda_3 c_2 & c_2 - c_3 & c_2 c_3 (\lambda_3 - \lambda_2) \\ \lambda_3 c_2 - \lambda_1 c_1 & c_1 - c_2 & c_1 c_2 (\lambda_1 - \lambda_3) \\ \lambda_1 c_1 - \lambda_2 c_3 & c_3 - c_1 & c_1 c_3 (\lambda_2 - \lambda_1) \end{bmatrix} \tag{321}$$

where

$$\det \boldsymbol{P} \equiv c_1(c_2 - c_3)\lambda_1 + c_3(c_1 - c_2)\lambda_2 + c_2(c_3 - c_1)\lambda_3 \ . \tag{322}$$

The preceding discussion provides all the information necessary to implement the SUPG scheme (310). In addition to the code required to implement the Galerkin part of the finite element method, one needs new code to define the stabilization matrix and quasi-linear residuals, and code to assemble the new residual contributions. For effective preconditioning and to implement solvers other than the JFNK method, one also needs to compute Jacobian contributions for the new stabilization terms, but this procedure is not discussed in detail here.

5.1.1.2.3 Pressure Gradient Stabilization Method RELAP-7 has two stabilization schemes based on pressure, both described by Nithiarasu [91]. The first, titled "Method II" in [91], uses the pressure gradient to indicate where artificial viscosity should be added, whereas the second, titled "Method III" in [91], uses the Laplacian of pressure to indicate where artificial viscosity should be added.

Let the equation for solution component m of the two-phase system be denoted as

$$\frac{\partial u_m}{\partial t} + \nabla \cdot \mathbf{f}_m(\mathbf{u}) = s_m(\mathbf{u}) \,, \tag{323}$$

where u_m denotes solution component m, $\mathbf{u}$ denotes the full solution vector, $\mathbf{f}_m$ denotes the flux function for solution component m, and s_m is the source function for solution component m. Adding a parabolic regularization term, the equation becomes

$$\frac{\partial u_m}{\partial t} + \nabla \cdot \mathbf{f}_m(\mathbf{u}) = s_m(\mathbf{u}) + \nabla \cdot (\mu \nabla u_m) \,, \tag{324}$$

where μ is a positive artificial viscosity coefficient, determined by the chosen artificial dissipation method.

The pressure gradient stabilization scheme computes this artificial viscosity as

$$\mu \equiv \zeta h^2 \lambda_{\text{max}} \frac{|\nabla p|}{p} \,, \tag{325}$$

where ζ is a user-defined tuning parameter, h is the mesh size, λ_{max} is an estimate of the maximum wave speed, and the pressure p in the denominator serves as a normalization for the pressure gradient term. For the mass, momentum, and energy equations, the maximum wave speed is estimated as $\lambda_{\text{max}} = |\mathbf{v}| + c$, and for the volume fraction, it is estimated as $\lambda_{\text{max}} = |\mathbf{v}_{\text{I}}|$, where $\mathbf{v}_{\text{I}}$ is the interfacial velocity. Note that the artificial viscosity definition above is comparable to a Lax-Friedrichs-type viscosity coefficient:

$$\mu_{\text{LF}} \equiv \frac{1}{2} h \lambda_{\text{max}} \,, \tag{326}$$

so the pressure-based method definition of the artificial viscosity coefficient can be viewed as

$$\mu = \tilde{\zeta} \xi \mu_{\text{LF}} \,, \tag{327}$$

where $\tilde{\zeta} = 2\zeta$ and ξ is a dimensionless artificial dissipation indicator or "switch":

$$\xi \equiv h \frac{|\nabla p|}{p} \,. \tag{328}$$

5.1.1.2.4 Pressure Curvature Stabilization Method The pressure curvature stabilization scheme is the other pressure-based stabilization method described in Section 5.1.1.2.3.

Instead of using the magnitude of the pressure *gradient* as the indicator for needing dissipation, it uses the pressure *Laplacian*:

$$\xi \equiv h^2 \frac{|\Delta p|}{p}\,, \tag{329}$$

so the artificial viscosity coefficient definition becomes

$$\mu \equiv \zeta h^3 \lambda_{\max} \frac{|\Delta p|}{p}\,. \tag{330}$$

To recover the Laplacian of pressure, L^2-projection is used: the following equation is added to the PDE system:

$$\theta = \Delta p\,, \tag{331}$$

where θ is the L^2-projection, which is what is used in the computation of μ. Testing this equation with trial function i and integrating by parts gives

$$(\theta, \phi_i)_\Omega = -(\nabla p, \nabla \phi_i)_\Omega + \langle \nabla p, \phi_i \hat{\mathbf{n}} \rangle_{\partial\Omega}\,, \tag{332}$$

and finally, dropping the boundary integral gives

$$(\theta, \phi_i)_\Omega = -(\nabla p, \nabla \phi_i)_\Omega\,. \tag{333}$$

5.1.2 Reconstructed Discontinuous Galerkin Finite Element Method

5.1.2.1 Introduction The class of reconstructed discontinuous Galerkin (rDG) methods, termed as PnPm schemes, were originally introduced by Dumbser et al. [92–94], where Pn indicates that a piecewise polynomial of degree of n is used to represent a discontinuous Galerkin (DG) solution, and Pm represents a reconstructed polynomial solution of degree of m ($m \geq n$) that is used to compute the fluxes. The PnPm schemes are designed to enhance the accuracy of the DG method by increasing the order of the underlying polynomial solution. The beauty of the PnPm schemes is that they provide a unified formulation for both the finite volume (FV) and DG methods, and contain both the classical cell-centered FV and standard DG methods as two special cases of PnPm schemes. When n = 0, i.e., a piecewise constant polynomial is used to represent a numerical solution, P0Pm is equivalent to the classical high-order FV schemes, where a polynomial solution of degree m ($m \geq 1$) is reconstructed from a piecewise constant solution. When $m = n$, the reconstruction reduces to the identity operator, and the PnPn scheme yields a standard DG method. Many variants of the original rDG methods have been developed

in recent years, for example, the hybrid HWENO+DG schemes by Balsara et al. [95], the least-squares reconstruction-based DG schemes by Luo et al. [96–98], and the class of Green-Gauss reconstruction-based hybrid DG/FV schemes by Zhang et al. [99, 100]. All of these schemes are able to improve the spatial accuracy of the underlying DG methods without significant extra cost in storage and computing time. RELAP-7 uses a second-order FV method variant from the PnPn schemes, namely rDG(P0P1) [101], as a spatial discretization option.

5.1.2.2 Discretization The 1-D, variable-area Euler equations can be expressed as

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = \mathbf{s}(\mathbf{u}) \,, \tag{334}$$

where $\mathbf{f}(\mathbf{u})$ is the advective (inviscid) flux vector, $\mathbf{s}(\mathbf{u})$ is the source vector, and $\mathbf{u}$ is the conservative variable vector:

$$\mathbf{u} = \begin{bmatrix} \rho A \\ \rho u A \\ \rho E A \end{bmatrix} \tag{335}$$

where ρ denotes density, u denotes velocity, E denotes specific total energy, and A denotes the cross-sectional area of the flow channel. The advective (inviscid) flux vector $\mathbf{f}(\mathbf{u})$ is defined by

$$\mathbf{f}(\mathbf{u}) = \begin{bmatrix} \rho u A \\ (\rho u^2 + p) A \\ u(\rho E + p) A \end{bmatrix} \tag{336}$$

where p denotes pressure, which is given by an equation of state.

Integrating over each cell volume V_i gives

$$\int_{V_i} \frac{\partial \mathbf{u}}{\partial t} \, \mathrm{d}V + \int_{V_i} \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} \, \mathrm{d}V = \int_{V_i} \mathbf{s}(\mathbf{u}) \, \mathrm{d}V \,. \tag{337}$$

Let $\mathbf{u}_i$ denote the *cell-average* value:

$$\mathbf{u}_i(t) \equiv \frac{1}{|V_i|} \int_{V_i} \mathbf{u}(x,t) \, \mathrm{d}V \,. \tag{338}$$

Using the divergence theorem, the following equations can then be derived from Equation (337):

$$|V_i| \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \mathbf{f}(\mathbf{u}(x_{i+1/2})) - \mathbf{f}(\mathbf{u}(x_{i-1/2})) = \int_{V_i} \mathbf{s}(\mathbf{u}) \, \mathrm{d}V \,. \tag{339}$$

The approach for computing the fluxes will be described in subsequent sections.

5.1.2.3 Slope Reconstruction To achieve greater than first-order spatial accuracy, a piecewise linear solution is reconstructed in a given cell. The reconstruction is performed in terms of slopes of a set of primitive variables $\mathbf{w} = [p, u, T]$:

$$\overline{\mathbf{u}}_i(x,t) = \mathbf{u}(\overline{\mathbf{w}}_i(x,t))\,, \tag{340}$$

$$\overline{\mathbf{w}}_i(x,t) = \mathbf{w}_i(t) + (x - x_i)\boldsymbol{\Delta}\mathbf{w}_i(t) \qquad x \in (x_{i-1/2}, x_{i+1/2})\,, \tag{341}$$

where $\boldsymbol{\Delta}\mathbf{w}_i$ denotes the vector of slopes for each of the primitive variables in cell i. To prevent the formation of spurious oscillations in the numerical solution, slope limiters are used to adjust the slopes: $\boldsymbol{\Delta}\mathbf{w}_i \to \overline{\boldsymbol{\Delta}\mathbf{w}}_i$:

$$\overline{\mathbf{w}}_i(x,t) = \mathbf{w}_i(t) + (x - x_i)\overline{\boldsymbol{\Delta}\mathbf{w}}_i(t) \qquad x \in (x_{i-1/2}, x_{i+1/2})\,, \tag{342}$$

A few classic slope limiters, having the TVD (Total Variation Diminishing) property, will be described. For a complete description of the TVD property and and slope limitation, see [59]. Each of the slope limiters are component-wise, but the component subscript will be omitted.

5.1.2.3.1 Minmod Slope Limiter The *minmod* slope limiter yields the following slope:

$$\overline{\Delta w}_i = \mathrm{minmod}\left(\Delta w_{i-1/2}, \Delta w_{i+1/2}\right)\,, \tag{343}$$

where the one-sided slopes are defined as

$$\Delta w_{i+1/2} \equiv \frac{w_{i+1} - w_i}{x_{i+1} - x_i}\,, \tag{344}$$

and the minmod function is defined by

$$\mathrm{minmod}(a,b) = \begin{cases} a & \text{if } |a| < |b| \text{ and } ab > 0\,, \\ b & \text{if } |b| < |a| \text{ and } ab > 0\,, \\ 0 & \text{if } ab \le 0\,. \end{cases} \tag{345}$$

If a and b have the same sign, then this selects the one that is smaller in modulus, else it returns zero. Rather than defining the slope on the ith cell by always using the downwind difference (which would give the Lax–Wendroff method), or by always using the upwind

difference (which would give the Beam–Warming method), the minmod method compares the two slopes and chooses the one that is smaller in magnitude. If the two slopes have different sign, then the value w_i must be a local maximum or minimum, and it is easy to check in this case that we must set $\overline{\Delta w}_i = 0$ in order to satisfy the TVD condition. The minmod method does a fairly good job of maintaining good accuracy in the smooth hump and also sharp discontinuities in the square wave, with no oscillations. Sharper resolution of discontinuities can be achieved with other limiters that do not reduce the slope as severely as minmod near a discontinuity.

5.1.2.3.2 Superbee Slope Limiter The *superbee* slope limiter is computed as follows [102]:

$$\overline{\Delta w}_i = \text{maxmod}\left(\overline{\Delta w}_i^{(1)}, \overline{\Delta w}_i^{(2)}\right) , \tag{346}$$

where

$$\overline{\Delta w}_i^{(1)} = \text{minmod}\left(\Delta w_{i+1/2}, 2\Delta w_{i-1/2}\right) \tag{347}$$

$$\overline{\Delta w}_i^{(2)} = \text{minmod}\left(2\Delta w_{i+1/2}, \Delta w_{i-1/2}\right) . \tag{348}$$

Each one-sided slope is compared with twice the opposite one-sided slope. Then the maxmod function in Equation (346) selects the argument with larger modulus. In regions where the solution is smooth this will tend to return the larger of the two one-sided slopes, but will still be giving an approximation, and hence we expect second-order accuracy. The superbee limiter is also TVD in general.

With the superbee method, the discontinuity stays considerably sharper than with the minmod method. On the other hand, there is a tendency of the smooth hump to become steeper and squared off. This is sometimes a problem with superbee: by choosing the larger of the neighboring slopes it tends to steepen smooth transitions near inflection points.

5.1.2.3.3 MC Slope Limiter The *monotonized central-difference* (MC) slope limiter, which was proposed by van Leer [103], is computed as follows:

$$\overline{\Delta w}_i = \text{minmod}\left(\Delta w_i, 2\Delta w_{i-1/2}, 2\Delta w_{i+1/2}\right) , \tag{349}$$

where the central-difference slope is computed as

$$\Delta w_i \equiv \frac{w_{i+1} - w_{i-1}}{x_{i+1} - x_{i-1}} . \tag{350}$$

This compares the central difference of Fromm's method with twice the one-sided slope to either side. In smooth regions this reduces to the centered slope of Fromm's method and hence does not tend to artificially steepen smooth slopes to the extent that superbee does. The MC limiter appears to be a good default choice for a wide class of problems.

5.1.2.4 Numerical Flux This section describes how the fluxes in Equation (339) are approximated. Simply evaluating the flux with the reconstructed solution at the interface would result in linear instability; to avoid this, the fluxes are evaluated with an approximate Riemann solver, which computes the approximate flux resulting in a Riemann problem with the adjacent values being used as the left and right initial conditions:

$$\mathbf{f}(\mathbf{u}(x_{i+1/2})) \approx A_{i+1/2}\{\left(\frac{1}{A_{i+1/2}}\overline{\mathbf{u}}_{i,R}, \quad \frac{1}{A_{i+1/2}}\overline{\mathbf{u}}_{i+1,L}, \quad \mathbf{n}_{i,i+1}\right), \tag{351}$$

$$\overline{\mathbf{u}}_{i,R} \equiv \overline{\mathbf{u}}_i(x_{i+1/2}), \qquad \overline{\mathbf{u}}_{i,L} \equiv \overline{\mathbf{u}}_i(x_{i-1/2}), \tag{352}$$

where $A_{i+1/2}$ is the cross-sectional area at the interface between cells i and $i+1$, $\mathbf{n}_{i,i+1}$ is the normal vector in the direction of cell $i+1$ from cell i, and $\{$ is a numerical flux function, which here will be the HLLC approximate Riemann solver, described in Section 5.1.2.4.1. Note that the cross-sectional area $A_{i+1/2}$ appears outside the flux function because it is assumed that the area is continuous at the interface. With this assumption, area drops out of this formulation, and a flux function for a 1-D (non-variable-area) formulation can be used without modification.

If cell i is a boundary cell, then the exterior side value ($\overline{\mathbf{u}}_{i,R}$ for the right boundary cell, $\overline{\mathbf{u}}_{i,L}$ for the left boundary cell) is a ghost cell value generated from boundary data, if any.

5.1.2.4.1 Explicit HLLC Riemann Solver The explicit form of the HLLC flux function [104], which computes a flux given a left state $\mathbf{u}_L$ and a right state $\mathbf{u}_R$, is defined by

$$\{(\mathbf{u}_L, \mathbf{u}_R, \mathbf{n}) = \begin{cases} \mathbf{f}_L & \text{if } S_L > 0 \\ \mathbf{f}_L^* & \text{if } S_L \leq 0 < S_M \\ \mathbf{f}_R^* & \text{if } S_M \leq 0 \leq S_R \\ \mathbf{f}_R & \text{if } S_R < 0 \end{cases}, \tag{353}$$

where the supersonic fluxes are

$$\mathbf{f}_K \equiv \mathbf{f}(\mathbf{u}_K)n_x = \begin{bmatrix} (\rho u)_K n_x \\ (\rho u^2 + p)_K n_x \\ ((\rho E + p)u)_K n_x \end{bmatrix}, \tag{354}$$

and the subsonic fluxes are

$$\mathbf{f}_K^* \equiv \mathbf{f}(\mathbf{u}_K^*) n_x \,, \qquad \mathbf{u}_K^* = \begin{bmatrix} \rho_K^* \\ (\rho u)_K^* \\ (\rho E)_K^* \end{bmatrix} . \tag{355}$$

It is assumed that the star-state values for pressure and velocity are equal on the left and right:

$$p_L^* = p_R^* = p^* \,, \tag{356}$$

$$u_L^* = u_R^* = u^* \,. \tag{357}$$

The wave speeds S_L and S_R are given in Section 5.1.2.4.3, and the middle wave speed S_M is assumed to be equal to the normal component of the star-state velocity:

$$S_M = u^* n_x \,. \tag{358}$$

Applying Rankine-Hugoniot conditions to the Rankine-Hugoniot conditions for mass and momentum equations gives the definition of the star-state pressures:

$$p_K^* = \rho_K (q_K - S_K)(q_K - S_M) + p_K \,, \tag{359}$$

where q_K denotes the normal component of velocity:

$$q_K = u_K n_x \,, \tag{360}$$

where n_x is the x-component of the normal vector going from state "L" to state "R"; normally, $n_x = 1$, but the arguments for $\mathbf{u}_L$ and $\mathbf{u}_R$ could be switched when passed into the numerical flux function, in which case $n_x = -1$.

Then combining this with the assumptions given by Equations (356), (357), and (358) gives the middle wave speed:

$$S_M = \frac{\rho_R q_R (S_R - q_R) - \rho_L q_L (S_L - q_L) + p_L - p_R}{\rho_R (S_R - q_R) - \rho_L (S_L - q_L)} \,. \tag{361}$$

Putting everything together gives the star-state solutions:

$$\mathbf{u}_K^* = \begin{bmatrix} \rho_K^* \\ (\rho u)_K^* \\ (\rho E)_K^* \end{bmatrix} = \Omega_K \begin{bmatrix} (S_K - q_K)\rho_K \\ (S_K - q_K)(\rho u)_K + (p^* - p_K) n_x \\ (S_K - q_K)(\rho E)_K - p_K q_K + p^* S_M \end{bmatrix} , \tag{362}$$

where

$$\Omega_K \equiv \frac{1}{S_K - S_M} \,, \tag{363}$$

and the subsonic fluxes are

$$\mathbf{f}_K^* = \begin{bmatrix} \rho_K^* S_M \\ (\rho u)_K^* S_M + p^* n_x \\ ((\rho E)_K^* + p^*) S_M \end{bmatrix} . \tag{364}$$

5.1.2.4.2 Implicit HLLC Riemann Solver For the implicit scheme, we need to evaluate the Jacobian of the numerical flux function. To avoid re-evaluation of the Jacobian for each nonlinear iteration, instead computing it only once per time step, the Jacobian is evaluated at the old time:

$$\{(\mathbf{u}_L, \mathbf{u}_R, \mathbf{n}) = \begin{cases} \mathbf{f}_L^n + \left(\dfrac{\partial \mathbf{f}_L}{\partial \mathbf{u}_L}\right)^n \Delta \mathbf{u}_L & \text{if } S_L > 0 \\ (\mathbf{f}_L^*)^n + \left(\dfrac{\partial \mathbf{f}_L^*}{\partial \mathbf{u}_L}\right)^n \Delta \mathbf{u}_L + \left(\dfrac{\partial \mathbf{f}_L^*}{\partial \mathbf{u}_R}\right)^n \Delta \mathbf{u}_R & \text{if } S_L \leq 0 < S_M \\ (\mathbf{f}_R^*)^n + \left(\dfrac{\partial \mathbf{f}_R^*}{\partial \mathbf{u}_L}\right)^n \Delta \mathbf{u}_L + \left(\dfrac{\partial \mathbf{f}_R^*}{\partial \mathbf{u}_R}\right)^n \Delta \mathbf{u}_R & \text{if } S_M \leq 0 \leq S_R \\ \mathbf{f}_R^n + \left(\dfrac{\partial \mathbf{f}_R}{\partial \mathbf{u}_R}\right)^n \Delta \mathbf{u}_R & \text{if } S_R < 0 \end{cases} \tag{365}$$

where $\Delta \mathbf{u}_K \equiv \mathbf{u}_K^{n+1} - \mathbf{u}_K^n$ and the flux derivatives are shown later in this section.

In addition to freezing the Jacobian at the old time, the acoustic wave speeds S_L and S_R are frozen as done by Batten et al. [104], which was shown to be have robust convergence for smooth flows. In this section, terms will be derived as if this assumption has not been made, but the terms that cancel out with this assumption will be highlighted.

Many equations are symmetric with respect to the subscripts L and R; in these cases the subscript K is used to represent either subscript. The subscript J is also a generic subscript that can either be taken to mean either both L and R or the opposite of K, depending on the context.

The derivatives of the supersonic fluxes are computed as follows:

$$\frac{\partial \mathbf{f}}{\partial \mathbf{u}} = n_x \begin{bmatrix} 0 & 1 & 0 \\ -u^2 + \dfrac{\partial p}{\partial \rho} & 2u + \dfrac{\partial p}{\partial \rho u} & \dfrac{\partial p}{\partial \rho E} \\ -\dfrac{u}{\rho}(\rho E + p) + u\dfrac{\partial p}{\partial \rho} & \dfrac{1}{\rho}(\rho E + p) + u\dfrac{\partial p}{\partial \rho u} & u\left(1 + \dfrac{\partial p}{\partial \rho E}\right) \end{bmatrix}. \tag{366}$$

For the subsonic case, the HLLC Jacobian matrices are given by

$$\frac{\partial \mathbf{f}_K^*}{\partial \mathbf{u}_J} = \begin{bmatrix} \dfrac{\partial \rho_K^*}{\partial \mathbf{u}_J} S_M + \dfrac{\partial S_M}{\partial \mathbf{u}_J} \rho_K^* \\ \dfrac{\partial (\rho u)_K^*}{\partial \mathbf{u}_J} S_M + \dfrac{\partial S_M}{\partial \mathbf{u}_J} (\rho u)_K^* + \dfrac{\partial p^*}{\partial \mathbf{u}_J} n_x \\ \left(\dfrac{\partial (\rho E)_K^*}{\partial \mathbf{u}_J} + \dfrac{\partial p^*}{\partial \mathbf{u}_J} \right) S_M + \dfrac{\partial S_M}{\partial \mathbf{u}_J} ((\rho E)_K^* + p^*) \end{bmatrix}, \tag{367}$$

where in this context, J is taken to mean either L or R.

The S_M derivatives are computed as follows:

$$S_M = \frac{A}{B}, \tag{368a}$$

$$A = \rho_R q_R (S_R - q_R) - \rho_L q_L (S_L - q_L) + p_L - p_R, \tag{368b}$$

$$B = \rho_R (S_R - q_R) - \rho_L (S_L - q_L), \tag{368c}$$

$$\frac{\partial S_M}{\partial \mathbf{u}_K} = \frac{1}{B} \frac{\partial A}{\partial \mathbf{u}_K} - \frac{A}{B^2} \frac{\partial B}{\partial \mathbf{u}_K}, \tag{369}$$

$$\begin{aligned} \frac{\partial A}{\partial \mathbf{u}_K} = {} & \frac{\partial (\rho q)_R}{\partial \mathbf{u}_K} (S_R - q_R) + (\rho q)_R \left(\frac{\partial S_R}{\partial \mathbf{u}_K} - \frac{\partial q_R}{\partial \mathbf{u}_K} \right) \\ & - \frac{\partial (\rho q)_L}{\partial \mathbf{u}_K} (S_L - q_L) - (\rho q)_L \left(\frac{\partial S_L}{\partial \mathbf{u}_K} - \frac{\partial q_L}{\partial \mathbf{u}_K} \right) + \frac{\partial p_L}{\partial \mathbf{u}_K} - \frac{\partial p_R}{\partial \mathbf{u}_K}, \end{aligned} \tag{370}$$

$$\begin{aligned} \frac{\partial B}{\partial \mathbf{u}_K} = {} & \frac{\partial \rho_R}{\partial \mathbf{u}_K} (S_R - q_R) + \rho_R \left(\frac{\partial S_R}{\partial \mathbf{u}_K} - \frac{\partial q_R}{\partial \mathbf{u}_K} \right) \\ & - \frac{\partial \rho_L}{\partial \mathbf{u}_K} (S_L - q_L) - \rho_L \left(\frac{\partial S_L}{\partial \mathbf{u}_K} - \frac{\partial q_L}{\partial \mathbf{u}_K} \right). \end{aligned} \tag{371}$$

The derivatives of the star-state pressure can be expressed as the following, where J denotes the subscript of the opposite of K:

$$\frac{\partial p^*}{\partial \mathbf{u}_K} = \rho_J (S_J - q_J) \frac{\partial S_M}{\partial \mathbf{u}_K} + \rho_J (S_M - q_J) \frac{\partial S_J}{\partial \mathbf{u}_K}, \tag{372}$$

The remaining derivatives are the following, where J denotes either L or R:

$$\frac{\partial \rho_K^*}{\partial \mathbf{u}_J} = \frac{\partial \Omega_K}{\partial \mathbf{u}_J}(S_K - q_K)\rho_K + \Omega_K \left(\frac{\partial S_K}{\partial \mathbf{u}_J} - \frac{\partial q_K}{\partial \mathbf{u}_J} \right) \rho_K + \Omega_K (S_K - q_K) \frac{\partial \rho_K}{\partial \mathbf{u}_J} , \quad (373)$$

$$\frac{\partial \Omega_K}{\partial \mathbf{u}_J} = -\frac{1}{(S_K - S_M)^2} \left(\frac{\partial S_K}{\partial \mathbf{u}_J} - \frac{\partial S_M}{\partial \mathbf{u}_J} \right) , \quad (374)$$

$$\begin{aligned} \frac{\partial (\rho u)_K^*}{\partial \mathbf{u}_J} &= \frac{\partial \Omega_K}{\partial \mathbf{u}_J} \left((S_K - q_K)(\rho u)_K + (p^* - p_K) n_x \right) \\ &+ \Omega_K \left(\left(\frac{\partial S_K}{\partial \mathbf{u}_J} - \frac{\partial q_K}{\partial \mathbf{u}_J} \right) (\rho u)_K + (S_K - q_K) \frac{\partial (\rho u)_K}{\partial \mathbf{u}_J} + \left(\frac{\partial p^*}{\partial \mathbf{u}_J} - \frac{\partial p_K}{\partial \mathbf{u}_J} \right) n_x \right) , \end{aligned} \quad (375)$$

$$\begin{aligned} \frac{\partial (\rho E)_K^*}{\partial \mathbf{u}_J} &= \frac{\partial \Omega_K}{\partial \mathbf{u}_J} \left((S_K - q_K)(\rho E)_K - p_K q_K + p^* S_M \right) \\ &+ \Omega_K \left(\left(\frac{\partial S_K}{\partial \mathbf{u}_J} - \frac{\partial q_K}{\partial \mathbf{u}_J} \right) (\rho E)_K + (S_K - q_K) \frac{\partial (\rho E)_K}{\partial \mathbf{u}_J} - \frac{\partial p_K}{\partial \mathbf{u}_J} q_K - p_K \frac{\partial q_K}{\partial \mathbf{u}_J} + \frac{\partial p^*}{\partial \mathbf{u}_J} S_M + p^* \frac{\partial S_M}{\partial \mathbf{u}_J} \right) , \end{aligned} \quad (376)$$

$$\frac{\partial q_K}{\partial \mathbf{u}_K} = n_x \begin{bmatrix} -\frac{(\rho u)_K}{\rho_K^2} \\ \frac{1}{\rho_K} \\ 0 \end{bmatrix} . \quad (377)$$

5.1.2.4.3 Wave Speed Estimates This section describes how the wave speeds S_L and S_R can be estimated. Following Einfeldt et al. [105],

$$S_L = \min \left(\lambda_1(\mathbf{u}_L), \lambda_1(\hat{\mathbf{u}}) \right) , \quad (378)$$

$$S_R = \max \left(\lambda_3(\hat{\mathbf{u}}), \lambda_3(\mathbf{u}_R) \right) , \quad (379)$$

where $\lambda_1(\mathbf{u})$ and $\lambda_3(\mathbf{u})$ are the smallest and largest eigenvalues of the normal flux Jacobian matrix:

$$\lambda_1(\mathbf{u}) = u n_x - c \qquad \lambda_3(\mathbf{u}) = u n_x + c , \quad (380)$$

and $\hat{\mathbf{u}}$ is the Roe-averaged state [106], in which u and c are computed as follows:

$$\hat{\rho} = \sqrt{\rho_L \rho_R} , \quad (381)$$

$$\hat{u} = \frac{\sqrt{\rho_L} u_L + \sqrt{\rho_R} u_R}{\sqrt{\rho_L} + \sqrt{\rho_R}}\,, \tag{382}$$

$$\hat{H} = \frac{\sqrt{\rho_L} H_L + \sqrt{\rho_R} H_R}{\sqrt{\rho_L} + \sqrt{\rho_R}}\,, \tag{383}$$

$$\hat{h} = \hat{H} - \frac{1}{2}\hat{u}^2\,, \tag{384}$$

$$\hat{v} = \frac{1}{\hat{\rho}}\,, \tag{385}$$

$$\hat{c} = c(\hat{v}, \hat{h})\,, \tag{386}$$

where $c(v, h)$ represents a call to the equation of state interface.

5.2 Temporal Discretization

RELAP-7, through MOOSE, supports a number of standard implicit time integration methods such as the backward Euler (Section 5.2.1) and BDF2 (Section 5.2.2) methods.

5.2.1 Backward Euler

The backward Euler method [107] is a well-known, first-order, A-stable implicit time integration method. Given a generic semi-discrete equation in a form similar to equations (278) with (280),

$$\int_\Omega \left(\frac{\partial u^h}{\partial t} + G(u^h) \right) \phi_i \, \mathrm{d}\Omega = 0 \tag{387}$$

the backward Euler method results in the temporal discretization

$$\int_\Omega \left(\frac{u^{n+1} - u^n}{\Delta t} + G(u^{n+1}) \right) \phi_i \, \mathrm{d}\Omega = 0 \tag{388}$$

where Δt is the timestep, $t^{n+1} = t^n + \Delta t$, and $u^n \equiv u^h(t^n)$ is a shorthand notation used to refer to the finite element solution at time level n. Equation (388) is a fully-discrete (possibly nonlinear) equation which must be satisfied for each i.

Note that the backward Euler method, when applied to the linear convection equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 \tag{389}$$

yields a leading-order truncation error term of the form

$$\begin{aligned}\left.\frac{\partial u}{\partial t}\right|_{t^{n+1}} &= \frac{u^{n+1}-u^n}{\Delta t} + \frac{\Delta t}{2}\left.\frac{\partial^2 u}{\partial t^2}\right|_{t^{n+1}} + \mathcal{O}(\Delta t^2) \\ &= \frac{u^{n+1}-u^n}{\Delta t} + \frac{a^2\Delta t}{2}\left.\frac{\partial^2 u}{\partial x^2}\right|_{t^{n+1}} + \mathcal{O}(\Delta t^2)\end{aligned} \tag{390}$$

where (390) follows from differentiating the continuous equation (389) with respect to time:

$$\frac{\partial^2 u}{\partial t^2} = -a\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial x}\right) = -a\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial t}\right) = -a\frac{\partial}{\partial x}\left(-a\frac{\partial u}{\partial x}\right) = a^2\frac{\partial^2 u}{\partial x^2}\ . \tag{391}$$

Rearranging terms in (390) and adding $a\frac{\partial u}{\partial x}$ to both sides allows us to write

$$\frac{u^{n+1}-u^n}{\Delta t} + a\frac{\partial u}{\partial x} = \frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} - \frac{a^2\Delta t}{2}\frac{\partial^2 u}{\partial x^2} + \mathcal{O}(\Delta t^2) \tag{392}$$

where all the continuous derivatives are assumed to be evaluated at time level t^{n+1}. Thus, the semi-discrete form of the linear convection on the left-hand side of (392) is equal to the continuous parabolic partial differential equation on the right-hand side, which includes "artificial" diffusion of $\mathcal{O}(\frac{a^2\Delta t}{2})$, to within $\mathcal{O}(\Delta t^2)$. For this reason, we often say that the backward Euler time discretization is inherently stabilizing for the hyperbolic equation (389). Obviously, the artificial viscosity for the complete scheme is a composite of the artificial viscosity of both the time and spatial discretization.

The backward Euler time integration method should only be used for transients with RELAP-7 as an initial scoping calculation, or if only the steady-state solution is of interest. For accurate transient solutions with RELAP-7, the BDF2 time integration method, described next, is highly recommended because it is a second-order (in time) discretization.

5.2.2 BDF2

The backward differentiation formula (BDF) is a family of implicit methods for numerically integrating ordinary differential equations. Some notable members of this family

include BDF1, which is equivalent to the backward Euler [108] method discussed in Section 5.2.1, and BDF2, which is the highest-order BDF method which is still A-stable. For fixed step-size Δt, the BDF2 method applied to the ordinary differential equation

$$\frac{\partial u}{\partial t} = f(t, u) \tag{393}$$

$$u(t = 0) = u_0 \tag{394}$$

yields the update step:

$$u^{n+1} = \frac{4}{3}u^n - \frac{1}{3}u^{n-1} + \frac{2}{3}\Delta t f(u^{n+1}, t^{n+1}) \tag{395}$$

Dividing through by $\frac{2}{3}\Delta t$, equation (395) can be alternatively written as

$$\frac{\frac{3}{2}u^{n+1} - 2u^n + \frac{1}{2}u^{n-1}}{\Delta t} = f(u^{n+1}, t^{n+1}) \tag{396}$$

The left-hand side of (396) can be interpreted as a backward-difference approximation to the continuous time derivative $\frac{\partial u}{\partial t}$, and may be employed in a manner analogous to (388) to derive a fully-discrete system of equations:

$$\int_\Omega \left(\frac{\frac{3}{2}u^{n+1} - 2u^n + \frac{1}{2}u^{n-1}}{\Delta t} + G(u^{n+1}) \right) \phi_i \, \mathrm{d}\Omega = 0 \tag{397}$$

based on the semi-discrete equations (278) with (280).

The second-order, backward difference temporal integrator BDF2 can be generalized for time varying time-step sizes. By considering three consecutive solutions, u^{n-1}, u^n and u^{n+1}, at times t^{n-1}, t^n and t^{n+1}, respectively, the temporal derivative above can be expressed with BDF2 as:

$$\int_\Omega \partial_t u \, \phi_i = \int_\Omega \left(\omega_0 u^{n+1} + \omega_1 u^n + \omega_2 u^{n-1} \right) \phi_i \,, \tag{398}$$

with

$$\omega_0 = \frac{2\Delta t^{n+1} + \Delta t^n}{\Delta t^{n+1}\left(\Delta t^{n+1} + \Delta t^n\right)} \tag{399}$$

$$\omega_1 = -\frac{\Delta t^{n+1} + \Delta t^n}{\Delta t^{n+1}\Delta t^n} \tag{400}$$

$$\omega_2 = \frac{\Delta t^{n+1}}{\Delta t^n\left(\Delta t^{n+1} + \Delta t^n\right)} \tag{401}$$

where $\Delta t^n = t^n - t^{n-1}$ and $\Delta t^{n+1} = t^{n+1} - t^n$.

Notice that because BDF2 requires two old timesteps, the method must be employ a single step method, such as backward Euler, for the first time-step when starting. The BDF2 method is recommended for most transient simulations with RELAP-7.

5.3 Jacobian-Free Newton Krylov Solver

The RELAP-7 code solves coupled multi-physics problems using the Jacobian-Free Newton Krylov (JFNK) approach via the MOOSE framework. Field equations solved in the current RELAP-7 code include PDEs to describe one-dimensional fluid flow in pipe systems and heat conduction in solids, as well as ODEs to describe physics in zero-dimensional components and the point kinetics equations.

The JFNK method is a fully-coupled, multi-level method for solving large nonlinear equation systems. In general, it consists of at least two levels: the outer Newton loop for the nonlinear solve and the inner Krylov loop for the linear systems of equations associated to Newton iteration. The JFNK method has become an increasingly popular option for solving large nonlinear equation systems arising from multi-physics problems over the last 20 years, and has branched out into a number of different disciplines [7].

In what follows, a brief description of the JFNK method as it applies to the RELAP-7 application is given. The FEM-discretized field equations (Equations (278) with suitable time, and viscous stabilization, discretizations) are first written as

$$\mathbb{F}(\boldsymbol{u}) = \mathbf{0} \tag{402}$$

where $\mathbb{F}$ represents the nonlinear equation system and $\boldsymbol{u}$ is the solution vector. Newton's method requires an initial guess, $\boldsymbol{u}^0$, computed either from the initial conditions or the previous time-step solution, to start the iteration process. For the transient problems of interest here, the solution at a previous time step is generally used as the initial guess for the method. At the k^{th} iteration, the residual vector is defined as

$$\boldsymbol{r}^k \equiv \mathbb{F}(\boldsymbol{u}^k)\,. \tag{403}$$

Clearly if $\boldsymbol{u}^k$ satisfies (402) *exactly*, the k^{th} residual will be zero. To update the solution vector, the following equation is solved for the update vector, $\delta\boldsymbol{u}^{k+1}$:

$$\boldsymbol{J}(\boldsymbol{u}^k)\delta\boldsymbol{u}^{k+1} = -\boldsymbol{r}^k \tag{404}$$

where $\boldsymbol{J}(\boldsymbol{u}^k)$ is the Jacobian matrix evaluated at $\boldsymbol{u}^k$. In index notation,

$$J_{i,j} \equiv \frac{\partial \mathbb{F}_i}{\partial u_j} \ . \tag{405}$$

After $\delta \boldsymbol{u}^{k+1}$ is obtained, the $(k+1)^{st}$ solution iterate is computed by

$$\boldsymbol{u}^{k+1} = \boldsymbol{u}^k + \delta \boldsymbol{u}^{k+1} \ . \tag{406}$$

The Newton iteration is terminated when one of the following conditions is met:

1. The residual vector norm, $|\boldsymbol{r}^k|$, is sufficiently small.
2. The relative residual vector norm $\frac{|\boldsymbol{r}^k|}{|\boldsymbol{r}^0|}$ is sufficiently small.
3. The step size norm, $|\delta \boldsymbol{u}^{k+1}|$ is sufficiently small.

Note that (404) represents a large linear system of equations. In the JFNK method, we need not explicitly form the matrix $\boldsymbol{J}$: only its action on a vector (via matrix-vector product) is required. Effective preconditioning is generally required for Krylov subspace methods to be efficient, i.e., for the method to converge in a reasonable number of iterations. A preconditioned version of equation (404) can be expressed as (using right preconditioning as an example),

$$\boldsymbol{J}^k \boldsymbol{P}^{-1} \left(\boldsymbol{P} \delta \boldsymbol{u}^{k+1} \right) = -\boldsymbol{r}^k \tag{407}$$

where $\boldsymbol{P}$ is the preconditioning matrix. In the approach current used in RELAP-7, an analytical Jacobian matrix is computed according to (405), and passed to the underlying numerical solver library as the matrix $\boldsymbol{P}$ for preconditioning purposes.

6 Boundary Conditions

For convenience, or of necessity, governing balance equations are usually solved over a finite, bounded spatial domain. However, from a physical point of view the domain is usually not really bounded. There is some physical object or material beyond the fixed domain with which the material within the domain of interest has some interaction. To represent the physical effects, at least partially, of the entities beyond the domain of interest upon the the material within the domain, so called *boundary conditions* are introduced.

These boundary conditions must supply adequate information to fulfill both the mathematical needs of the governing balance equations solved within the domain and also to adequately represent the physical effects of the entities beyond the domain. It is emphasized that sometimes the approximations inherent in these boundary conditions adversely effects the solution of the governing balance equations. When this occurs, the domain should be modified, if possible, so that the boundary conditions can be placed where the effects of their approximate nature will be minimized.

Because the governing two-phase flow equations used in RELAP-7 are hyperbolic, the boundary conditions that will be specified must be consistent with the *method of characteristics* [1], [109], [110]. For a single-phase, one-dimensional flow with variable cross-sectional area $A(x)$, method of characteristics theory shows that solution information propagates along three characteristic, or wave, directions $\frac{dx}{dt}$ and that the solution at any point in time and space is constructed from the *characteristic information* carried by the three characteristic waves convergent at that point. One wave carries its charcteristic information, propagating at the material velocity, $\frac{dx}{dt} = u$. The other two characteristic waves travel, carrying their characteristic information, at acoustic or sound speed, w, relative to the flowing material, i.e. $\frac{dx}{dt} = u + w$ and $\frac{dx}{dt} = u - w$. The three characteristic equations which carry the solution information and their respective characteristic directions in $(x,t) - space$ are

$$dp + \rho w\, du = F_1\, dt \quad along \quad \frac{dx}{dt} = u + w \tag{408}$$

$$dp - \rho w\, du = F_2\, dt \quad along \quad \frac{dx}{dt} = u - w \tag{409}$$

$$d\rho - \frac{1}{w^2}\, dp = F_3\, dt \quad along \quad \frac{dx}{dt} = u\,, \tag{410}$$

where the source terms F_1, F_2, and F_3 are generally functions of the fluid's thermodynamic state, the flow's cross-sectional area $A(x)$ and its gradient $A'(x)$, flow velocity u, the fluid's specific heat at constant pressure c_p along with its coefficient of volume expansion $-\frac{1}{\rho}(\frac{\partial \rho}{\partial T})_p$, the external heat transfer to/from the fluid, and the pipe wall friction factor. The first two characteristics listed above are *acoustic* and, for subsonic flows, are sometimes referred to as *right running* and *left running characteristics*, respectively. The third characteristic listed above is referred to, for obvious reasons, as the *material motion*, *particle path*, or *entropic characteristic*. For the 7-equation two-phase model, seven of these characteristic equations and characteristic directions occur. Six of the seven characteristic equations are basically comprised of a set of three characteristics, similar to those above, for each of the two phases. The seventh characteristic equation and direction corresponds

to the volume fraction evolution equation. It specifies how the (liquid) volume fraction will change along the characteristic wave traveling with velocity $\frac{dx}{dt} = u_{int}$. Thus, for the 7-equation two-phase model the characteristics are

$$dp_{liq} + (\rho w)_{liq}\, du_{liq} = F_1\, dt \quad along \quad \frac{dx}{dt} = (u+w)_{liq} \tag{411}$$

$$dp_{liq} - (\rho w)_{liq}\, du_{liq} = F_2\, dt \quad along \quad \frac{dx}{dt} = (u-w)_{liq} \tag{412}$$

$$d\rho_{liq} - \frac{1}{w_{liq}^2}\, dp_{liq} = F_3\, dt \quad along \quad \frac{dx}{dt} = u_{liq} \tag{413}$$

$$dp_{vap} + (\rho w)_{vap}\, du_{vap} = F_4\, dt \quad along \quad \frac{dx}{dt} = (u+w)_{vap} \tag{414}$$

$$dp_{vap} - (\rho w)_{vap}\, du_{vap} = F_5\, dt \quad along \quad \frac{dx}{dt} = (u-w)_{vap} \tag{415}$$

$$d\rho_{vap} - \frac{1}{w_{vap}^2}\, dp_{vap} = F_6\, dt \quad along \quad \frac{dx}{dt} = u_{vap} \tag{416}$$

$$d\alpha_{liq} = F_7\, dt \quad along \quad \frac{dx}{dt} = u_{int}\,, \tag{417}$$

where now the functions F_j, $j = 1, \cdots, 7$, depend (in addition to those dependencies give above for single phase) also upon all of the interphase interaction terms.

At points on domain boundaries, it often occurs, e.g. for single phase flow, that less than three of the waves propagate their characteristic information to the boundary point. The information that does come from the characteristic waves propagating from the domain interior must be utilized to form the solution at the boundary. The additional information necessary to obtain the solution at the boundary must be supplied by appropriate boundary condition equations that approximate the physical and mathematical effect of the truncated material or fluid. Detailed discussion of the method of characteristics is beyond our scope here, but the parts of characteristic theory that will be employed will be apparent. RELAP-7 uses a continuous finite element method to numerically construct the solution for the governing two-phase flow equations. The elemental equations for the element adjacent to the boundary, in a sense, supplies all of the same information that the characteristic equations that propagate to the boundary from the domain interior provides. These finite element equations are also deficient, and must be supplimented with additional boundary condition information. There is, however, one caviat. The finite element equations are in an implicit form, i.e. they have a so called *mass matrix* which must be inverted, usually iteratively, to obtain the solution. So with some care, at each iterate the correct characteristic information will be extracted from the incomplete finite element equations for the

element adjacent to the boundary, and combined with the appropriate boundary condition equations to compute the complete boundary solution or flux approximation for the next iterate. Upon numerical convergence, the complete finite element solution at the boundary nodel will be consistent with both the characteristic information propagating to the boundary from the domain interior and the correctly specified boundary condition equations.

In the ensuing development, the following nomenclature is used: subscript *1 or nnd* on a variable denotes the elemental value of the variable at the terminous node (i.e. the first or last node) denoted with the subscript for the current solution iterate, subscript *bc* on a variable denotes a supplied boundary condition value, and superscript $*$ denotes a generic value of the variable at the boundary node. For each of the boundary conditions given, a detailed discription will first be given for single-phase flow, then in somewhat more terse, but concise, form for the 7-equation two-phase flow.

6.1 Closed End, Single-Phase

Perhaps the simplest boundary condition is that of a *closed end*, *dead-end*, or *wall* boundary condition. At a closed end the governing physics is that there can be no flow through the boundary, i.e. $u_{boundary} = 0$. To begin this description, assume that the closed end or wall boundary condition is located at the right, terminal end of a duct at its last node, numbered nnd. Subscript nnd on a variable will denote the finite element solution value of the variable at that boundary node of the duct for the current solution iterate. Superscript $*$ denotes a generic value of the variable at the boundary node. From the method of characteristics it is known that the *material motion, particle path, or entropic characteristic* from the domain interior requires either

$$\rho^* = \rho_{nnd}$$

or

$$p^* = p_{nnd}$$

while the right running *acoustic characteristic* from the domain interior requires either

$$u^* = u_{nnd}$$

or

$$p^* = p_{nnd}\,.$$

Futhermore, this is the *only* information that can be used from the solution's domain interior. But because the physical no-flow end condition $u_{nnd} = u_{bc} = 0$ is specified, this forces the following choice:

$$\rho^* = \rho_{nnd} \quad (from\, nodal\, solution\, iterate) \tag{418}$$
$$p^* = p_{nnd} \quad (from\, nodal\, solution\, iterate) \tag{419}$$
$$u^* = 0.0 \quad (specified\, boundary\, condition)\ . \tag{420}$$

To get p_{nnd}, the mass density and specific internal energy e_{nnd} for the current solution iterate must first be determined from the element nodal conservative variables

$$\rho_{nnd} = \frac{(\rho A)_{nnd}}{A_{nnd}} \tag{421}$$
$$e_{nnd} = \frac{(\rho E A)_{nnd}}{(\rho A)_{nnd}} - \frac{1}{2}\frac{(\rho u A)^2_{nnd}}{(\rho A)^2_{nnd}} \tag{422}$$

and then from the equation of state

$$p_{nnd} = p(v_{nnd}, e_{nnd}) \tag{423}$$

where the specific volume is the reciprical of mass density, $v_{nnd} = \frac{1}{\rho_{nnd}}$. For a *weakly specified* element boundary, boundary fluxes are set for node *nnd* as

$$(F_\rho)_{nnd} = \rho_{nnd} u_{bc} A_{nnd} = 0 \tag{424}$$
$$(F_{\rho u})_{nnd} = \rho_{nnd} u^2_{bc} A_{nnd} + p_{nnd} A_{nnd} = p_{nnd} A_{nnd} \tag{425}$$
$$(F_{\rho E})_{nnd} = \rho_{nnd} u_{bc} A_{nnd} [e_{nnd} + \frac{p_{nnd}}{\rho_{nnd}} + \frac{1}{2} u^2_{bc}] = 0\,. \tag{426}$$

Thus the nodal fluxes depend implicitly upon the current element nodal solution iterate (which provides characteristic information from the domain interior) and the supplied boundary condition information (which replaces the "missing" characteristic information). Solution with finite element methods are typically iterative. Even for explicit time integration, because of the necessity to invert the mass matrix, iteration is often used. Upon iterative convergence, the finite element boundary node variables will be consistent with

the characteristic information from the domain interior and the specified boundary condition $u_{nnd} = \frac{(\rho uA)_{nnd}}{(\rho A)_{nnd}} \approx 0$.

Remark: If so called *mass lumping* (in which the mass matrix is diagonalized) is used with explicit time itegration, iteration will not be performed, and the nodal information from the domain interior will need to be replaced with approximations to the actual characteristic equations. This same remark applies to all of the boundary conditions described in this chapter regarding the use of characteristic information.

Alternatively, instead of a weak boundary condition for the momentum equation, i.e. the specification of $(F_{\rho u})_{nnd}$, a *Dirichlet* or *strongly specified* boundary condition could be specified for the momentum equation. In this case, the momentum flux boundary condition (only) would be replaced by the condition

$$Residual_{(\rho uA)_{nnd}} = (\rho uA)_{nnd} - (\rho)_{nnd} u_{bc} A_{nnd}$$

which upon iterative convergence, i.e $Residual_{(\rho uA)_{nnd}} \approx 0$, gives $(\rho uA)_{nnd} \approx 0$ because of the specification $u_{bc} = 0$. Thus the two methods of setting a closed end boundary condition are equivalent to within iterative convergence tolerance. It is emphasized that, in this case, the weak boundary conditions (flux specification) are still used for the mass and total energy balance equations.

If the closed end boundary condition is specified for the other end of the pipe, i.e. node *1*, the the exact procedure above is mirrored and the fluxes for a weakly specified boundary are

$$(F_\rho)_1 = \rho_1 u_{bc} A_1 = 0 \tag{427}$$

$$(F_{\rho u})_1 = \rho_1 u_{bc}^2 A_1 + p_1 A_1 = p_1 A_1 \tag{428}$$

$$(F_{\rho E})_1 = \rho_1 u_{bc} A_1 [e_1 + \frac{p_1}{\rho_1} + \frac{1}{2} u_{bc}^2] = 0 \tag{429}$$

with

$$\rho_1 = \frac{(\rho A)_1}{A_1} \tag{430}$$

$$e_1 = \frac{(\rho EA)_1}{(\rho A)_1} - \frac{1}{2} \frac{(\rho uA)_1^2}{(\rho A)_1^2} \tag{431}$$

and again from the equation of state

$$p_1 = p(v_1, e_1)\,. \tag{432}$$

6.2 Closed End, Two-Phase

The closed end boundary condition for the nonequilibrium 7-equation two-phase model is very much similar to its single-phase counterpart described above. Because the 7-equation model has a special eigenstructure in which three of the seven eigenvalues are identical to those of liquid single phase and another three are identical those of the vapor phase, the method described above can be duplicated for each of the two phases. The characteristic equation due to the seventh eigenvalue, u_{int}, is satisfied from the element interior equation.

To make the notation lucid, another subscript is added to denote the phase or interface values. The physical no-flow boundary condition for the closed end with the 7-equation model is $u_{nnd,liq} = u_{nnd,vap} = u_{bc} = 0$. This forces the choice

$$\rho^*_{liq} = \rho_{nnd,liq} \quad (from\, nodal\, solution\, iterate) \tag{433}$$
$$\rho^*_{vap} = \rho_{nnd,vap} \quad (from\, nodal\, solution\, iterate) \tag{434}$$
$$p^*_{liq} = p_{nnd,liq} \quad (from\, nodal\, solution\, iterate) \tag{435}$$
$$p^*_{vap} = p_{nnd,vap} \quad (from\, nodal\, solution\, iterate) \tag{436}$$
$$u^*_{liq} = 0.0 \quad (specified\, boundary\, condition) \tag{437}$$
$$u^*_{vap} = 0.0 \quad (specified\, boundary\, condition)\ . \tag{438}$$

Thus, the phasic nodal variables $\rho_{nnd,k}$ and $e_{nnd,k}$, with $k = \{liq, vap\}$, for the current solution iterate are first determined from the current iterate of elemental nodal variables

$$\rho_{nnd,liq} = \frac{(\alpha\rho A)_{nnd,liq}}{(\alpha A)_{nnd,liq}} \tag{439}$$

$$\rho_{nnd,vap} = \frac{(\alpha\rho A)_{nnd,vap}}{(\alpha A)_{nnd,vap}} \tag{440}$$

$$e_{nnd,liq} = \frac{(\alpha\rho EA)_{nnd,liq}}{(\alpha\rho A)_{nnd,liq}} - \frac{1}{2}\frac{(\alpha\rho uA)^2_{nnd,liq}}{(\alpha\rho A)^2_{nnd,liq}} \tag{441}$$

$$e_{nnd,vap} = \frac{(\alpha\rho EA)_{nnd,vap}}{(\alpha\rho A)_{nnd,vap}} - \frac{1}{2}\frac{(\alpha\rho uA)^2_{nnd,vap}}{(\alpha\rho A)^2_{nnd,vap}} \tag{442}$$

which are then used with the equation of state to obtain the element nodal phasic pressures

$$p_{nnd,liq} = p_{liq}(v_{nnd,liq}, e_{nnd,liq}) \tag{443}$$

$$p_{nnd,vap} = p_{vap}(v_{nnd,vap}, e_{nnd,vap}) \tag{444}$$

with, again, the specific volumes being the reciprical of their respective mass densities, $v_{nnd,k} = \frac{1}{\rho_{nnd,k}}$, $k = \{liq, vap\}$. Finally, for a *weakly specified* element boundary, the boundary fluxes are set for node *nnd* as

$$(F_\rho)_{nnd,liq} = (\alpha\rho A)_{nnd,liq}\, u_{bc} = 0 \tag{445}$$

$$(F_\rho)_{nnd,vap} = (\alpha\rho A)_{nnd,vap}\, u_{bc} = 0 \tag{446}$$

$$(F_{\rho u})_{nnd,liq} = (\alpha\rho A)_{nnd,liq}\, u^2_{bc} + (\alpha A)_{nnd,liq}\, p_{nnd,liq} = (\alpha A)_{nnd,liq}\, p_{nnd,liq} \tag{447}$$

$$(F_{\rho u})_{nnd,vap} = (\alpha\rho A)_{nnd,vap}\, u^2_{bc} + (\alpha A)_{nnd,vap}\, p_{nnd,vap} = (\alpha A)_{nnd,vap}\, p_{nnd,vap} \tag{448}$$

$$(F_{\rho E})_{nnd,liq} = (\alpha\rho A)_{nnd,liq}\, u_{bc}\, [e_{nnd,liq} + \frac{p_{nnd,liq}}{\rho_{nnd,liq}} + \frac{1}{2}u^2_{bc}] = 0 \tag{449}$$

$$(F_{\rho E})_{nnd,vap} = (\alpha\rho A)_{nnd,vap}\, u_{bc}\, [e_{nnd,vap} + \frac{p_{nnd,vap}}{\rho_{nnd,vap}} + \frac{1}{2}u^2_{bc}] = 0\,. \tag{450}$$

Remark: There are only six fluxes specified here for the 7-equation model because the volume fraction evolution equation has no *Eulerian flux* (and thus cannot be integrated by parts for the finite element solution). The advection term in the volume fraction evolution equation results from the so called *Lagrangian flux* [13], and is not treated as a flux in the RELAP-7 finite element method. Because the volume fraction evolution equation is

already in characteristic form, with eigenvalue u_{int}, it is satisfied from the element interior solution, and nothing more needs to be done for the volume fraction at a closed end or wall boundary.

6.3 Stagnation Inlet

The *stagnation inlet*, or *tank* boundary condition is designed to approximate the effect of attaching a very large tank or volume of fluid (at rest) to the inlet so that flow can be driven into the duct. It could physically be a tank, or it could be a large volume such as the atmosphere. Regardless, it is ideally assumed that its volume is sufficiently large that flow velocities within the volume are negligible and, consequently, that the stagnation values of its thermodynamic properties are temporally invariant. A complicating feature of such a boundary condition is that, while this boundary condition may drive flow into the duct at steady state, during a transient, waves of sufficient magnitude may reflect from this boundary so as to cause a temporary flow reversal, i.e. the normal inlet becomes a temporary outlet until such time as the flow reverses again, becoming an inlet once more.

From the method of characteristics it is known that for a *subsonic* inlet there are two characteristics entering the domain; these characteristics will carry the user-supplied boundary values for the stagnation pressure $p_{0,b}$ and stagnation temperature $T_{0,b}$, where the subscript 0 denotes a stagnation quantity and the subscript b denotes boundary data. Note that in addition, the user also supplies the inlet void fraction, which is applied as a Dirichlet BC to the volume fraction; this is not discussed further in this section. The information taken to come from the interior solution is the velocity u.

To summarize the stagnation pressure and temperature inlet boundary condition, the following modifications are made to the nonlinear residuals, where the superscripts denote the associated equation, and the subscript i denotes the inlet node index:

- The mass and energy equations use strongly-imposed Dirichlet BC:

$$R_i^{\text{mass}} := (\alpha\rho A)_i - (\alpha A)_i \rho_b \,, \tag{451}$$

$$R_i^{\text{energy}} := (\alpha\rho E A)_i - (\alpha A)_i \rho_b E_b \,. \tag{452}$$

- The momentum equation uses a weakly-imposed BC by evaluating the boundary

fluxes with the appropriate boundary data:

$$R_i^{\text{momentum}} := R_i^{\text{momentum}} + \int\limits_{\partial\Omega^{\text{inlet}}} \alpha A(\rho_b u^2 + p_b)\varphi_i n_x d\Omega \,. \tag{453}$$

Here, the subscript b not only denotes user-supplied boundary data, as in $p_{0,b}$ and $T_{0,b}$, but also denotes quantities that depend on boundary data, i.e., the quantities are not computed directly from the interior solution. The following sequence summarizes how one computes the values ρ_b, E_b, and p_b referenced above:

1. First, enthalpy, specific internal energy, and density are directly computed from the boundary data and equation of state:

$$h_{0,b} \equiv h(p_{0,b}, T_{0,b}) \,, \tag{454}$$

$$e_{0,b} \equiv e(p_{0,b}, T_{0,b}) \,, \tag{455}$$

$$\rho_{0,b} \equiv \rho(p_{0,b}, T_{0,b}) \,. \tag{456}$$

2. Specific volume is computed using $\rho_{0,b}$:

$$v_{0,b} \equiv \frac{1}{\rho_{0,b}} \,. \tag{457}$$

3. Stagnation entropy is computed using the stagnation specific internal energy and specific volume:

$$s_{0,b} \equiv s(e_{0,b}, v_{0,b}) \,. \tag{458}$$

4. By definition of the stagnation state, the process is isentropic, so the entropy before and after stagnation is equal:

$$s_b \equiv s_{0,b} \,. \tag{459}$$

5. By definition of the stagnation state, kinetic energy should be the difference between static and stagnation enthalpy:

$$h_b \equiv h_{0,b} - \frac{1}{2}u^2 \,. \tag{460}$$

6. Static pressure is computed using the static enthalpy and entropy:

$$p_b \equiv p(h_b, s_b) \,. \tag{461}$$

7. Static density and specific internal energy are computed using the static pressure and entropy:

$$\rho_b \equiv \rho(p_b, s_b)\,, \tag{462}$$

$$e_b \equiv e(p_b, s_b)\,. \tag{463}$$

8. Finally, the static specific total energy is computed using the static specific internal energy and the interior-solution velocity:

$$E_b \equiv e_b + \frac{1}{2}u^2\,. \tag{464}$$

Note that currently, no check is performed to ensure that the flow does not reverse, and also, no check is performed to ensure that the flow is subsonic. These checks may be added in the future.

6.4 Static Pressure Outlet, Single-Phase

The *static pressure outlet* boundary condition is designed to approximate the effect of attaching a very large tank or volume of fluid, such as the atmosphere to the outlet of a duct so that the flow may be absorbed or captured by this large volume. The tank or volume is so large that its thermodynamic conditions do not change with this fluid addition. Especially it is assumed that the *static pressure* of the tank or volume static pressure is temporally invariant, and is therefore neccessarily specified, p_b or p_{back}. This specified static pressure is sometimes called the *back pressure*. A complicating feature of this boundary condition is that, while the duct may be outflowing at this boundary at steady state, during a transient, waves of sufficient magnitude may reflect from this boundary so as to cause a temporary flow reversal. Then the normal outlet becomes a temporary inlet until such time as the flow reverses again, becoming an outlet once more. This is very much like the *opposite* of that discussed for the stagnation inlet boundary above, and will be further elaborated upon subsequently.

To begin this description, assume first that the static pressure outlet boundary (and volume) is located at the terminus of a duct, at its last node *nnd*. For discussion sake, it is imagined to be also at the right end of the duct, so that the normal flow direction in the duct is from left to right. As with the closed end boundary condition, subscript *nnd* on a variable indicates the value at the last elemental (terminus) node of the duct for the current solution iterate. A superscript * denotes a generic value of the variable at this boundary

node. A subscript *b* denotes a supplied boundary condition value at the node that will be used in the computation of fluxes.

From the method of characteristics it is known that at a subsonic outlet there are two right running characteristics, a right running *acoustic characteristic* and a right running *material motion, particle path, or entropic characteristic*. The material motion, particle path, or entropic characteristic coming from the solution domain interior requires either

$$\rho^* = \rho_{nnd}$$

or

$$p^* = p_{nnd}$$

while the acoustic characteristic coming from the solution domain interior requires either

$$u^* = u_{nnd}$$

or

$$p^* = p_{nnd}\,.$$

Furthermore, this is the *only* information that can be used from the solution's interior domain. On physical basis, for a subsonic outflow the pressure at the outlet will be enforced to be the specified back pressure value p_b. This forces the following choice:

$$\rho^* = \rho_{nnd} \;\; (from\, nodal\, solution\, iterate) \tag{465}$$
$$u^* = u_{nnd} \;\; (from\, nodal\, solution\, iterate) \tag{466}$$
$$p^* = p_b \;\; (specified\, boundary\, condition)\,, \tag{467}$$

where ρ_{nnd} and u_{nnd} are obtained from the current solution iterate

$$\rho_{nnd} = \frac{(\rho A)_{nnd}}{A_{nnd}} \tag{468}$$
$$u_{nnd} = \frac{(\rho u A)_{nnd}}{(\rho A)_{nnd}}\,. \tag{469}$$

Then the internal energy is determined from the equation of state

$$
\begin{aligned}
e^* &= e(v^*, p^*) \\
&= e(\frac{1}{\rho^*}, p_b)\,,
\end{aligned} \tag{470}
$$

and the sound speed is determined by either

$$
\begin{aligned}
w^* &= w(v^*, p^*) \\
&= w(\frac{1}{\rho^*}, p_b)\,,
\end{aligned} \tag{471}
$$

or (equivalently)

$$
\begin{aligned}
w^* &= w(v^*, e^*) \\
&= w(\frac{1}{\rho^*}, e^*)\,.
\end{aligned} \tag{472}
$$

At this point, the velocity is verified to be subsonic, i.e. if $0 < u_{nnd} \leq w^*$ then the fluxes are set for node *nnd* as

$$
(F_\rho)_{nnd} = (\rho u A)_{nnd} \tag{473}
$$

$$
(F_{\rho u})_{nnd} = \frac{(\rho u A)^2_{nnd}}{(\rho A)_{nnd}} + p_b A_{nnd} \tag{474}
$$

$$
(F_{\rho E})_{nnd} = (\rho u A)_{nnd} \left[e^* + \frac{p_b}{\rho_{nnd}} + \frac{1}{2} \frac{(\rho u A)^2_{nnd}}{(\rho A)^2_{nnd}} \right]\,. \tag{475}
$$

Thus the nodal fluxes depend implicitly upon the current element nodal solution iterate (which provides characteristic information from the domain interior) and the supplied boundary condition information (which replaces the "missing" characteristic information). The exit Mach number can also be computed form $M_{exit} = \frac{u_{nnd}}{w^*}$.

If, however, the outflow is supersonic, i.e. if $u_{nnd} > w^*$ then the characteristic information that is used must be modified. In addition to the two right running characteristics considered above for subsonic outflow, there is another right running characteristics coming from the solution domain interior. Because the outflow is supersonic, pressure distrubances cannot propagate upstream from the large volume or tank into the duct, and the

pressure at the exit no longer is required to match the specified pressure p_b. This requires that

$$\rho^* = \rho_{nnd}$$

or

$$p^* = p_{nnd}$$

from the material motion, particle path, or entropy characteristic; while the two acoustic characteristics coming from the solution domain interior require **both**

$$u^* = u_{nnd}$$

and

$$p^* = p_{nnd}\,.$$

Thus $\rho^* = \rho_{nnd}$, $u^* = u_{nnd}$, and $p^* = p_{nnd}$. For supersonic outflow, all of the fluxes are computed from the interior finite element nodal solution information. That is, from the current solution iterate

$$\rho_{nnd} = \frac{(\rho A)_{nnd}}{A_{nnd}} \tag{476}$$

$$u_{nnd} = \frac{(\rho u A)_{nnd}}{(\rho A)_{nnd}} \tag{477}$$

$$e^{**} = e_{nnd} = \frac{(\rho E A)_{nnd}}{(\rho A)_{nnd}} - \frac{1}{2}\frac{(\rho u A)^2_{nnd}}{(\rho A)^2_{nnd}} \tag{478}$$

and from the equation of state

$$p^* = p_{nnd} = p(\frac{1}{\rho^*}, e^{**}) \tag{479}$$

$$w^{**} = w(\frac{1}{\rho^*}, e^{**})\,, \tag{480}$$

the second of which can be used to determine the exit Mach number $M_{exit} = \frac{u_{nnd}}{w^{**}}$. The

supersonic outflow fluxes at node *nnd* are then set as

$$(F_\rho)_{nnd} = (\rho u A)_{nnd} \tag{481}$$

$$(F_{\rho u})_{nnd} = \frac{(\rho u A)^2_{nnd}}{(\rho A)_{nnd}} + p(\frac{1}{\rho^*}, e^{**})A_{nnd} \tag{482}$$

$$(F_{\rho E})_{nnd} = (\rho u A)_{nnd} \left[e^{**} + \frac{p(\frac{1}{\rho^*}, e^{**})}{\rho^*} + \frac{1}{2}\frac{(\rho u A)^2_{nnd}}{(\rho A)^2_{nnd}} \right] . \tag{483}$$

Remark: For sonic or supersonic outflows, if $p_{nnd} > p_b$ at solution convergence, the exit flow is said to be *underexpanded*. If, on the other hand, $p_{nnd} < p_b$ at solution convergence, the exit flow is said to be *overexpanded*. If, at solution convergence, the supersonic flow has $p_{nnd} = p_b$ the flow is said to be *perfectly expanded*. Nuclear vessel, and other high-pressure tank blowdowns are commonly underexpanded.

Last, but not least, the flow at the "exit" node must be checked to see if it has reversed, i.e. the "outlet" has temporarily become an inlet. If $u_{nnd} \cdot n_{nnd} < 0$ **and** $|u_{nnd}| < w^*$ the flow has become a subsonic inlet (here n_{nnd} is the solution domain outward normal at node nnd). If $u_{nnd} \cdot n_{nnd} < 0$ **and** $|u_{nnd}| > w^{**}$ the flow has become a supersonic inlet. If either of these conditions occurs during an iterate, the boundary condition should be changed to stagnation inlet boundary condition with the specified back pressure p_b becoming the specified stagnation inlet pressure p_0, i.e. $p_0 = p_b$. Obviously, a value for stagnation temperature, T_0, must also be specified, to be used only in this event. The stagnation inlet boundary conditions were fully described in the previous two sections.

6.5 Static Pressure Outlet, Two-Phase

Again the eigenstructure for the 7-equation two-phase model, without source terms, is like that of two, separate instances of the single-phase model. The source terms which would occur with the characteristic equations are accounted for in the solution of the element equations. To physically and mathematically describe the static pressure outlet for the 7-equation two-phase model, the procedure described in the previous section for a single phase is duplicated for each of the two phases, except that the effective flow areas are modulated by the local, phasic volume fractions. That accounts for six of the seven characteristic equations. The effect of the additional characteristic which occurs due to the volume fraction evolution equation for the 7-equation must also be taken into account.

As was done with the extension of the closed end and stagnation boundary conditions from single-phase to the 7-equation two-phase equation set an additional subscript indicating the phase or interface is utilized. As with the single-phase version, the two-phase static pressure outlet boundary for RELAP-7 would generally require the specification of a static outlet pressure, or back pressure, for each phase, $p_{b,k}$ for $k = \{liq, vap\}$. However, it is difficult to imagine a case where discharge of two-phase flow from a duct would occur due to each phase responding to a separate receiver, i.e. $p_{b,liq} \neq p_{b,vap}$. So for subsonic outflow of each phase, a common static outlet or back pressure p_b is employed for both phases.

From the method of characteristics, for subsonic outflow of each phase

$$\rho^*_{liq} = \rho_{nnd,liq} \;\; (from\, nodal\, solution\, iterate) \tag{484}$$
$$u^*_{liq} = u_{nnd,liq} \;\; (from\, nodal\, solution\, iterate) \tag{485}$$
$$p^*_{liq} = p_b \;\; (specified\, boundary\, condition)\,, \tag{486}$$

and

$$\rho^*_{vap} = \rho_{nnd,vap} \;\; (from\, nodal\, solution\, iterate) \tag{487}$$
$$u^*_{vap} = u_{nnd,vap} \;\; (from\, nodal\, solution\, iterate) \tag{488}$$
$$p^*_{vap} = p_b \;\; (specified\, boundary\, condition)\,, \tag{489}$$

where $\rho_{nnd,k}$ and $u_{nnd,k}$ for $k = \{liq, vap\}$ are obtained from the current solution iterate

$$\rho_{nnd,k} = \frac{(\alpha\rho A)_{nnd,k}}{(\alpha A)_{nnd,k}} \tag{490}$$
$$u_{nnd,k} = \frac{(\alpha\rho u A)_{nnd,k}}{(\alpha\rho A)_{nnd,k}}\,. \tag{491}$$

Then the phasic internal energies are determined from the equation of state

$$\begin{aligned} e^*_k &= e_k(v^*_k, p^*_k) \\ &= e_k(\frac{1}{\rho^*_k}, p_b)\,, \end{aligned} \tag{492}$$

and the phasic sound speeds are determined by either

$$\begin{aligned} w_k^* &= w_k(v_k^*, p^*) \\ &= w_k(\frac{1}{\rho_k^*}, p_b)\,, \end{aligned} \tag{493}$$

or (equivalently)

$$\begin{aligned} w_k^* &= w_k(v_k^*, e_k^*) \\ &= w_k(\frac{1}{\rho_k^*}, e_k^*)\,. \end{aligned} \tag{494}$$

From the seventh characteristic due to the volume fraction evolution equation, phase k volume fractions must come from the phase's solution domain interior if $u_{int} \cdot n_{nnd} > 0$, i.e. they must also come from the finite element current solution iterate

$$\alpha_{nnd,k} = \frac{(\alpha A)_{nnd,k}}{A_{nnd}}\,. \tag{495}$$

At this point, the phasic velocities are each verified to be subsonic, i.e. if $0 < |u_{nnd,k}| \leq w_k^*$ for each phase $k = \{liq, vap\}$ then the fluxes are set for each subsonic phase, for node *nnd* as

$$(F_\rho)_{nnd,k} = (\alpha\rho u A)_{nnd,k} \tag{496}$$

$$(F_{\rho u})_{nnd,k} = \frac{(\alpha\rho u A)_{nnd,k}^2}{(\alpha\rho A)_{nnd,k}} + p_b(\alpha A)_{nnd,k} \tag{497}$$

$$(F_{\rho E})_{nnd,k} = (\alpha\rho u A)_{nnd,k}\left[e_k^* + \frac{p_b}{\rho_{nnd,k}} + \frac{1}{2}\frac{(\alpha\rho u A)_{nnd,k}^2}{(\alpha\rho A)_{nnd,k}^2}\right]\,. \tag{498}$$

The exit Mach number for each phase is then $M_{exit,k} = \frac{u_{nnd,k}}{w_k^*}$, for $k = \{liq, vap\}$.

If, however, the outflow for either phase is supersonic, i.e. $u_{nnd,k} > w_k^*$, for $k = \{liq, vap\}$, then the characteristic information that is used for that phase must be modified. In addition to the two right running characteristics considered above for subsonic outflow, there is another right running characteristic for that phase coming from the solution

domain interior. Because the phasic outflow is supersonic, pressure distrubances cannot propagate upstream in that phase from the large volume or tank into the duct, and the pressure for that phase at the exit no longer is required to match the specified pressure p_b. This requires that, for each supersonic phase k, $\rho_k^* = \rho_{nnd,k}$, $u_k^* = u_{nnd,k}$, and $p_k^* = p_{nnd,k}$. For each phase with supersonic outflow, all of the fluxes are computed from the interior finite element nodal solution information. That is, for each supersonic phase at node *nnd*, from the current solution iterate

$$\rho_{nnd,k} = \frac{(\alpha\rho A)_{nnd,k}}{(\alpha A)_{nnd,k}} \tag{499}$$

$$u_{nnd,k} = \frac{(\alpha\rho u A)_{nnd,k}}{(\alpha\rho A)_{nnd,k}} \tag{500}$$

$$e_k^{**} = e_{nnd,k} = \frac{(\alpha\rho E A)_{nnd,k}}{(\alpha\rho A)_{nnd,k}} - \frac{1}{2}\frac{(\alpha\rho u A)_{nnd,k}^2}{(\alpha\rho A)_{nnd,k}^2} \tag{501}$$

and from the equation of state

$$p_k^* = p_{nnd,k} = p_k(\frac{1}{\rho_k^*}, e_k^{**}) \tag{502}$$

$$w_k^{**} = w_k(\frac{1}{\rho_k^*}, e_k^{**})\,, \tag{503}$$

the second of which can be used to determine the phase's exit Mach number $M_{exit,k} = \frac{u_{nnd,k}}{w_k^{**}}$. The supersonic phase's outflow fluxes at node *nnd* are then set as

$$(F_\rho)_{nnd,k} = (\alpha\rho u A)_{nnd,k} \tag{504}$$

$$(F_{\rho u})_{nnd,k} = \frac{(\alpha\rho u A)_{nnd,k}^2}{(\alpha\rho A)_{nnd,k}} + p_k(\frac{1}{\rho_k^*}, e_k^{**})(\alpha A)_{nnd,k} \tag{505}$$

$$(F_{\rho E})_{nnd,k} = (\alpha\rho u A)_{nnd,k}\left[e_k^{**} + \frac{p_k(\frac{1}{\rho_k^*}, e_k^{**})}{\rho_k^*} + \frac{1}{2}\frac{(\alpha\rho u A)_{nnd,k}^2}{(\alpha\rho A)_{nnd,k}^2}\right]. \tag{506}$$

Remark: In the discharge of a two-phase mixture from a duct into a receiver (large tank or atmosphere, etc.), each phase can be subsonic, underexpanded, or overexpanded. That,

coupled with the complicated dynamics of the 7-equation two-phase model, with its multiple inherent relaxation processes, in the element adjacent to the boundary, can lead to very complicated discharge flows with complicated choking behavior.

Lastly, as was done for single-phase, the flow for each of the two phases at the "exit" node must be checked to see if it has reversed, i.e. the "outlet" has temporarily become an inlet for that phase. If $u_{nnd,k} \cdot n_{nnd} < 0$ **and** $|u_{nnd,k}| < w_k^*$ the flow has become a subsonic inlet for phase k. If $u_{nnd,k} \cdot n_{nnd} < 0$ **and** $|u_{nnd,k}| > w_k^{**}$ the flow has become a supersonic inlet for phase k. If either of these conditions occurs for phase k during an iterate, the boundary condition for that phase should be changed to a phasic stagnation inlet boundary condition with the specified back pressure p_b becoming the specified phasic stagnation inlet pressure $p_{0,k}$, i.e. $p_{0,k} = p_b$. Obviously, a value for the phasic stagnation temperature, $T_{0,k}$, and a value for the liquid phase volume fraction, $\alpha_{bc,liq}$, ($\alpha_{bc,vap} = 1 - \alpha_{bc,liq}$) must also be specified, to be used *only* in this event. The stagnation inlet boundary conditions were fully described previously.

6.6 Specified Charging Rate, Single-Phase

The *specified charging rate* boundary condition, sometimes also refered to as an *injection boundary*, joins the stagnation inlet boundary condition, discussed previously, as another way to approximate an inlet flow boundary. With the specified charging rate boundary condition the *inlet mass flow rate* is specified along with the *stagnation enthalpy* of the inlet flow. This is a fairly strong physical condition in that it enforces the specified inlet mass flow rate no matter what is happening inside the duct to which it is attached. It is emphasized that, unlike the stagnation inlet boundary, no flow reversal can occur with the specified charging rate boundary condition. In the following description it will be assumed that these specified parameters are such as to produce a *subsonic* inlet flow condition.

It is assumed first that the specified charging rate boundary is located at the terminus of a duct at its first node, *node 1*, which for discrussion sake will be imagined to be also at the left end of the duct, so that the flow will be driven into the duct from left to right. Subscripts on a variable will again indicate the variable at that finite element node number, except that the subscript 0 will also unambiguously be use to indicate a stagnation value. Superscript $*$ denotes a generic value of the variable at the boundary node.

From the method of characteristics it is known that for a subsonic inlet there is only

one left running *acoustic characteristic* coming from the solution domain interior which requires either

$$u^* = u_1$$

or

$$p^* = p_1 \,, \tag{507}$$

and this is the *only* information that can be used from the solution's interior domain. For simplicity, the former, $u^* = u_1$, is picked. The other information which would be carried by a right running *acoustic characteristic*, u^* or p^*, and right running *material motion, particle path, or entropy characteristic*, ρ^* or p^*, is missing and must be supplied by the boundary conditions. The choice of u^* to come from the interior solution forces the following choice:

$$\begin{aligned} u^* = u_1 \quad & (from\, nodal\, solution\, iterate) \\ p^* \quad & (from\, specified\, boundary\, condition/solution) \\ \rho^* \quad & (from\, specified\, boundary\, condition/solution) \;. \end{aligned}$$

The missing information will be supplied from two physical approximations based on: (1) From the energy equation, the stagnation enthalpy, h_0, is invariant at the inlet and is specified, and (2) From the specified mass flow rate, $\dot{m}$, at the duct inlet. This information is embodied in the following two equations which must be satisfied simultaneously:

$$\rho^* u_1 A_1 = \dot{m}$$

and the *transcendental equation*

$$h(\rho^*, p^*) + \frac{1}{2} u_1^2 = h_0 \,,$$

where the nodal velocity, u_1 is determined from

$$u_1 = \frac{(\rho u A)_1}{(\rho A)_1} \,. \tag{508}$$

These two equations can be simplified. From the first, an explicit expression of ρ^* is obtained

$$\rho^* = \frac{\dot{m}}{A_1} \frac{(\rho A)_1}{(\rho u A)_1} \,, \tag{509}$$

and from the second, a transcendental equation in p^* only (with ρ^* as a dependency) is obtained

$$e(\rho^*, p^*) + \frac{p^*}{\rho^*} + \frac{1}{2}\frac{(\rho u A)_1^2}{(\rho A)_1^2} = h_0\,, \tag{510}$$

which, of course, must be solved iteratively for p^*. One way that this can be accomplished for an iterative finite element solver (to avoid an iteration within an iteration) is to rearrange this equation into the *functional iteration* form

$$p^{*,\,\nu+1} = \rho^* h_0 - \rho^* e(\rho^*, p^{*,\,\nu}) - \frac{1}{2}\rho^* \frac{(\rho u A)_1^2}{(\rho A)_1^2} \tag{511}$$

where superscripts $\nu+1$ and ν denote iterate levels. Thus the p^* dependency in $e(\rho^*, p^{*,\,\nu})$ *lags* by one iteration level. However, at convergence, all the variables will be consistent to with convergence tolerance.

With ρ^*, u^*, and $p^{*,\,\nu+1}$ determined for the current solution iterate, the fluxes can be computed as

$$(F_\rho)_1 = \dot{m} \tag{512}$$

$$(F_{\rho u})_1 = \frac{\dot{m}^2}{\rho^* A_1} + p^{*,\,\nu+1} A_1 \tag{513}$$

$$(F_{\rho E})_1 = \dot{m} h_0\,. \tag{514}$$

As with the previously described boundaries, the nodal fluxes again depend implicitly upon the current element nodal solution iterate (which provides characteristic information from the domain interior) and the supplied boundary condition information (which replaces the "missing" characteristic information).

Remark: It is tempting to replace the weakly imposed boundary condition for momentum, where flux $(F_{\rho u})_1$ is specified above, by the strongly imposed Dirichlet boundary condition $Residual = (\rho u A)_1 - \dot{m} \approx 0$. This approximation, however, violates the characteristic conditions for a subsonic inlet because, in that case, *all* of the boundary condition information would be coming from the specified information, and none of the information would be coming from the solution in the domain interior.

6.7 Specified Charging Rate, Two-Phase

The specified charging rate boundary condition is designed to approximate an inlet in which the mass flow rate is specified for each phase, $\dot{m}_k$, and the flow for each phase enters the duct with a specified stagnation enthalpy, $h_{0,k}$, $k = \{liq, vap\}$. As with the single-phase version of this boundary condition, it is assumed that the conditions specified are such that the inlet flow velocity for each phase is subsonic. No flow reversal can occur for either phase. Subscripts and superscripts are identical to those for single-phase, except that an additional subscript is added to variables to indicate that the variable applies to a particular phase k, with $k = \{liq, vap\}$.

As with the other two-phase boundary conditions, because of the eigenstructure, the two-phase models for each phase mimick those of the corresponding single phase. To account for the volume fraction evolution equation, a seventh characteristic condition requires that a volume fraction of liquid be specified at the boundary, $\alpha_{bc,liq}$, if u_{int} is directed into the duct and that the liquid volume fraction at the boundary comes from the interior of the duct if u_{int} is directed out of the duct.

From the method of characteristics it is known that for a subsonic inlet there is only one left running *acoustic characteristic* for each phase coming from the solution domain interior which requires either

$$u_k^* = u_{1,k}$$

or

$$p_k^* = p_{1,k}\,, \tag{515}$$

for each $k = \{liq, vap\}$, and this is the *only* information that can be used from the solution's interior domain. For simplicity, the former, $u_k^* = u_{1,k}$, is picked for each phase k. The other information which would be carried by right running *acoustic characteristics*, u_k^* or p_k^* for each $k = \{liq, vap\}$, and right running *material motion, particle path, or entropy characteristics*, ρ_k^* or p_k^* for each $k = \{liq, vap\}$, is missing and must be supplied by the boundary conditions. The choice of u_k^* for each phase to come from the interior solution forces the following choices for each $k = \{liq, vap\}$:

$$\begin{aligned} u_k^* = u_{1,k} \quad & (from\, nodal\, solution\, iterate) \\ p_k^* \quad & (from\, specified\, boundary\, condition/solution) \\ \rho_k^* \quad & (from\, specified\, boundary\, condition/solution)\ . \end{aligned}$$

In addition, the linear degenerate characteristic resulting from the volume fraction evolution equation forces the following choice

$$\alpha^*_{liq} = \alpha_{bc,liq} \quad if\ u_{int} \cdot n_1 \leq 0 \quad (u_{int}\ into\ duct) \tag{516}$$

$$\alpha^*_{liq} = \alpha_{1,liq} \quad if\ u_{int} \cdot n_1 > 0 \quad (u_{int}\ out\ from\ duct)\,, \tag{517}$$

and, obviously, $\alpha^*_{vap} = 1 - \alpha^*_{liq}$.

As with the single-phase case, the missing information will be supplied from the two physical approximations for each phase (which must be satisfied simultaneously for each phase)

$$\alpha^*_k \rho^*_k u_{1,k} A_1 = \dot{m}_k$$

and the *transcendental equations*

$$h_k(\rho^*_k, p^*_k) + \frac{1}{2} u^2_{1,k} = h_{0,k}\,,$$

where the nodal velocities, $u_{1,k}$ are determined from

$$u_{1,k} = \frac{(\alpha \rho u A)_{1,k}}{(\alpha \rho A)_{1,k}}\,, \tag{518}$$

for each $k = \{liq, vap\}$. The interface velocity u_{int} is determined from the phasic velocities and thermodynamic properties

$$u_{int} = \frac{Z_{liq} u_{1,liq} \cdot n_1 + Z_{vap} u_{1,vap} \cdot n_1}{Z_{liq} + Z_{vap}} - sgn(\alpha_{1,liq} - \alpha_{bc,liq}) \frac{p^*_{liq} - p^*_{vap}}{Z_{liq} + Z_{vap}} \tag{519}$$

where $Z_k = \rho^*_k w^*_k$ for $k = \{liq, vap\}$ is the phasic acoustic impedance. If $u_{int} \leq 0$ the interface velocity is flowing into the duct and the specified boundary value of $\alpha_{bc,liq}$ is used, i.e. $\alpha^*_{liq} = \alpha_{bc,liq}$, which is handled as a *Dirichlet* boundary. If, on the other hand, when $u_{int} > 0$ the interface velocity is interface flow out from the duct and the specified boundary value of $\alpha_{bc,liq}$ must not be used. In this condition, no boundary condition is necessary and $\alpha^*_{liq} = \alpha_{1,liq}$ from the current finite element solution iterate for node *1*.

These equations can be simplified. From the first set, explicit expressions of ρ^*_k are obtained

$$\rho^*_k = \frac{\dot{m}_k}{\alpha^*_k A_1} \frac{(\alpha \rho A)_{1,k}}{(\alpha \rho u A)_{1,k}}\,, \tag{520}$$

and from the second set, transcendental equations in p_k^* only for each phase k (with ρ_k^* as a dependency, respectively in each phase) are obtained

$$e_k(\rho_k^*, p_k^*) + \frac{p_k^*}{\rho_k^*} + \frac{1}{2}\frac{(\alpha\rho u A)_{1,k}^2}{(\alpha\rho A)_{1,k}^2} = h_{0,k}\,. \tag{521}$$

As with the single-phase version, for an iterative finite element solver, a way to handle the additional iterative requirements of the transcendental equations is to rearrange them into *functional iteration* form

$$p_k^{*,\nu+1} = \rho_k^* h_{0,k} - \rho_k^* e_k(\rho_k^*, p_k^{*,\nu}) - \frac{1}{2}\rho_k^* \frac{(\rho u A)_{1,k}^2}{(\rho A)_{1,k}^2} \tag{522}$$

for each $k = \{liq, vap\}$, where superscripts $\nu + 1$ and ν denote iterate levels.

With ρ_k^*, u_k^*, and $p_k^{*,\nu+1}$ for $k = \{liq, vap\}$ determined, the fluxes for each phase can be computed as

$$(F_\rho)_{1,k} = \dot{m}_k \tag{523}$$

$$(F_{\rho u})_{1,k} = \frac{\dot{m}_k^2}{\alpha_k^*\rho_k^* A_1} + \alpha_k^* p_k^{*,\nu+1} A_1 \tag{524}$$

$$(F_{\rho E})_{1,k} = \dot{m}_k h_{0,k}\,. \tag{525}$$

7 Flow Topology - Dependent Closure Models

The purpose of this chapter is to describe the correlations that close the equation system for both the single- and two-phase flows. Because information was lost in the averaging process used to homogenize the 1-D balance equations, some of this information must be approximately reconstructed to restore closure of the governing equation systems. For single-phase flow, closure correlations are required to describe the interaction between the fluid and the wall, e.g., wall drag and wall heat transfer models. In addition to those types of closures necessary for single-phase flows, two-phase flows require closures related to the fact that interfaces, having dynamically changing topologies, are present between the

two phases. For two-phase flow, wall drag and wall heat transfer models, which now must also include the phasic partitioning, are required to describe the interaction between the two phases and the wall. Additional closure correlations to describe the interfacial interactions between the two phases are also necessary to close the system, such as interfacial area concentration and distribution, interfacial drag (friction) and interfacial heat/mass transfer models. For two-phase flow, most of the closure correlations depend critically on the local topology of the two-phase flow, for example bubbly flow in vertical pipes.

When a vapor-liquid mixture flows in ducts, pipes, or channels, the two phases may distribute in a variety of patterns or topologies, often referred to as *flow regimes*. In general, these patterns depend on the phasic flow rates, fluid properties, and channel geometries. In two-phase flows, interfacial heat/mass and momentum exchanges vary greatly with these topologies, so it is important to correctly identify the local flow regime in the simulations of two-phase flow. For example, Fig. 5 shows the flow regimes in determining the interfacial drag models used in the TRACE code [2]. In this particular case, the interfacial drag models are considered for the dispersed bubble flow regime, the combined slug and Taylor cap flow regime, and the annular/mist flow regime.

Many of these flow topology-dependent closure correlations have been extensively investigated and partially validated in existing system analysis codes, for example TRACE [2]. For the current stage of RELAP-7 code development, the existing closure correlations for vertical pre-CHF flows (i.e. for vertical flows before the critical heat flux state is reached) from the TRACE code [2] are used exclusively. Thus for all references in this chapter, the reader is referred to the TRACE code manual [2]. More details of the flow topology-dependent closure models are discussed in the following subsections.

7.1 Wall Drag

In this section, the wall friction model will be discussed for both single-phase and two-phase flow. The single-phase flow will be discussed first. TRACE closures for wall friction are implemented in terms of wall "drag coefficients"; for single-phase flow, the wall drag coefficient is defined as

$$C_{wall,k} = \frac{2 f_{wall,k} \rho_k}{d_h} \,, \tag{526}$$

where the subscript $k = \{liq, vap\}$ indicates the fluid phase, $f_{wall,k}$ is the Fanning friction factor[5] for single-phase flow of phase k, and d_h is the hydraulic diameter. For 2-phase

[5]Note that the Fanning friction factor is equal to 1/4 times the Darcy-Weisbach friction factor.

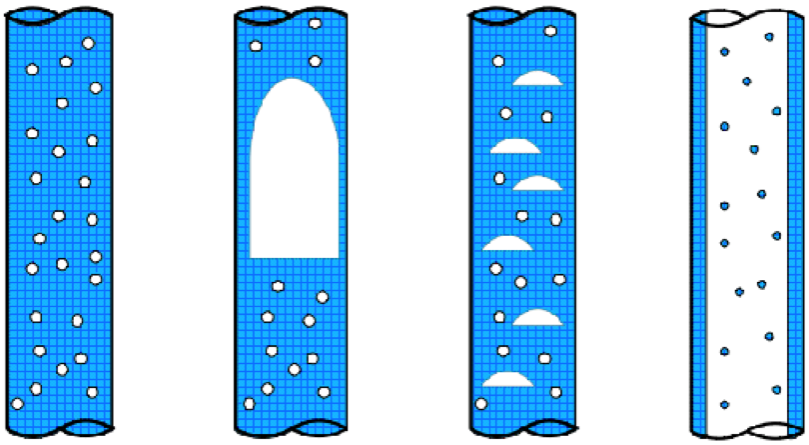

Figure 5. Flow regimes in vertical pipes under the pre-CHF conditions. From left to right, dispersed bubble, slug flow, taylor cap bubble, and annular/mist [2].

flow, one must multiply by the fraction χ_k of the surface area that phase k occupies, as discussed in Section 3.2.4, and additionally one can apply a 2-phase multiplier ϕ_k^{friction}:

$$C_{wall,k} = \phi_k^{\text{friction}} \frac{2 f_{wall,k} \rho_k}{d_h} \chi_k \,. \tag{527}$$

Note that usually, the terminology "drag coefficient" represents a dimensionless quantity C_D relating drag force F_D and dynamic pressure $\frac{1}{2}\rho u^2$:

$$F_D = C_D \left(\frac{1}{2} \rho u^2 \right) A_S \,, \tag{528}$$

where A_S is the area of the surface giving friction. However, in the TRACE context, "drag coefficient" will represent the quantity $C_{wall,k}$ defined above. The two-phase flow wall friction model will be further discussed for both the pre-CHF and post-CHF flow, and also for a special horizontal stratified flow condition.

7.1.1 Single-Phase Flow

The single-phase wall friction is calculated as

$$F_{\text{wall}}^{\text{friction}} = C_{\text{wall}}|u|uA\,, \tag{529}$$

where u is the velocity, and C_{wall} is the wall drag coefficient. As suggested in TRACE [2], the Churchill formula is used to calculate the wall friction factor for single-phase flow:

$$f_{\text{wall}} = 2\left[\left(\frac{8}{\text{Re}}\right)^{12} + \frac{1}{(a+b)^{3/2}}\right]^{1/12}\,, \tag{530}$$

where

$$a = \left\{2.475\ln\left[\frac{1}{\left(\frac{7}{\text{Re}}\right)^{0.9} + 0.27\left(\frac{\epsilon}{D_h}\right)}\right]\right\}^{16}\,, \tag{531}$$

and

$$b = \left(\frac{3.753\times 10^4}{\text{Re}}\right)^{16}\,, \tag{532}$$

with ϵ being the surface roughness.

The Reynolds number is computed as

$$\text{Re} = \max\left(\frac{\rho|u|D_h}{\mu}, 10\right)\,, \tag{533}$$

where μ is the dynamic viscosity, and a lower limit of 10 is imposed.

7.1.2 Two-Phase Flow: Pre-CHF Flow Regimes

For two-phase flow regimes, the two-phase multiplier concept is used to determine the two-phase flow wall drag. By comparing the 7-equation model presented in Section 3.1 and the traditional 6-equation two-fluid model used in TRACE, it can be observed that

$$F_{\text{wall},\ell}^{\text{friction}} = AC_{\text{wall},\ell}u_{liq}|u_{liq}| \tag{534}$$

and

$$F_{\text{wall},v}^{\text{friction}} = AC_{\text{wall},v}u_{vap}|u_{vap}|\,, \tag{535}$$

where $C_{\text{wall},\ell}$ and $C_{\text{wall},v}$ are the phasic wall drag coefficients used in TRACE, A is the pipe cross-sectional area, and u_{liq} and u_{vap} are the phasic velocities. For pre-CHF two-phase flow regimes, wall drag coefficients are modeled in bubbly/slug and annular/mist flow regimes. For all pre-CHF two-phase flow regimes, it is assumed that all of the wall drag is applied to the liquid phase alone.

7.1.2.1 Bubbly/Slug Flow Regime When wall nucleate boiling effect is not considered, the liquid phase wall drag coefficient is modeled as

$$C_{\text{wall},\ell} = f_{\text{wall},\ell} \frac{2\rho_{liq}}{D_h} \,, \tag{536}$$

with $f_{\text{wall},\ell}$ calculated from the Churchill formula using the liquid phase Reynolds number:

$$\text{Re}_{2\Phi,liq} = \frac{(1-\alpha_{vap})\rho_{liq}|u_{liq}|D_h}{\mu_{liq}} \,. \tag{537}$$

to consider the two-phase flow vapor volume fraction effect. When nucleate boiling effect is considered, following TRACE [2], a correction factor is introduced, and the wall drag coefficient is then modeled as

$$C_{\text{wall},\ell} = f_{\text{wall},\ell} \frac{2\rho_{liq}}{D_h} (1 + C_{NB})^2 \,, \tag{538}$$

where the empirical coefficient C_{NB} is given as

$$C_{NB} = \min \left\{ 2,\, 155 \frac{d_B}{D_h} [\alpha_{vap}(1-\alpha_{vap})]^{0.62} \right\} \,, \tag{539}$$

where

$$\frac{d_B}{D_h} = 0.015 \left(\frac{\sigma}{\tau_w D_h} \right)^{0.5} \,, \tag{540}$$

and

$$\tau_w = \frac{f_{\text{wall},\ell}}{2} \rho_{liq} u_{liq}^2 \,. \tag{541}$$

The gas phase wall drag is assumed to be zero: $C_{\text{wall},v} = 0$.

7.1.2.2 Annular/Mist Flow Regime As in TRACE [2], in the annular/mist flow regime, if the channel surface is considered to be fully covered by liquid film, the wall drag coefficient for the liquid phase is modeled as

$$C_{\text{wall},\ell} = f_{\text{film}} \frac{2\rho_{liq}}{D_h} \,, \tag{542}$$

where f_{film} is the friction factor for the annular flow regime and is given by a power law combination of the laminar and turbulent values:

$$f_{\text{film}} = (f_{\text{lam}}^3 + f_{\text{turb}}^3)^{1/3} \,. \tag{543}$$

Here the friction factor for the laminar regime is modeled as

$$f_{\text{lam}} = \begin{cases} \frac{16}{\text{Re}_{2\Phi,liq}} & \alpha_{vap} \leq 0.95 \\ \frac{\left[16+8\left(\frac{\alpha_{vap}-0.95}{0.99-0.95}\right)\right]}{\text{Re}_{2\Phi,liq}} & 0.95 < \alpha_{vap} < 0.99 \\ \frac{24}{\text{Re}_{2\Phi,liq}} & \alpha_{vap} \geq 0.99 \end{cases} . \tag{544}$$

The linear interpolation between vapor volume fraction 0.95 and 0.99 reflects the transition between pipe geometry correlation and parallel plate geometry, which is more appropriate for thinner film with void fraction larger than 0.99. The friction factor for the turbulent regime is modeled as

$$f_{\text{turb}} = \frac{1}{\left\{3.6 \log_{10}\left[\frac{6.9}{\text{Re}_{2\Phi,liq}} + \left(\frac{\epsilon/D}{3.7}\right)^{1.11}\right]\right\}^2} \,. \tag{545}$$

In these correlations, $\text{Re}_{2\Phi,liq}$ is calculated using equation (537). In addition, according to the TRACE closure module [111], the value of $\text{Re}_{2\Phi,liq}$ is limited to be larger than 1 in Equation (544), and to be larger than 100 in Equation (545). The vapor phase wall drag is assumed to be zero, i.e., $C_{\text{wall},v} = 0$, if the channel surface is fully covered by liquid film.

When the channel surface is considered to be partially covered by liquid film, taken as the condition wherein liquid film is present with thickness smaller than 25 μm, the wall drag coefficient for the liquid phase is modeled as

$$C_{\text{wall},\ell} = f_{\text{wet}} f_{\text{film}} \frac{2\rho_{liq}}{D_h} \,, \tag{546}$$

where

$$f_{\text{wet}} = \frac{(1-\alpha_{vap})D_h}{4(25\times 10^{-6})} \,. \tag{547}$$

In this case, the vapor phase wall drag coefficient is no longer zero and is modeled as

$$C_{\text{wall},v} = (1 - f_{\text{wet}}) f_{2\Phi,vap} \frac{2\rho_{vap}}{D_h} \,, \tag{548}$$

with $f_{2\Phi,vap}$ denoting the vapor phase friction factor, which is computed with the Churchill formula with the vapor phase Reynolds number,

$$\text{Re}_{2\Phi,vap} = \frac{\alpha_{vap}\rho_{vap}|u_{vap}|D_h}{\mu_{vap}} \,. \tag{549}$$

7.1.2.3 Transition between Bubbly/Slug and Annular/Mist Flow Regimes Following TRACE [2], a smooth transition from the bubbly/slug flow regime to the annular/mist flow regime is used. The transition is assumed to occur between vapor volume fractions of 0.8 and 0.9; here, the liquid phase wall drag coefficient is modeled as

$$C_{\text{wall},\ell} = wf_{\text{BS}}(C_{\text{wall},\ell})_{\text{BS}} + (1 - wf_{\text{BS}})(C_{\text{wall},\ell})_{\text{AM}} \,, \tag{550}$$

where subscripts BS and AM represent bubbly/slug flow and annular mist flow conditions, respectively. The weight factor is defined as

$$wf_{\text{BS}} = \frac{0.9 - \alpha_{vap}}{0.9 - 0.8} \,. \tag{551}$$

7.1.2.4 Horizontal Stratified Flow For horizontal stratified flow, both phases are in contact with the channel wall, so the wall drag coefficient is modeled separately for each phase. For phase *k* = {*liq, vap*}, the wall friction factor is calculated from the Churchill formula with the phasic Reynolds number, Re_k, which is defined as

$$\text{Re}_k = \frac{\rho_k u_k D_{h,k}}{\mu_k} \,, \tag{552}$$

with $D_{h,k}$ being the phasic hydraulic diameter. Following the definition used in TRACE [2], the two phasic hydraulic diameters are defined as

$$D_{h,liq} = \frac{4A_{liq}}{S_{liq}} = \frac{4A\alpha_{liq}}{S_{liq}} \tag{553}$$

and

$$D_{h,vap} = \frac{4A_{vap}}{S_{vap} + S_{int}} = \frac{4A\alpha_{vap}}{S_{vap} + S_{int}} \,. \tag{554}$$

As shown in Figure 6, S_{liq}, S_{vap}, and S_{int} can be obtained as

$$S_{liq} = D\theta\,, \tag{555}$$

$$S_{vap} = D(\pi - \theta)\,, \tag{556}$$

and

$$S_{int} = D\sin\theta\,, \tag{557}$$

where θ is in radians and is computed as

$$\theta = \cos^{-1}\left(\frac{D/2 - h}{D/2}\right) = \cos^{-1}\left(1 - \frac{2h}{D}\right)\,. \tag{558}$$

The depth of the liquid phase, h, can be obtained as a curve-fitting function for round pipe as

$$\frac{h}{D} = \begin{cases} 1 - 7.0612668\alpha_{vap} & \alpha_{vap} \leq 0.001 \\ 1.0 - a_1\alpha_{vap}^{2/3} - a_2\alpha_{vap} - a_3\alpha_{vap}^2 & 0.001 < \alpha_{vap} \leq 0.5 \\ b_1\alpha_{liq}^{2/3} + b_2\alpha_{liq} + b_3\alpha_{liq}^2 & 0.5 < \alpha_{vap} \leq 0.999 \\ 7.0612668\alpha_{liq} & \alpha_{vap} > 0.999 \end{cases}\,, \tag{559}$$

where $a_1 = 0.70269591$, $a_2 = 0.034146667$, $a_3 = 0.161023911$, $b_1 = 0.70269591$, $b_2 = 0.034146667$, and $b_3 = 0.161023911$.

Then, considering the wetted perimeter for each phase, the wall drag coefficients are modeled as

$$C_{\text{wall},k} = f_{\text{wall},k}\frac{2\rho_k}{D_{h,wet}} = f_{\text{wall},k}\frac{\rho_k}{2}\frac{S_k}{A}\,, \tag{560}$$

where

$$D_{h,wet} = \frac{4A}{S_k}\,. \tag{561}$$

That is,

$$C_{\text{wall},\ell} = f_{\text{wall},\ell}\frac{\rho_{liq}}{2}\frac{S_{liq}}{A} = f_{\text{wall},\ell}\frac{2\rho_{liq}}{D}\frac{\theta}{\pi}\,, \tag{562}$$

and

$$C_{\text{wall},v} = f_{\text{wall},v}\frac{\rho_{vap}}{2}\frac{S_{vap}}{A} = f_{\text{wall},v}\frac{2\rho_{vap}}{D}\frac{(\pi - \theta)}{\pi}\,. \tag{563}$$

It must be noted that all previous derivations are based on a round pipe assumption.

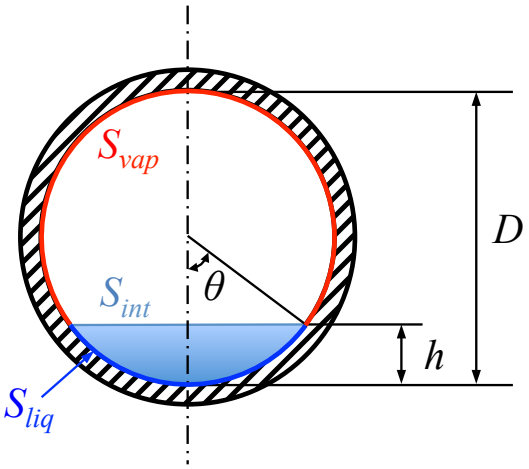

Figure 6. Schematic drawing of the horizontal stratified flow.

7.1.2.5 Transition between Horizontal Stratified and Non-Stratified Flow If the flow regime is considered to be in transition between stratified and non-stratified condition, a linear interpolation is used:

$$C_{\text{wall},k} = wf_{\text{strat}}(C_{\text{wall},k})_{\text{strat}} + (1 - wf_{\text{strat}})(C_{\text{wall},k})_{\text{non-strat}} \,, \tag{564}$$

where $(C_{\text{wall},k})_{\text{strat}}$ and $(C_{\text{wall},k})_{\text{non-strat}}$ are wall drag coefficients for stratified and non-stratified conditions, respectively. Following the TRACE manual [2], the weighting factor, wf_{strat}, is determined from several factors:

$$wf_{\text{strat}} = wf_{\text{TD}} wf_{\text{CW}} wf_{\text{CCFL}} \,, \tag{565}$$

where wf_{TD} is the weighting factor for the Taitel-Dukler transition from stratified flow condition, wf_{CF} is the weighting factor for mass flux, and wf_{CCFL} is the weighting factor for counter-current flow limit.

The weighting factor for the Taitel-Dukler transition from stratified flow condition, wf_{TD}, is defined as

$$wf_{\text{TD}} = \max\left[0, \min\left(1,\, 2 - \frac{|u_r|}{u_{r,crit}}\right)\right] \,, \tag{566}$$

with $u_r = u_{vap} - u_{liq}$, and

$$u_{r,crit} = \left(1 - \frac{h_l}{D}\right) \left[\frac{g\Delta\rho \cos\phi A_{vap}}{\rho_g (dA_l/dh_l)}\right]^{1/2} , \tag{567}$$

where ϕ is the pipe inclination angle (for example, $\phi = 0$ for horizontal pipe and $\pi/2$ for vertical pipe), and (dA_l/dh_l) is the derivative of liquid flow area with respect to height, which is obtained as

$$\frac{dA_l}{dh_l} = \max\left\{ D\left[1 - \left(\frac{2h_l}{D} - 1\right)^2\right]^{1/2}, 0.0001D \right\} . \tag{568}$$

As in the TRACE manual [2], the weighting factor for mass flux, wf_{CW}, is calculated as

$$wf_{\text{CW}} = \max\left[0,\, \min\left(1, \frac{2700 - G}{2700 - 2000}\right)\right] , \tag{569}$$

where G is the total mass flux of the two-phase mixture.

Again as in the TRACE manual [2], the weighting factor for counter-current flow limit, wf_{CCFL}, is calculated as

$$wf_{\text{CCFL}} = \max\left[0,\, \min\left(1, \frac{1.2 - \left(j^*_{vap}\right)^{1/2} - \left(j^*_{liq}\right)^{1/2}}{1.2 - 0.65}\right)\right] , \tag{570}$$

with the non-dimensional superficial phasic velocities defined as

$$j^*_k = |j_k| \left(\frac{\rho_k}{g\Delta\rho D}\right)^{1/2} . \tag{571}$$

Here the superficial phasic velocity is defined by $j_k = \alpha_k u_k$.

7.1.3 Two-Phase Flow: Post-CHF Flow Regimes

For post-CHF conditions, wall drag is modeled for two flow regimes: inverted annular and dispersed flow regimes. A linear interpolation is used in the transitional region between these two regimes.

7.1.3.1 Inverted Annular Flow The inverted annular flow regime is applied for vapor volume fraction less than 0.6 under the post-CHF conditions. In this regime, it is assumed that only the vapor phase is in contact with the wall, and the wall-to-liquid drag is zero, which leads to $C_{\text{wall},\ell} = 0$. For the wall drag to the vapor phase, a similar two-phase multiplier concept is used. The wall drag coefficient for the vapor phase is modeled as

$$C_{\text{wall},v} = \Phi_{vap}^2 \left[\frac{2 f_{2\Phi,vap} \alpha_{vap}^2 \rho_{vap}}{D_h} \right] . \tag{572}$$

By analogy with annular flow, the two-phase multiplier for the inverted annular flow is

$$\Phi_{vap}^2 = \frac{1}{\alpha_{vap}^2} , \tag{573}$$

so the wall coefficient is

$$C_{\text{wall},v} = f_{2\Phi,vap} \frac{2\rho_{vap}}{D_h} . \tag{574}$$

Here, the friction factor $f_{2\Phi,vap}$ is the Fanning friction factor calculated from the Churchill formula, Equation (530), with

$$\text{Re}_{2\Phi,vap} = \frac{|G_{vap}| D_h}{\mu_{vap}} = \frac{\alpha_{vap} \rho_{vap} |u_{vap}| D_h}{\mu_{vap}} . \tag{575}$$

7.1.3.2 Dispersed Flow The dispersed flow regime is applied for vapor volume fraction larger than 0.9 under the post-CHF conditions. The wall drag coefficient for the dispersed flow regime is modeled as

$$C_{\text{wall},v} = f_{pg} \frac{2\rho_{vap}}{D_h} , \tag{576}$$

where f_{pg} is the effective friction fraction for dilute suspensions:

$$f_{pg} = f_{wall,vap} \left[1 + \min(12, \text{LF}) \right]^{0.3} . \tag{577}$$

In this equation, LF is the loading factor that is calculated as

$$\text{LF} = \frac{(1 - \alpha_{vap}) \rho_{liq} u_{liq}}{\alpha_{vap} \rho_{vap} u_{vap}} . \tag{578}$$

The vapor phase wall friction factor, $f_{wall,vap}$, is calculated from the Churchill formula, Equation (530), similar to the inverted annular flow condition.

7.1.3.3 Transition between Inverted Annular and Dispersed Flow Under post-CHF flow conditions, when the vapor volume fraction is between 0.6 and 0.9, a linear interpolation method is used. A dispersed flow weighting factor is first defined as

$$wf_{\text{DF}} = \frac{\alpha_{vap} - 0.6}{0.9 - 0.6}\,, \tag{579}$$

and then the wall drag coefficient for this transitional region is interpolated as

$$C_{\text{wall},v} = (1 - wf_{\text{DF}})\,(C_{\text{wall},v})_{\text{IA}} + wf_{\text{DF}}(C_{\text{wall},v})_{\text{DF}}\,, \tag{580}$$

where subscripts IA and DF represent inverted annular flow and dispersed flow conditions, respectively.

7.2 Interfacial Drag

By comparing the 7-equation model presented in Section 3.1 and the 6-equation two-fluid model used in TRACE [2], it can be found that

$$F_{\text{friction},vap} = -F_{\text{friction},liq} = AC_i u_r|u_r|\,, \tag{581}$$

where A is the pipe cross-sectional area, C_i is the interfacial drag coefficient, and u_r is the relative velocity between the two phases:

$$u_r = u_{vap} - u_{liq}\,. \tag{582}$$

All correlations discussed in this section closely follow those used in TRACE [2].

7.2.1 Pre-CHF Flow Regimes

For pre-CHF regimes in vertical pipes/bundles, interfacial drag is modeled in three major regimes: bubbly, cap/slug, and annular/mist. The bubbly and slug flow regimes are grouped together using a similar approach that is based on the drift-flux model concept.

7.2.1.1 Bubbly/Slug Flow For the combined bubbly/slug flow regime, the interfacial drag coefficient, C_i, is modeled as

$$C_{i,\text{BS}} = \frac{\alpha_{vap}(1 - \alpha_{vap})^3 g\Delta\rho}{\bar{v}_{gj}^2} P_s\,, \tag{583}$$

where the subscript BS stands for the combined bubbly and slug flow regimes, P_S is the profile slip factor, $\bar{v}_{gj}$ is the weighted area-average value of the drift velocity, and $\Delta\rho$ is the density difference between the two phases: $\Delta\rho = \rho_{liq} - \rho_{vap}$.

The profile slip factor is calculated as

$$P_s = \frac{\left(\frac{1-C_0\alpha_{vap}}{1-\alpha_{vap}} u_{vap} - C_0 u_{liq}\right)^2}{u_r^2 + \epsilon} \,. \tag{584}$$

Here ϵ is a small positive value (10^{-8}) that is added to the denominator to avoid potential divided-by-zero numerical issue. The distribution coefficient, C_0, and the weighted area-average value of the drift velocity, $\bar{v}_{gj}$, are both dependent on flow regime (dispersed bubbly flow or slug flow) and flow channel geometry (pipe or rod bundle).

7.2.1.1.1 Pipe Geometry For pipe flow, the two drift-flux-model-related parameters $\bar{v}_{gj}$ and C_0 are modeled separately in each of the dispersed bubbly flow and cap/slug flow conditions.

For the dispersed bubbly flow regime in a pipe, they are modeled as

$$(\bar{v}_{gj})_{\text{DB}} = \sqrt{2}\left(\frac{\sigma g \Delta\rho}{\rho_{liq}^2}\right)^{1/4} \tag{585}$$

and

$$C_{0,\text{DB}} = 1.2 - 0.2\sqrt{\frac{\rho_{vap}}{\rho_{liq}}} \,, \tag{586}$$

where σ is the surface tension of the interface, and subscript DB represents the dispersed bubbly flow regime.

For the cap/slug flow regime, the distribution coefficient, C_0, is modeled the same as in the dispersed bubbly flow regime: $C_{0,\text{CS}} = C_{0,\text{DB}}$. The subscript CS represents the cap/slug flow regime. For the cap/slug flow regime, the weighted drift velocity is given by

$$(\bar{v}_{gj})_{\text{CS}} = \bar{v}_{gj}^{+}\left(\frac{\sigma g \Delta\rho}{\rho_{liq}^2}\right)^{1/4} \,, \tag{587}$$

where the non-dimensional drift velocity is modeled as

$$\bar{v}_{gj}^{+} = 0.0019\left(\min[30, D_h^*]\right)^{0.809}\left(\frac{\rho_{vap}}{\rho_{liq}}\right)^{-0.157} N_{\mu,liq}^{-0.562} \,. \tag{588}$$

Here, the non-dimensional hydraulic diameter is given as

$$D_h^* = \frac{D_h}{\sqrt{\frac{\sigma}{g\Delta\rho}}}\,, \tag{589}$$

and the liquid viscosity number is defined as

$$N_{\mu,liq} \equiv \frac{\mu_{liq}}{\left(\rho_{liq}\sigma\sqrt{\frac{\sigma}{g\Delta\rho}}\right)^{1/2}}\,. \tag{590}$$

The quantity $\sqrt{\frac{\sigma}{g\delta\rho}}$ is a *capillary number*, Ca, which will recur subsequently, so for convenience, this definition shall be made:

$$\mathrm{Ca} = \sqrt{\frac{\sigma}{g\Delta\rho}}\,.$$

A transition region is added between the dispersed bubbly and cap/slug flow regimes. In this transition region, a simple linear interpolation is used to model the weighted drift velocity:

$$\bar{v}_{gj} = wf_{\text{DB}}\left(\bar{v}_{gj}\right)_{\text{DB}} + \left(1 - wf_{\text{DB}}\right)\left(\bar{v}_{gj}\right)_{\text{CS}}\,. \tag{591}$$

The weighted drift velocity $\left(\bar{v}_{gj}\right)_{\text{DB}}$ and $\left(\bar{v}_{gj}\right)_{\text{CS}}$ are calculated from Equations (585) and (587), respectively. The linear interpolation coefficient wf_{DB} is calculated as

$$wf_{\text{DB}} = \frac{\alpha_{vap,\text{CS}} - \alpha_{vap}}{\alpha_{vap,\text{CS}} - \alpha_{vap,\text{DB}}}\,, \tag{592}$$

where

$$\alpha_{vap,\text{DB}} = 0.2\ \min\left[1, \frac{T_{sat} - T_l}{5}\right]\,, \tag{593}$$

and

$$\alpha_{vap,\text{CS}} = \alpha_{vap,\text{DB}} + 0.1\,, \tag{594}$$

with $T_{sat} = T_{int}$.

7.2.1.1.2 Rod Bundle Geometry For rod bundle geometry, the Bestion model is used to calculate $\bar{v}_{gj}$, which is

$$\bar{v}_{gj} = 0.188\sqrt{g\Delta\rho D_h/\rho_{vap}}\,. \tag{595}$$

The distribution parameter, C_0, for rod bundle geometry is simply set to be 1.

7.2.1.2 Annular/Mist Flow For the annular/mist flow regime, the interfacial drag comes from two parts: the interfacial drag between the liquid film and the vapor core, and the interfacial drag between the liquid droplets and the vapor core.

Considering the velocity difference between the liquid film and the liquid droplets entrained in the vapor core, the overall interfacial drag coefficient for the annular/mist flow regime is modeled as

$$C_{i,\text{AM}} = C_{i,\text{film}} + C_{i,\text{drop}}\frac{u_{r,d}^2}{(u_{vap}-u_{liq})^2}\,, \tag{596}$$

where $C_{i,\text{film}}$ and $C_{i,\text{drop}}$ are the interfacial drag coefficients for the liquid film and droplets, respectively, and $u_{r,d}$ is the droplet relative velocity. In code implementation, the magnitude of $u_{vap} - u_{liq}$ is defined to be larger than 10^{-6} m/s. The interfacial drag coefficient for the liquid film part is modeled as

$$C_{i,\text{film}} = f_{i,\text{film}}A_{int,\text{film}}\frac{1}{2}\rho_{vap}\,. \tag{597}$$

Here, the specific interfacial area (interfacial area per unit volume) is computed from

$$A_{int,\text{film}} = \frac{4}{D_h}\sqrt{\alpha_{vap}}\,, \tag{598}$$

and the friction factor for the liquid film is approximated by

$$f_{i,\text{film}} = 0.005[1 + 75(1-\alpha_{vap})]\,. \tag{599}$$

For the interfacial drag between the liquid droplets and the vapor core, the fraction of the liquid flow that is entrained as droplets in the vapor core must be estimated first. For small diameter pipes ($D_h \leq 3.2\,\text{cm}$), the entrainment fraction is modeled as

$$E_\infty = \tanh\left[7.25\times10^{-7}\,\text{We}_{vap}^{1.25}\ \min(6400,\text{Re}_f)^{0.25}\right]\,, \tag{600}$$

where the liquid film Reynolds number is defined as

$$\mathrm{Re}_f = \frac{(1-\alpha_{vap})\rho_{liq}u_{liq}D_h}{\mu_{liq}}\,, \tag{601}$$

and the effective Weber number for entrainment is defined as

$$\mathrm{We}_{vap} = \frac{\rho_{vap}j_{vap}^2 D_h}{\sigma}\left(\frac{\Delta\rho}{\rho_{vap}}\right)^{1/3}\,, \tag{602}$$

with j_{vap} being the superficial velocity of the vapor phase. The superficial phasic velocities, or phasic volumetric fluxes, are defined to be

$$j_k = \alpha_k u_k$$

for $k = \{liq, vap\}$.

For large diameter pipes, ($D_h > 3.2\,\mathrm{cm}$), the entrained fraction is modeled as

$$E_\infty = 0.015 + 0.44\log_{10}\left[0.9245\left(\frac{\pi_2}{\pi_{2,crit}}\right)^2\right]\,, \tag{603}$$

where π_2 is the non-dimensional vapor velocity that is defined as

$$\pi_2 = \frac{j_{vap}\mu_{vap}}{\sigma}\sqrt{\frac{\rho_{vap}}{\rho_{liq}}}\,, \tag{604}$$

and $\pi_{2,crit} = 2.46\times10^{-4}$ is the inception criteria for liquid droplet entrainment. In code implementation, the value of $0.9245\left(\pi_2/\pi_{2,crit}\right)^2$ is limited to be larger than 10^{-10}, and the final value of entrained fraction, E_∞, is limited to be larger or equal to zero.

Following TRACE [2], the interfacial drag coefficient for the droplets is modeled as

$$C_{i,\text{drop}} = C_D A_{int,d}\frac{1}{2}\rho_{vap} \tag{605}$$

$$= C_D\ \rho_{vap}\frac{3\alpha_c\alpha_d}{4d_d}\,, \tag{606}$$

where α_c is the volume fraction of the annular core region (containing vapor and droplets), and α_d is the fraction of the annular core occupied by the droplets alone. For this equation, the following relation for the projected area of droplets per unit mixture volume was used:

$$A_{int,d} = \frac{3}{2}\frac{\alpha_{vap}\alpha_d}{(1-\alpha_d)d_d}\,. \tag{607}$$

The drop drag coefficient is modeled as

$$C_D = \frac{24}{\mathrm{Re}_d}\left(1 + 0.1\mathrm{Re}_d^{0.75}\right) , \tag{608}$$

with the drop Reynolds number defined as

$$\mathrm{Re}_d = \frac{\rho_{vap}|u_{vap} - u_d|d_d}{\mu_m} . \tag{609}$$

Here, d_d is the droplet Sauter mean diameter, and μ_m is the mixture viscosity. The mixture viscosity is given by

$$\mu_m = \frac{\mu_{vap}}{(1-\alpha_d)^{2.5}} . \tag{610}$$

The fraction of the annular vapor core occupied by the droplets is approximated by

$$\alpha_d = E_\infty \frac{|j_{liq}|}{|j_{vap}|} . \tag{611}$$

To avoid numerical issues, the denominator is modified to $|j_{vap}| + 10^{-6}$. The value of α_d is also limited by

$$\alpha_d \leq E_\infty(1 - \alpha_{vap}) , \tag{612}$$

where again, j_{liq} and j_{vap} are the superficial velocities of the two phases. The Sauter mean diameter is modeled as

$$d_d = 0.008\left(\frac{\sigma}{\rho_{vap}j_{vap}^2}\right)\mathrm{Re}_{vap}^{2/3}\left(\frac{\mu_{vap}}{\mu_{liq}}\right)^{2/3}\left(\frac{\rho_{vap}}{\rho_{liq}}\right)^{-1/3} , \tag{613}$$

where the gas Reynolds number is defined as

$$\mathrm{Re}_{vap} = \frac{\rho_{vap}|j_{vap}|D_h}{\mu_{vap}} . \tag{614}$$

In addition, the Sauter mean diameter is limited to be

$$84\mu\mathrm{m} \leq d_d \leq 4\,\mathrm{mm} . \tag{615}$$

The drop relative velocity, $u_{r,d} \equiv u_{vap} - u_d$, is modeled as

$$u_{r,d} = \begin{cases} 1.718\sqrt{d_d}\left[\frac{g\Delta\rho}{\rho_{vap}}\right]^{1/2}(1-\alpha_d)^{1.5} & d_d \leq d_{d,Newton} \\ \sqrt{2}\left[\frac{\sigma g\Delta\rho}{\rho_{vap}^2}\right]^{1/4}(1-\alpha_d)^{1.5} & d_d > d_{d,Newton} \end{cases} , \tag{616}$$

with

$$d_{d,Newton} = 0.678\sqrt{\frac{\sigma}{g\Delta\rho}} = 0.678\mathrm{Ca}\,. \tag{617}$$

When $u_{r,d}$ is used to calculate droplet Reynolds number, its value is also defined to be larger than $10^{-6}\,\mathrm{m/s}$ to avoid a zero-valued Reynolds number that causes numerical issues when evaluating the drop drag coefficient, C_D.

7.2.1.3 Mixing of Bubbly/Slug and Annular/Mist Flows Finally, in order to avoid a discontinuous change of the interfacial drag coefficient between the combined bubbly/slug flow regime and the annular/mist flow regime, the interfacial coefficient is averaged using a simple power law weighting scheme:

$$C_i = \sqrt{C_{i,\mathrm{BS}}^2 + C_{i,\mathrm{AM}}^2}\,, \tag{618}$$

where the subscripts BS and AM stand for bubbly/slug and annular/mist flow conditions, respectively.

7.2.1.4 Horizontal Stratified Flow For horizontal stratified flow conditions, the interfacial drag coefficient is given as

$$C_{i,\mathrm{strat}} = \frac{1}{2}\rho_{vap} f_i A_{int}\,. \tag{619}$$

The interfacial area per unit volume is defined as

$$A_{int} = \frac{S_{int}}{A}\,, \tag{620}$$

with S_{int} defined the same as in Equation (557). Following TRACE [2], the interfacial friction factor is given as

$$f_i = 1.84 f_{\mathrm{wall},v}\,, \tag{621}$$

where $f_{\mathrm{wall},v}$ is the single-phase wall drag friction factor for the gas phase. Its value is evaluated using the Churchill formula with the gas phase Reynolds number,

$$\mathrm{Re}_{vap} = \frac{\rho_{vap} u_{vap} D_{h,vap}}{\mu_{vap}}\,, \tag{622}$$

where the gas phase hydraulic diameter $D_{h,vap}$ is calculated using Equation (554).

7.2.1.5 Transition between Horizontal Stratified and Non-Stratified Flow If the flow regime is considered to be in a transition between stratified and non-stratified condition, a power law based interpolation is used:

$$C_i = C_{i,\text{strat}}^{n} C_{i,\text{non-strat}}^{1-n} , \tag{623}$$

where subscripts 'strat' and 'non-strat' correspond to stratified flow and non-stratified flow conditions, respectively. The exponent, n, is set to wf_{strat}, defined in Equation (565).

7.2.2 Post-CHF Flow Regimes

The post-CHF regimes are defined when the surface temperature exceeds the Leidenfrost point, where the liquid phase cannot contact the hot surface. As in TRACE [2], the post-CHF regimes take place between the bottom quench front and the top quench front. The post-CHF flow regimes include inverted annular, inverted slug, and dispersed flow regimes. As in TRACE [2], the post-CHF flow regimes are void-fraction-dependent, as illustrated in Figure 7.

7.2.2.1 Inverted Annular Flow For the inverted annular regime (void fraction below 0.6), the interfacial drag coefficient is given by

$$C_{i,\text{IA}} = \frac{1}{2} \rho_{vap} f_{i,\text{IA}} A_{int} , \tag{624}$$

where the subscript IA represents inverted annular flow condition. A_{int} is the interfacial area per unit volume and $f_{i,IA}$ is the interfacial friction factor. Assuming the liquid core is circular, the interfacial area per unit volume can be calculated as

$$A_{int} = 4 \frac{\sqrt{1-\alpha_{vap}}}{D_h} . \tag{625}$$

The interfacial friction factor, $f_{i,IA}$, is calculated as a power-law-averaged value of both the smooth and wavy interface conditions:

$$f_{i,\text{IA}} = \sqrt{f_{i,\text{smooth}}^2 + f_{i,\text{wavy}}^2} , \tag{626}$$

where $f_{i,\text{smooth}}$ is for the laminar flow with a smooth interface, and $f_{i,wavy}$ is for the case with a way interface:

$$f_{i,\text{smooth}} = \frac{144}{(\delta^*)^3} , \tag{627}$$

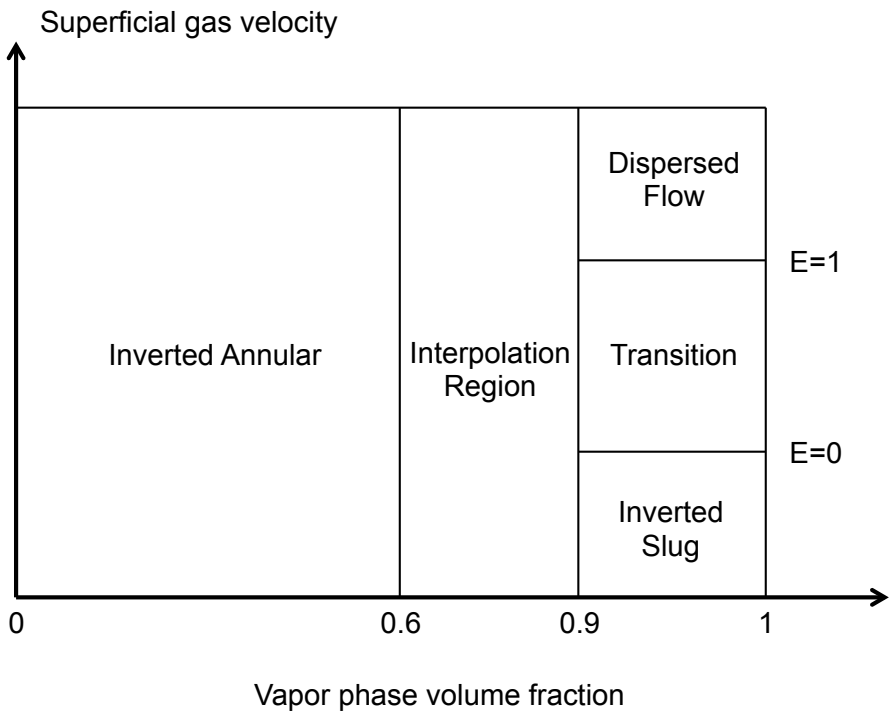

Figure 7. Post-CHF flow regime map [2].

$$f_{i,\text{wavy}} = 0.35 \left(\frac{\delta}{\text{La}} \right)^{0.72} . \tag{628}$$

The vapor film thickness is calculated as

$$\delta = \frac{D_h}{2} \left(1 - \sqrt{1 - \alpha_{vap}} \right) \tag{629}$$

for pipe geometry, and

$$\delta = \frac{D_h}{2} \left\{ \sqrt{1 + \alpha_{vap} \left[\frac{4}{\pi} \left(\frac{P}{D_r} \right)^2 - 1 \right]} - 1 \right\} \tag{630}$$

for rod bundle geometry. The non-dimensional vapor film thickness for evaluating the smooth interface interfacial friction factor is

$$\delta^* = \delta \left(\frac{\rho_{vap} g \Delta\rho}{\mu_{vap}^2} \right)^{1/3} . \tag{631}$$

The Laplace number is given by

$$\mathrm{La} = \sqrt{\frac{\sigma}{g\Delta\rho}}\,. \tag{632}$$

7.2.2.2 Inverted Slug Flow For the inverted slug regime, the interfacial drag coefficient is given by

$$C_{i,\mathrm{IS}} = \frac{1}{24}\frac{\rho_{vap}}{\mathrm{La}}\frac{1-\alpha_{vap}}{\alpha_{vap}^{1.8}}\,, \tag{633}$$

following the recommendation made in the TRACE manual [2] that the coefficient has been adjusted to better match FLECHT-SEASET high flooding rate reflood data.

7.2.2.3 Dispersed Flow For the post-CHF dispersed flow regime, the interfacial drag coefficient is given by

$$C_{i,\mathrm{DF}} = \frac{1}{2}\rho_{vap}C_{D,\mathrm{MP}}A_{int,p}\,, \tag{634}$$

where $C_{D,\mathrm{MP}}$ is the drag coefficient corrected from multi-particle effects and $A_{int,p}$ is the projected area per unit volume. The projected area per unit volume is calculated from the droplet Sauter mean diameter by

$$A_{int,p} = \frac{6(1-\alpha_{vap})}{4d_{SM}}\,. \tag{635}$$

The Sauter mean diameter is estimated as about one-third of the maximum droplet diameter:

$$d_{\mathrm{SM}} = \frac{1}{3}d_{max}\,. \tag{636}$$

The maximum droplet diameter is modeled differently for downflow and upflow conditions. For downflow condition, the maximum droplet diameter is estimated as

$$d_{max,down} = \min\left[3.52\mathrm{La}, D_h\right]\,. \tag{637}$$

For upflow condition, the maximum droplet diameter is estimated as

$$d_{max,up} = \min\left[5.07\mathrm{La}N_{\mu g}^{0.176}, \frac{18\sigma}{\rho_{vap}j_{vap}^2}, d_{max,down}\right]\,. \tag{638}$$

The multi-particle drag coefficient can be obtained from the single-particle model:

$$C_{D,\mathrm{MP}} = \frac{C_{D,\mathrm{SP}}}{\alpha_{vap}^{1.8}}\,. \tag{639}$$

The single-partical drag coefficient is calculated as

$$C_{D,\text{SP}} = \max\left[\frac{24}{\text{Re}_d}\left(1 + 0.15\text{Re}_d^{0.687}\right), 0.44\right], \tag{640}$$

where the drop Reynolds number is defined by

$$\text{Re}_d = \frac{\rho_{vap} V_r d_{SM}}{\mu_{vap,film}}. \tag{641}$$

The relative velocity, V_r, is calculated as

$$V_r = \max\left[\frac{|u_{vap} - u_{liq}|}{\alpha_{vap}^{1.4}}, V_\infty\right], \tag{642}$$

where V_∞ is the terminal velocity for large spherical drops. The vapor-phase viscosity, $\mu_{vap,film}$, should be estimated at the film temperature given by

$$T_{\text{film}} = \frac{T_{vap} + T_{sat}}{2}. \tag{643}$$

The non-dimensional terminal velocity for large spherical drops is modeled as

$$V_\infty^* = 0.693\left(r^*\right)^{0.858}, \tag{644}$$

with

$$V_\infty^* = V_\infty\left[\frac{\rho_{vap}^2}{\mu_{vap} g \Delta\rho}\right]^{1/3}, \tag{645}$$

and

$$r^* = r\left[\frac{\rho_{vap} g \Delta\rho}{\mu_{vap}^2}\right]^{1/3}, \tag{646}$$

where r is the droplet radius.

7.2.2.4 Transition between Inverted Annular and Inverted Slug Flow Regimes For void fractions between 0.6 and 0.9 under post-CHF condition, an interpolation region is applied between the inverted annular and inverted slug flow regimes. In this transition regime, the interfacial drag coefficient is computed as

$$C_i = wf_{\text{IA}} C_{i,\text{IA}} + (1 - wf_{\text{IA}}) C_{i,\text{IS}}, \tag{647}$$

where wf_{IA} is a spline weighting factor for inverted annular flow, given by

$$wf_{\text{IA}} = x(2 - x), \tag{648}$$

where

$$x = \frac{0.9 - \alpha_{vap}}{0.9 - 0.6}. \tag{649}$$

7.2.2.5 Transition between Inverted Slug and Dispersed Flow Regimes The transition between the inverted slug and dispersed flow regimes is determined from the liquid entrainment fraction. In this transition region, part of the liquid phase is considered to be entrained in the dispersed droplet form, and the remaining part is considered to be inverted slugs. The interfacial drag coefficient is modeled with a linear interpolation:

$$C_i = EC_{i,\text{DF}} + (1-E)C_{i,\text{IS}}\,, \tag{650}$$

where E is the entrainment fraction. The entrainment fraction is defined as

$$E = \frac{G_d}{G_{liq}}\,, \tag{651}$$

where G_d is the entrained mass flux, and G_{liq} is the magnitude of the liquid phase mass flux. In code implementation, G_{liq} is taken as the larger value between $|\alpha_{liq}\rho_{liq}u_{liq}|$ and 10^{-10}. As suggested in the TRACE manual [2], the entrained mass flux is modeled as

$$G_d = \begin{cases} 0 & j_{vap} \le j_{vap,crit} \\ 2.16\times10^{-4}\left[\left(\frac{j_{vap}}{j_{vap,crit}}\right)^3 - 1\right] N_{\mu g}^{0.236}\Delta\rho j_{vap} & j_{vap} > j_{vap,crit} \end{cases}\,, \tag{652}$$

where $j_{vap,crit}$ is the critical vapor phase superficial velocity:

$$j_{vap,crit} = 0.6\left[\frac{\sigma^{0.316}g^{0.228}\Delta\rho^{0.228}}{\rho_{vap}^{0.456}\mu_{vap}^{0.0879}}\right]\,. \tag{653}$$

The value of the entrainment fraction is also limited to be smaller or equal to 1.

7.3 Interfacial Heat Transfer

When the two phases are under thermal non-equilibrium conditions, the temperature difference between the two phases drives heat transfer towards their interface, which consequently leads to mass transfer (via evaporation or condensation) on the interface. Recall that the interfacial mass transfer term is modeled as

$$\Gamma_{liq\to vap}^{int}A_{int} = \frac{h_{conv,liq}A_{int}(T_{liq}-T_{int}) + h_{conv,vap}A_{int}(T_{vap}-T_{int})}{h_{vap,int}-h_{liq,int}}\,. \tag{654}$$

For two-phase flow, the volumetric heat transfer coefficient, $H_{ik} = h_{conv,k}A_{int}$, is discussed in this section for both pre-CHF and post-CHF flow regimes.

7.3.1 Pre-CHF Flow Regimes

Similar to interfacial drag, interfacial heat transfer correlations distinguish the following pre-CHF flow regimes: bubbly, cap/slug, annular/mist flow, and transition. The flow regime map for vertical pipes under pre-CHF conditions is shown in Fig. 8.

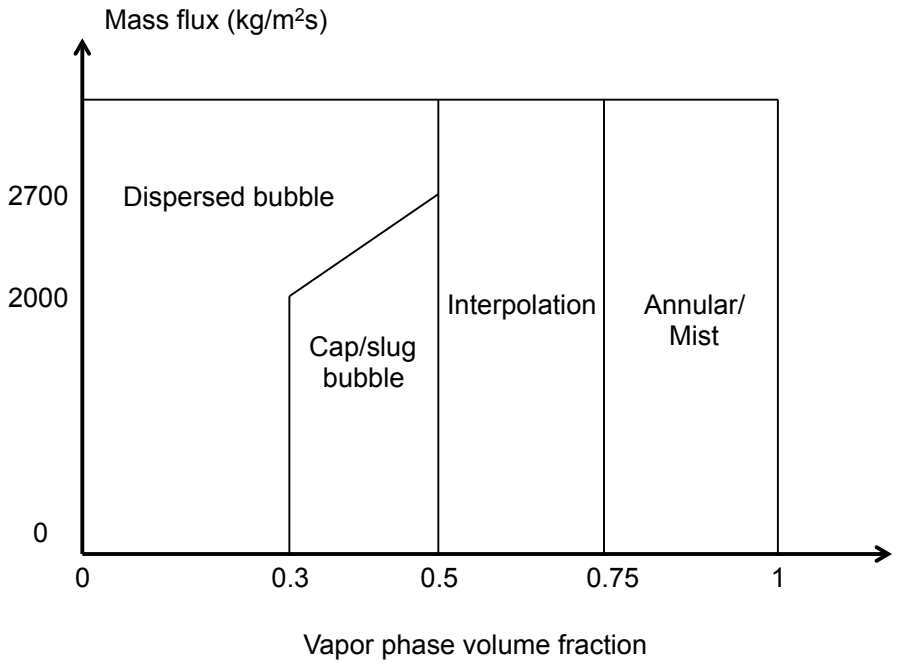

Figure 8. Flow regime map in vertical pipes under the pre-CHF conditions, for interfacial heat transfer [2].

7.3.1.1 Bubbly Flow For the bubbly flow regime, only the dispersed bubbles contribute to the interfacial heat transfer, and thus

$$(h_{conv,liq}A_{int})_{\text{Bubbly}} = (h_{conv,liq}A_{int})_{\text{DB}} \,. \tag{655}$$

Here, subscript DB stands for the dispersed bubbly flow. When evaluating interfacial heat transfer, the bubbly flow regime is defined as

$$\alpha_{vap} < \alpha_{vap,\text{DB}} \,, \tag{656}$$

where α_{DB} is modeled as

$$\alpha_{vap,\text{DB}} = \begin{cases} 0.3 & G \leq 2000\,\text{kg/m}^2\text{s} \\ 0.3 + 0.2\left(\frac{G-2000}{2700-2000}\right) & 2000 < G < 2700\,\text{kg/m}^2\text{s} \\ 0.5 & G \geq 2700\,\text{kg/m}^2\text{s} \end{cases}, \tag{657}$$

and G is the total mass flux of the two phases.

The interfacial area for the dispersed bubbles is simply modeled as

$$A_{int,\text{DB}} = \frac{6\alpha_{vap}}{d_{\text{DB}}}\,, \tag{658}$$

where the diameter of the dispersed bubbles is approximated by

$$d_{\text{DB}} = 2\,\text{La} = 2\sqrt{\frac{\sigma}{g\Delta\rho}}\,, \tag{659}$$

where La is the the Laplace coefficient.

This value is limited to be in the range

$$10^{-4}\,\text{m} \leq d_{\text{DB}} \leq 0.9D_h\,, \tag{660}$$

where D_h is the hydraulic diameter of the channel.

For the heat transfer between the liquid and bubble interface, in both evaporation and condensation, the heat transfer coefficient is

$$h_{conv,liq,\text{DB}} = \frac{k_{liq}}{d_{\text{DB}}}\text{Nu}_{\text{DB}}\,, \tag{661}$$

where the Nusselt number is given by

$$\text{Nu}_{\text{DB}} = 2.0 + 0.6\text{Re}_{\text{DB}}^{1/2}\text{Pr}_{liq}^{1/3} \tag{662}$$

and Pr_{liq} is the Prandtl number of the liquid phase.

The bubble Reynolds number, Re_{DB}, defined as a function of the dispersed bubble relative velocity, is

$$\text{Re}_{\text{DB}} = \frac{\rho_{liq}u_{r,\text{DB}}d_{\text{DB}}}{\mu_{liq}}\,. \tag{663}$$

The relative velocity between the dispersed bubble and liquid phase is limited to be smaller than its terminal velocity:

$$u_{r,\text{DB}} = \min\left(|u_{vap} - u_{liq}|, u_{\text{DB, term}}\right) . \tag{664}$$

The terminal velocity can be calculated as

$$u_{\text{DB, term}} = u_{r,\infty}(1 - \alpha_{vap})^{1.39} , \tag{665}$$

where $u_{r,\infty}$ is, for a single distorted particle,

$$u_{r,\infty} = \sqrt{2}\left(\frac{\sigma g \Delta\rho}{\rho_{liq}^2}\right)^{1/4} . \tag{666}$$

The heat transfer between the vapor and bubble interface is simplified by using a constant value,

$$h_{conv,vap} = 1000\,\mathrm{W/m^2K} . \tag{667}$$

7.3.1.2 Cap Bubble/Slug Flow For the cap bubble/slug flow regime, defined by the boundary $\alpha_{vap,DB} \leq \alpha \leq 0.5$, the interfacial heat transfer consists of contributions of both small dispersed bubbles and of large bubbles:

$$(h_{conv,liq}A_{int})_{\text{CS}} = (h_{conv,liq}A_{int})_{\text{DB}} + (h_{conv,liq}A_{int})_{\text{LB}} , \tag{668}$$

where subscripts CS and LB denote cap/slug flow and large bubble, respectively. For the small dispersed bubbles, the heat transfer coefficient between the liquid and the bubble interface is calculated the same way as in the bubbly flow regime. However, in the cap bubble/slug flow regime, the interfacial area for the small dispersed bubbles are calculated differently:

$$A_{int,\text{DB}} = \frac{6\alpha_{vap,\text{DB}}}{d_{\text{DB}}}\left(\frac{1 - \alpha_{vap}}{1 - \alpha_{vap,\text{DB}}}\right) . \tag{669}$$

The interfacial area associated with the large bubbles is computed as

$$A_{int,\text{LB}} = \frac{C^*}{D^*}\left(\frac{\alpha_{vap} - \alpha_{vap,\text{DB}}}{1 - \alpha_{vap,\text{DB}}}\right) . \tag{670}$$

Here the coefficient C^* and the diameter D^* depend on the channel hydraulic diameters:

$$C^* = \begin{cases} 4.5; & D_h < D_{h,crit} \\ 16; & D_h \geq D_{h,crit} \end{cases} , \tag{671}$$

and

$$D^* = \begin{cases} D_h; & D_h < D_{h,crit} \\ D_{h,crit}; & D_h \geq D_{h,crit} \end{cases} , \tag{672}$$

where $D_{h,crit} = 50\text{La}$, defined in Equation (632).

For these large bubbles, the heat transfer coefficient between the liquid and the bubble interface is calculated as

$$h_{conv,liq,\text{LB}} = \frac{k_{liq}}{D^*}\text{Nu}_{\text{LB}} , \tag{673}$$

where the Nusselt number, Nu_{LB}, is calculated using Equation (662), with a modified bubble Reynolds number that is defined as

$$\text{Re}_{\text{LB}} = \frac{\rho_{liq} u_{r,\text{LB}} D^*}{\mu_{liq}} . \tag{674}$$

The relative velocity of the large bubbles is defined as

$$u_{r,\text{LB}} = \min\left(|u_{vap} - u_{liq}|, u_{\text{LB, term}}\right) , \tag{675}$$

and the terminal velocity for the large bubbles are calculated as

$$u_{\text{LB, term}} = \begin{cases} u_{r,\infty} & d^*_{\text{LB}} < 0.125 \\ 1.13 u_{r,\infty} e^{-d^*_{LB}} & 0.125 \leq d^*_{\text{LB}} < 0.6 \\ \frac{0.496 u_{r,\infty}}{\sqrt{d^*_{\text{LB}}}} & d^*_{\text{LB}} \geq 0.6 \end{cases} . \tag{676}$$

Here the non-dimensional bubble diameter is defined as

$$d^*_{\text{LB}} = \frac{D^*}{D_h} , \tag{677}$$

and a relative velocity in an infinite medium is given by

$$u_{r,\infty} = \frac{\sqrt{2}}{2}\frac{\sqrt{g\Delta\rho D^*}}{\rho_{liq}} . \tag{678}$$

For the cap bubble/slug flow regime, the heat transfer between the vapor and the bubble interface are calculated similarly to that from the liquid phase to the bubble interface:

$$(h_{conv,vap} A_{int}) = h_{conv,vap}(A_{int,\text{DB}} + A_{int,\text{LB}}) , \tag{679}$$

and the same constant value of $1000\,\text{W}/\text{m}^2\text{K}$ is used for $h_{conv,vap}$.

7.3.1.3 Correction for Subcooled Boiling When subooled boiling is present, TRACE [2] recommends a correction factor for the volumetric heat transfer coefficient of the liquid phase for the dispersed bubble part. The subcooled boiling condition is determined by the following two conditions:

$$\Gamma^{wall}_{liq\rightarrow vap} > 0\,, \tag{680}$$

and

$$T_{liq} < T_{sat}\,. \tag{681}$$

Here, T_{sat} is set to be T_{int}. Under subcooled boiling condition, the volumetric heat transfer coefficient of the liquid phase for the dispersed bubble part is given by

$$(h_{conv,liq}A_{int})_{\text{DB}} = (1 - wf_{SB})(h_{conv,liq}A_{int})_{\text{DB}} + wf_{SB}(h_{conv,liq}A_{int})_{\text{SB}}\,, \tag{682}$$

in which $(h_{li}A_{int})_{\text{SB}}$ is modeled as

$$(h_{conv,liq}A_{int})_{\text{SB}} = 0.075 h_{fg}\frac{\rho_{liq}\rho_{vap}}{\Delta\rho}\max(10^{-4}, \alpha_{vap})\,. \tag{683}$$

The subcooled boiling weighting factor is defined as

$$wf_{\text{SB}} = \max\left\{0, \min[1, 10(0.2 - \alpha_{vap})]\right\}\,. \tag{684}$$

Note that such a correction factor is not applied to the vapor phase.

7.3.1.4 Annular/Mist Flow For the annular/mist flow regime, the interfacial heat transfer consists of the sum of two components, corresponding to the annular liquid film and entrained liquid droplets,

$$(h_{conv,liq}A_{int})_{\text{AM}} = (h_{conv,liq}A_{int})_{\text{film}} + (h_{conv,liq}A_{int})_{\text{drop}}\,. \tag{685}$$

The film thickness is computed as

$$\delta = \frac{D_h}{2}\left(1 - \sqrt{\alpha_{vap}}\right)\,, \tag{686}$$

but this value is limited to be greater than 10 μm. The interfacial area associated with the liquid film is calculated as a function of the volume fraction of the vapor phase and hydraulic diameter:

$$A_{int} = \frac{4}{D_h}\sqrt{\alpha_{vap}}\,. \tag{687}$$

For a special case where the surface is considered to be partially wetted by the liquid film, the interfacial area is modified to consider the partially wetted condition, such that

$$A_{int} = \frac{4}{D_h}\sqrt{\alpha_{vap}} f_{\text{wet}} \,, \tag{688}$$

where the fraction of the surface wetted by the liquid film, f_{wet}, is estimated by

$$f_{\text{wet}} = \frac{(1-\alpha_{vap})D_h}{4(25\times 10^{-6})} \,. \tag{689}$$

The surface is considered to be partially wetted when the liquid film thickness reaches the minimum critical value, 25 μm.

The liquid-film-to-interface heat transfer, $h_{conv,liq,\text{film}}$, is modeled by using a power-law weighting of the turbulent and laminar regimes:

$$h_{conv,liq,\text{film}} = \left(h^2_{conv,liq,\text{film,lam}} + h^2_{conv,liq,\text{film,turb}}\right)^{1/2} \,. \tag{690}$$

The correlation for the laminar regime is based on the Kuhn-Schroch-Peterson correlation:

$$\text{Nu}_{conv,liq,\text{film,lam}} = 2(1 + 1.83\times 10^{-4}\text{Re}_f) \,, \tag{691}$$

where the film Reynolds number is calculated as

$$\text{Re}_f = \frac{G_{liq}D_h}{\mu_{liq}} \,, \tag{692}$$

and G_{liq} is the liquid phase mass flux. For the turbulent regime, the Gnielinski correlation is used with a multiplier:

$$\text{Nu}_{conv,liq,\text{film,turb}} = 0.7\text{Nu}_{Gnielinski} \,. \tag{693}$$

The Gnielinski correlation will be discussed in the wall heat transfer section. For both the laminar and turbulent regimes, the heat transfer coefficient is related to the Nusselt number using the film thickness as reference length scale:

$$\text{Nu}_{conv,liq,\text{film,lam/turb}} = \frac{h_{conv,liq,\text{film,lam/turb}}\delta}{k_{liq}} \,. \tag{694}$$

For the liquid film part, the vapor-core-to-liquid-film-interface heat transfer coefficient is modeled as

$$h_{conv,vap,\text{film}} = \frac{k_{vap}}{D_c}\text{Nu}_{conv,vap} \,, \tag{695}$$

where D_c is the diameter of the annular vapor core approximated by

$$D_c \approx \sqrt{\alpha_{vap}} D_h \,. \tag{696}$$

The vapor-interface Nusselt number is modeled by the Dittus-Boelter correlation, and is limited to be larger than 4:

$$\mathrm{Nu}_{conv,vap} = \max\{4, 0.23\mathrm{Re}_c^{0.8}\mathrm{Pr}_{vap}^{0.4}\}\,, \tag{697}$$

where the Reynolds number for the annular vapor core is calculated as

$$\mathrm{Re}_c = \frac{G_{vap} D_c}{\mu_{vap}}\,, \tag{698}$$

and G_{vap} is the mass flux of the vapor phase, and Pr_{vap} is the Prandtl number of the vapor phase.

For the droplets, the volumetric interfacial area density is modeled as

$$A_{int,\text{drop}} = \frac{6\alpha_{vap}\alpha_d}{(1-\alpha_d)d_d}\,, \tag{699}$$

for which α_d is the fraction of the annular core region occupied by the droplets, and d_d is the droplet size. Both of these two quantities have been discussed in the interfacial drag model section, in Equations (611) and (613), respectively.

The liquid-to-interface heat transfer coefficient for the droplets is modeled as

$$h_{conv,liq,\text{drop}} = 2\pi^2 \frac{k_{liq}}{d_d}\,. \tag{700}$$

The vapor-to-interface heat transfer coefficient for the droplets part is modeled as

$$h_{conv,vap,\text{drop}} = \frac{\mathrm{Nu}_{conv,vap,\text{drop}} d_d}{k_{vap}}\,, \tag{701}$$

where $\mathrm{Nu}_{conv,vap,\text{drop}}$ is the Nusselt number, modeled as

$$\mathrm{Nu}_{conv,vap,\text{drop}} = 2 + \sqrt{u^*_{max}\text{Pé}}\,, \tag{702}$$

where u^*_{max} is the maximum dimensionless circulation velocity at the surface of the drop. Here Pé is the droplet Peclet number, defined by

$$\text{Pé} \equiv \frac{\rho_{vap} c_{p,vap} d_d u_r}{k_{vap}}\,, \tag{703}$$

where u_r is the drop relative velocity, defined as

$$u_r = 2.462\sqrt{\frac{g\Delta\rho d_d}{2\rho_{vap}}}\,. \tag{704}$$

The maximum dimensionless circulation velocity at the surface of the droplet, u^*_{max}, is defined as

$$u^*_{max} = \frac{1.5}{1 + \frac{2.8(1+2\lambda)(2+3\kappa)}{(2+3\lambda)\sqrt{\mathrm{Re}_d}}}\,, \tag{705}$$

where

$$\mathrm{Re}_d = \frac{\rho_{vap} u_r d_d}{\mu_{vap}}\,, \tag{706}$$

$$\lambda = \sqrt{\frac{\rho_{liq}\mu_{liq}}{\rho_{vap}\mu_{vap}}}\,, \tag{707}$$

and

$$\kappa = \frac{\mu_{liq}}{\mu_{vap}}\,. \tag{708}$$

The drop Reynolds number, Re_d, is limited to be in the range $0.5 \leq \mathrm{Re}_d \leq 200$. The maximum dimensionless circulation velocity at the surface of the drop, u^*_{max}, is limited to be in the range $0.0001 \leq u^*_{max} \leq 1.0$.

7.3.1.5 Transition between Bubbly/Slug and Annular/Mist Flow For the transition region, i.e., $0.5 \leq \alpha_{vap} \leq 0.75$, a simple linear interpolation is used to determine the interfacial heat transfer coefficients:

$$(h_{conv,k}A_{int}) = wf_{AM}(h_{conv,k}A_{int})_{\text{AM}} + (1 - wf_{\text{AM}})(h_{conv,k}A_{int})_{\text{BS}}\,, \tag{709}$$

where the weighting factor is defined by

$$wf_{AM} = \frac{\alpha_{vap} - 0.5}{0.75 - 0.5}\,. \tag{710}$$

The value of $(h_{conv,k}A_{int})_{\text{AM}}$ is calculated using the annular/mist flow regime model. The value of $(h_{conv,k}A_{int})_{\text{BS}}$ is calculated using either the dispersed bubble flow regime model or the cap/slug model, depending on the two-phase flow mass flux (see Figure 8).

7.3.1.6 Horizontal Stratified Flow When the flow is considered to be horizontal stratified flow, the volumetric interfacial heat transfer coefficients are calculated as

$$(h_{conv,k}A_{int})_{\text{strat}} = h_{conv,k,\text{strat}}A_{int,\text{strat}} \,. \tag{711}$$

The interfacial area density, $A_{int,strat}$, is obtained as

$$A_{int,\text{strat}} = \frac{S_{int}}{A} \,, \tag{712}$$

with S_{int} being the width of the stratified two-phase interface; see Equation (557) and Figure 6. The liquid side interfacial heat transfer coefficient is modeled as

$$h_{conv,liq,\text{strat}} = \frac{k_{liq}}{h_{liq}}\text{Nu}_{conv,liq} \,, \tag{713}$$

where h_{liq} is the liquid level, see Equation (559) and Figure 6. The Nusselt number, $\text{Nu}_{conv,liq}$, for the liquid phase interfacial heat transfer coefficient is calculated the same way as it is calculated for the annular/mist flow regime, namely, Equations (690), (691), and (693). The liquid phase Reynolds number used in these equations is the same as defined in Equation (552). Following TRACE [2], the final value of $h_{conv,liq,\text{strat}}$ is limited to be smaller than 2.5×10^4.

For the vapor phase, the volumetric interfacial heat transfer coefficient is simply calculated as

$$(h_{conv,vap}A_{int})_{\text{strat}} = 1000A_{int,\text{strat}} \,. \tag{714}$$

7.3.1.7 Transition between Stratified and Non-Stratified Flow If the flow regime is considered to be in a transition between stratified and non-stratified condition, a power law based interpolation is used:

$$(h_{conv,k}A_{int}) = (h_{conv,k}A_{int})^{n}_{\text{strat}}(h_{conv,k}A_{int})^{(1-n)}_{\text{non-strat}} \,, \tag{715}$$

where subscripts 'strat' and 'non-strat' represent for stratified flow and non-stratified flow conditions, respectively. The exponent, n, is set equal to wf_{strat}, given by Equation (565).

7.3.2 Post-CHF Flow Regimes

For post-CHF flow regimes, interfacial heat transfer coefficients must be defined in the following flow regimes: inverted annular, inverted slug, dispersed flow regimes, and transition regimes between them. The flow regime map for post-CHF interfacial heat transfer is shown in Figure 7.

7.3.2.1 Inverted Annular Flow Under post-CHF condition, if the void fraction is smaller than 0.6, the flow is treated as inverted annular flow. Assuming that the liquid core takes a cylindrical shape, its diameter can be obtained as

$$D_c = \sqrt{1-\alpha_{vap}} D_h \,, \tag{716}$$

and the interfacial area can be obtained as

$$A_{int} = \sqrt{1-\alpha_{vap}} \frac{4}{D_h} \,. \tag{717}$$

If the liquid core is in subcooled condition, following TRACE [2], the liquid phase interfacial heat transfer coefficient is simply calculated as

$$h_{conv,liq} = \frac{\mathrm{Nu}_{conv,liq} k_{liq}}{D_c} \,, \tag{718}$$

with $\mathrm{Nu}_{conv,liq}$ = 100. If the liquid core is in superheated condition, a correction factor for superheating condition is used, and the liquid phase interfacial heat transfer coefficient is corrected as

$$h_{conv,liq} = h_{conv,liq} \left[1 + \Delta T_{sup}(250 + 50\Delta T_{sup})\right] \,, \tag{719}$$

where ΔT_{sup} is the liquid superheat.

For the vapor phase interfacial heat transfer coefficient, the vapor film thickness, δ, must be determined first from Equation (629) for tube geometry, and Equation (630) for rod bundle geometry. Following TRACE [111], the vapor phase interfacial heat transfer coefficient is then calculated as

$$h_{conv,vap} = 2\frac{k_{vap}}{\delta} \,. \tag{720}$$

7.3.2.2 Inverted Slug and Dispersed Flow Under post-CHF conditions, if the void fraction is larger than 0.9, the flow is treated as inverted slug flow, dispersed flow, or a transition regime between them, depending on the liquid entrainment fraction. Thus, the interfacial heat transfer coefficients in these regimes are modeled as

$$(h_{conv,k} A_{int}) = (1-E)(h_{conv,k} A_{int})_{\mathrm{IS}} + E(h_{conv,k} A_{int})_{\mathrm{DF}} \,, \tag{721}$$

where subscripts IS and DF denote inverted slug and dispersed flow conditions, respectively.

For dispersed flow condition, the interfacial area density is modeled as

$$A_{int} = \frac{6(1-\alpha_{vap})}{d_{\text{SM}}}\,, \tag{722}$$

where d_{SM} is the droplet Sauter mean diameter, see Equation (636). For the liquid phase interfacial heat transfer coefficient, the same correlation for annular/mist flow droplet filed is used, i.e., Equation (700) with d_d replaced by d_{SM}. Again, the liquid phase superheated condition needs to be considered, similar to the inverted annular flow condition. Putting this together, the liquid phase interfacial heat transfer coefficient becomes

$$h_{conv,liq} = 2\pi^2 \frac{k_{liq}}{d_{\text{SM}}} \left[1 + \Delta T_{sup}(250 + 50\Delta T_{sup})\right]\,. \tag{723}$$

The vapor phase interfacial heat transfer coefficient is modeled as

$$h_{conv,vap} = \frac{k_{vap}}{d_{\text{SM}}} \text{Nu}_d\,, \tag{724}$$

with Nu_d being the droplet Nussult number. Following TRACE [2], Nu_d is modeled as

$$\text{Nu}_d = \frac{2 + 0.57\text{Re}_d^{1/2}\text{Pr}_{vap}^{1/3}}{(1+B_f)^{0.7}}\,. \tag{725}$$

The droplet Reynolds number, Re_d, is defined as

$$\text{Re}_d = \frac{\rho_{vap} V_r d_{\text{SM}}}{\mu_{vap}}\,, \tag{726}$$

with

$$V_r = \min\left[|u_{vap} - u_{liq}|, V_\infty \alpha_{vap}^{1.4}\right]\,. \tag{727}$$

The single-particle terminal velocity, V_∞, is modeled as

$$V_\infty = 0.6 \frac{\sigma^{0.316}(g\Delta\rho)^{0.228}}{\rho_{vap}^{0.456}\mu_{vap}^{0.0879}}\,. \tag{728}$$

Following TRACE [2], the blowing factor is simply approximated as

$$B_f = \frac{h_{vap} - h_{vap,sat}}{h_{fg}}\,. \tag{729}$$

For inverted slug flow, the interfacial area density is calculated as

$$A_{int} = \frac{4.8104(1-\alpha_{vap})}{d_{\text{IS}}}\,, \tag{730}$$

where d_{IS} is the diameter of the ligament, with subscript IS denoting inverted slug flow condition. Following TRACE [2], d_{IS} is approximated as

$$d_{\text{IS}} = 0.6325 D_h\,. \tag{731}$$

The liquid phase interfacial heat transfer coefficient is modeled in a two-step manner. First, it is calculated the same way as it is done for the dispersed flow, i.e., Equation (723) with d_{SM} replaced by d_{IS}. Second, the coefficient is corrected with an enhancement factor:

$$h_{conv,liq} = h_{conv,liq,1} 4 \min\left[1, \left(\frac{d_{\text{IS}}}{d_{\text{RT}}}\right)^2\right]\,, \tag{732}$$

in which $h_{conv,liq,1}$ is the interfacial heat transfer coefficient calculated from the first step. Following TRACE [2], the Sauter mean diameter for a distribution having a maximum diameter that is stable against the Rayleigh-Taylor instability, d_{RT}, is modeled as

$$d_{\text{RT}} = \frac{3.52}{4}\sqrt{\frac{\sigma}{g\Delta\rho}}\,. \tag{733}$$

For the vapor phase interfacial heat transfer coefficient, the same model used in the dispersed flow regime is used, i.e., Equation (724). Note that in both Equations (724) and (726), d_{SM} is replaced by d_{IS}.

7.3.2.3 Interpolation Region For the interpolation region, which is defined between void fraction 0.6 and 0.9, linear interpolation is used to obtain interfacial heat transfer coefficients,

$$(h_{conv,k}A_{int}) = wf_{\text{IA}}(h_{conv,k}A_{int})_{\text{IA}} + (1-wf_{\text{IA}})(h_{conv,k}A_{int})_{\text{IS-DF}} \tag{734}$$

in which subscripts IA and IS-DF denote inverted annular and inverted slug-dispersed flow conditions, respectively. Recall that, $(h_{conv,k}A_{int})_{\text{IS-DF}}$ is calculated as entrainment fraction weighted function of inverted slug and dispersed flow interfacial heat transfer coefficients, i.e., Equation (721). The final expression for the interpolation region is

$$(h_{conv,k}A_{int}) = wf_{\text{IA}}(h_{conv,k}A_{int})_{\text{IA}} + (1-wf_{\text{IA}})\left[(1-E)(h_{conv,k}A_{int})_{\text{IS}} + E(h_{conv,k}A_{int})_{\text{DF}}\right]\,. \tag{735}$$

The weighting function, wf_{IA}, is modeled as

$$wf_{\text{IA}} = 2x_{\text{IA}} - x_{\text{IA}}^2 \,, \tag{736}$$

with

$$x_{\text{IA}} = \frac{0.9 - \alpha_{vap}}{0.9 - 0.6} \,. \tag{737}$$

7.4 Wall Heat Transfer

For the 7-equation two-phase flow model presented in Section 3.1, the wall heat transfer model is required both in the determination of the heat transferred from solid surfaces, and also the rate of vapor phase generation occurring near those surfaces. Currently, a simplified heat flux partitioning model is used to determine the mass transfer associated with wall boiling. Recall that,

$$Q_{wall,liq,boil} = f_{\text{boil}} h_{\text{wall},\ell} (T_{wall} - T_{liq}) \, \kappa \, P_{hf} \,, \tag{226}$$

$$Q_{wall,liq,conv} = (1 - f_{\text{boil}}) h_{\text{wall},\ell} (T_{wall} - T_{liq}) \, \kappa \, P_{hf} \,, \tag{225}$$

$$Q_{wall,vap} = h_{\text{wall},v} (T_{wall} - T_{vap}) (1 - \kappa) P_{hf} \,, \tag{223}$$

and the total wall heat flux balance is given as

$$Q_{wall,total} = Q_{wall,liq,boil} + Q_{wall,liq,conv} + Q_{wall,vap} \,. \tag{222}$$

The wall boiling mass transfer is then determined as

$$\Gamma_{liq \to vap}^{wall} = \frac{Q_{wall,liq,boil}}{\left[e_{vap,sat}(\bar{p}_{int}) + \frac{\bar{p}_{int}}{\rho_{vap,sat}(\bar{p}_{int})} \right] - h_{liq}(\rho_{liq}, e_{liq})} \,. \tag{738}$$

The main purpose of this section is to describe the phasic heat transfer coefficient $h_{\text{wall},k}$ for both the pre-CHF and post-CHF conditions. More details on wall heat flux partitionings (parameters β and κ) can be found in Section 3.1.

7.4.1 Pre-CHF Wall Heat Transfer

For pre-CHF wall heat transfer, the logic to determine which heat transfer mode is shown in Figure 9.

7.4.1.1 Single-Phase Liquid Flow For single-phase liquid flow condition, the wall heat transfer coefficient, $h_{w,liq}$, is taken to be the maximum value of laminar flow, turbulent flow, and natural circulation condition:

$$h_{\text{wall},\ell} = \max\left\{h_{\text{lam}}, h_{\text{turb}}, h_{\text{NC}}\right\} \,, \tag{739}$$

in which the subscripts lam, turb, and NC stand for laminar flow, turbulent flow, and natural circulation flow conditions, respectively. They are discussed for both the tube and rod bundle geometries.

7.4.1.1.1 Tube Geometry For tube geometry, the laminar flow heat transfer coefficient is calculated as

$$h_{\text{lam}} = \text{Nu}_{\text{lam}} \frac{k_{liq}}{D_h} \,, \tag{740}$$

where Nu_{lam} is set to be constant for all cases: $\text{Nu}_{\text{lam}} = 4.36$.

For tube geometry, following TRACE [2], the turbulent flow heat transfer coefficient is modeled with the Gnielinski correlation and corrected by a temperature factor. The Gnielinski correlation is given by

$$\text{Nu}_{Gnielinski} = \frac{(f/2)(\text{Re} - 1000)\,\text{Pr}}{1 + 12.7(f/2)^{1/2}\left(\text{Pr}^{2/3} - 1\right)} \,, \tag{741}$$

where the friction factor, f, is given as

$$f = \left[1.58\ln(\text{Re}) - 3.28\right]^{-2} \tag{742}$$

Under single-phase liquid flow condition, the corresponding Reynolds number is defined as

$$\text{Re}_{liq} = \frac{\rho_{liq} u D_h}{\mu_{liq}} \tag{743}$$

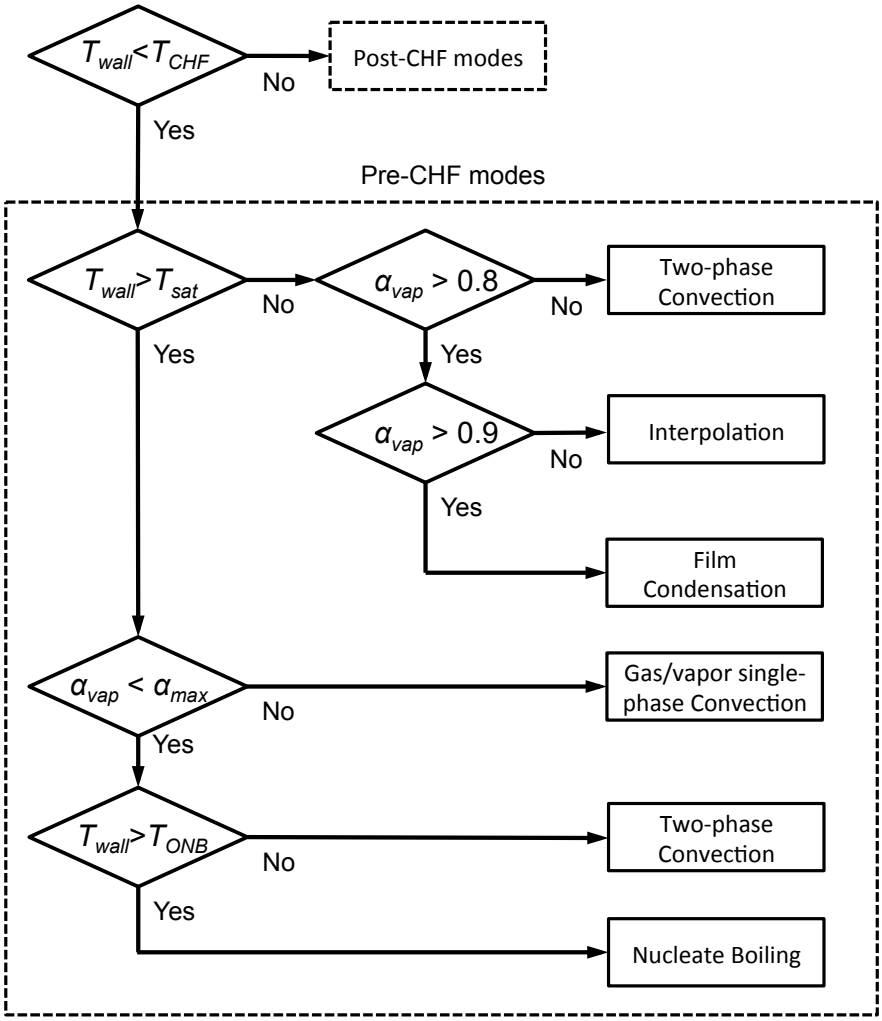

Figure 9. Logic to determine pre-CHF wall heat transfer mode following TRACE [2].

and is limited to be larger than 1000. The Nusselt number obtained from Equation (741) is then corrected with a correction factor as

$$\mathrm{Nu}_{\text{turb}} = \mathrm{Nu}_{Gnielinski} \left(\frac{\mathrm{Pr}_{liq}}{\mathrm{Pr}_w} \right)^{0.11} , \tag{744}$$

in which Pr_{liq} is the Prandtl number evaluated with liquid phase temperature, and Pr_w is the Prandtl number evaluated with wall temperature. The value of $\mathrm{Pr}_{liq}/\,\mathrm{Pr}_w$ is limited in the range of [0.05, 20].

For tube geometry, following TRACE [2], the natural circulation heat transfer coefficient is taken as the larger one between the laminar flow and turbulent flow conditions:

$$\mathrm{Nu}_{\text{NC}} = \max\left(\mathrm{Nu}_{\text{NC,lam}}, \mathrm{Nu}_{\text{NC,turb}}\right) , \tag{745}$$

with

$$\mathrm{Nu}_{\text{NC,lam}} = 0.59 \left(\mathrm{Gr}_{liq} \underset{liq}{\mathrm{Pr}} \right)^{1/4} , \tag{746}$$

and

$$\mathrm{Nu}_{\text{NC,turb}} = 0.13 \left(\mathrm{Gr}_{liq} \underset{liq}{\mathrm{Pr}} \right)^{1/3} . \tag{747}$$

In both equations, the liquid phase Grashof number is defined as

$$\mathrm{Gr}_{liq} = \frac{g\beta_{liq}\Delta T D_h^3}{\left(\mu_{liq}/\rho_{liq,film}\right)^2} , \tag{748}$$

where

$$\Delta T = \left| T_w - T_{liq} \right| , \tag{749}$$

and $\rho_{liq,\text{film}}$ is the liquid density evaluated from the film temperature:

$$T_{\text{film}} = 0.5(T_w + T_{liq}) . \tag{750}$$

7.4.1.1.2 Rod Bundle Geometry For rod bundle geometry, the laminar flow wall heat transfer coefficient is taken as the larger between the value by the El-Genk laminar formula and the value by the Kim & Li formula [2]:

$$h_{\text{lam}} = \max\left(h_{EG,\text{lam}}, h_{KL,\text{lam}}\right) , \tag{751}$$

where the subscripts EG and KL denote the El-Genk laminar formula and Kim & Li formula, respectively. The El-Genk laminar formula is given by

$$\mathrm{Nu}_{EG,\text{lam}} = A\mathrm{Re}^{B}\,\mathrm{Pr}^{0.33}\,, \tag{752}$$

where

$$\begin{aligned} A &= 2.97 - 1.76(P/D_r) \\ B &= 0.56(P/D_r) - 0.30 \end{aligned}\,. \tag{753}$$

Following TRACE [2], a curve-fitted formula is given to represent the Kim & Li tabular data:

$$\mathrm{Nu}_{KL,\text{lam}} = -5.6605\left(\frac{P}{D_r}\right)^2 + 31.061\left(\frac{P}{D_r}\right) - 24.473\,. \tag{754}$$

In both equations, P/D_r is the pitch-to-diameter ratio for the rod bundle geometry.

For rod bundle geometry, the turbulent flow wall heat transfer coefficient is modeled using the El-Genk turbulent formula, and then corrected by taking into consideration the wall temperature effect. The El-Genk turbulent correlation is given by

$$Nu_{EG,\text{turb}} = C_{EG}Re^{0.8}Pr^{0.33}\,, \tag{755}$$

where

$$C_{EG} = 0.028\left(\frac{P}{D_r}\right) - 0.006\,. \tag{756}$$

Again, P/D_r is the pitch-to-diameter ratio. The wall temperature effect is then considered, and the final form becomes

$$\mathrm{Nu}_{\text{turb}} = \mathrm{Nu}_{EG,\text{turb}}\left(\frac{\mathrm{Pr}_{liq}}{\mathrm{Pr}_w}\right)^{0.11}\,, \tag{757}$$

with $\mathrm{Pr}_{liq}\,/\,\mathrm{Pr}_w$ limited in the range of [0.05, 20].

For rod bundle geometry, the natural circulation wall heat transfer coefficient is modeled with the Sarma's formulation, as suggested by TRACE [2]

$$\mathrm{Nu}_{\text{NC}} = 0.7\left(\mathrm{Gr}_D\,\mathrm{Pr}_{liq}\right)^{1/4} \tag{758}$$

where the Grashof number is defined the same as Equation (748), and the same wall temperature effect is considered.

7.4.1.2 Two-Phase Forced Convection Following TRACE suggestion [2], with respect to single-phase forced convection flow condition, the primary role of two-phase forced convection enhancement is to correct the liquid Reynolds number. Thus, for two-phase forced convection flow condition, the same single-phase forced convection formulas are used, with a modified liquid phase Reynolds number:

$$\mathrm{Re}_{2\Phi} = \frac{\rho_{liq} u_{liq} D_h}{\mu_{liq}} \,. \tag{759}$$

Note that, under two-phase forced convection condition, the wall heat transfer coefficient is applied to the liquid phase only, and thus, $h_{\text{wall},v} = 0$.

7.4.1.3 Film Condensation For film condensation condition, the wall heat transfer is assumed to take place between the wall and the liquid phase, and

$$h_{\text{wall},\ell} = \frac{k_{liq}}{\delta} \mathrm{Nu}_{w,liq} \,, \tag{760}$$

in which δ is the film thickness, given by

$$\delta = \frac{D_h}{2}(1 - \sqrt{\alpha_{vap}}) \tag{761}$$

and is limited to be greater than 10 μm.

The Nusselt number, $\mathrm{Nu}_{w,liq}$, is computed using a power-law-based weighting function as

$$\mathrm{Nu}_{w,liq} = \left(\mathrm{Nu}_{\text{lam}}^2 + \mathrm{Nu}_{\text{turb}}^2\right)^{1/2} \,, \tag{762}$$

where the subscripts 'lam' and 'turb' denote laminar flow and turbulent flow conditions, respectively. Following TRACE suggestion [2], the Kuhn-Shrock-Peterson (K-S-P) correlation is used for the laminar flow condition:

$$\mathrm{Nu}_{\text{lam}} = 2\left(1 + 1.83 \times 10^{-4} \mathrm{Re}_f\right) \,, \tag{763}$$

with Re_f defined as

$$\mathrm{Re}_f = \frac{\alpha_{liq} \rho_{liq} u_{liq} D_h}{\mu_{liq}} \,. \tag{764}$$

For turbulent flow condition, the El-Genk correlation, Equation (755), is used, with a $1/4$ correction factor:

$$\mathrm{Nu}_{\text{turb}} = \frac{1}{4} \mathrm{Nu}_{EG,\text{turb}} \,. \tag{765}$$

The same liquid phase Reynolds number, Re_f, is used in the El-Genk correlation. Again, for vapor phase, the wall heat transfer coefficient is taken as zero: $h_{\text{wall},v} = 0$.

7.4.1.4 Transition between Two-Phase Forced Convection and Film Condensation

As shown in Figure 9, for void fractions between 0.8 and 0.9, a linear interpolation is used between the two-phase forced convection value and the film condensation value:

$$h_{\text{wall},\ell} = wf_{ann} h_{ann} + (1 - wf_{ann}) h_{2\Phi} \,, \tag{766}$$

in which h_{ann} is the film condensation wall heat transfer coefficient, and $h_{2\Phi}$ is the two-phase forced convection wall heat transfer coefficient. The weighting factor, wf_{ann} is defined as

$$wf_{ann} = \frac{\alpha_{vap} - 0.8}{0.9 - 0.8} \,. \tag{767}$$

7.4.1.5 Wall Boiling Heat Transfer

For pre-CHF condition, when wall temperature exceeds the onset-of-nucleate-boiling (ONB) temperature, T_{ONB}, the wall heat transfer mode becomes subcooled nucleate boiling or nucleate boiling, depending on the liquid phase temperature. In RELAP-7, the wall heat transfer for both modes are modeled the same.

7.4.1.5.1 Onset of Nucleate Boiling

The onset-of-nucleate-boiling temperature is discussed first. Following TRACE [2], the wall temperature for onset of nucleate boiling is modeled as

$$T_{\text{ONB}} = T_{liq} + \frac{1}{4} \left(\sqrt{\Delta T_{\text{ONB},sat}} + \sqrt{\Delta T_{\text{ONB},sat} + 4\Delta T_{sub}} \right)^2 \,, \tag{768}$$

in which ΔT_{sub} is the liquid phase subcooling temperature:

$$\Delta T_{sub} = T_{sat} - T_{liq} \,, \tag{769}$$

and $\Delta T_{\text{ONB},sat}$ is the wall superheat necessary for the onset of nucleate boiling when the liquid is at saturation temperature, defined as

$$\Delta T_{\text{ONB},sat} = \frac{2 h_{\text{FC}} \sigma T_{sat}}{F^2(\phi) \rho_{vap} h_{fg} k_{liq}} \,. \tag{770}$$

In this equation, h_{FC} is the two-phase flow forced-convection wall heat transfer coefficient; $F(\phi)$ is the contact angle correction factor:

$$F(\phi) = 1 - exp(-\phi^3 - 0.5\phi) \,. \tag{771}$$

Following TRACE [2], a constant value of 38 degrees is taken for the contact angle.

7.4.1.5.2 Nucleate Boiling Heat Transfer For both subcooled nucleate boiling and nucleate boiling conditions, the wall heat flux is modeled the same:

$$q''_{\text{NB}} = \left[(q''_{FC})^3 + (q''_{\text{PB}} - q''_{\text{BI}})^3 \right]^{1/3} , \tag{772}$$

in which q''_{FC} is the forced convection wall heat flux component:

$$q''_{\text{FC}} = h_{\text{FC}}(T_{wall} - T_{liq}) , \tag{773}$$

q''_{PB} is the pool boiling wall heat flux component based on wall temperature, and q''_{BI} is the pool boiling wall heat flux component at boiling initiation, based on T_{ONB}. Following TRACE [2], the Gorenflo correlation is used to model pool boiling wall heat flux:

$$q''_{\text{PB}} = \left[\frac{h_0 F_P}{(q''_0)^n} \right]^{\frac{1}{1-n}} (T_{wall} - T_{sat})^{\frac{1}{1-n}} , \tag{774}$$

with $h_0 = 5600\,\text{W/m}^2\text{K}$ and $q''_0 = 20000\,\text{W/m}^2$, and

$$n = 0.9 - 0.3P_r^{0.15} . \tag{775}$$

F_P is pressure dependent and is given as a function of the reduced pressure, P_r:

$$F_P = 1.73P_r^{0.27} + \left(6.1 + \frac{0.68}{1 - P_r} \right) P_r^2 , \tag{776}$$

with P_r defined as

$$P_r = \frac{P}{P_{crit}} , \tag{777}$$

and P_{crit} is the critical pressure.

7.4.2 Post-CHF Wall Heat Transfer

For post-CHF conditions, where wall temperature exceeds the CHF wall temperature, post-CHF wall heat transfer modes are used, which include inverted annular film boiling, dispersed flow boiling, a transition between these two boiling conditions, and a transition boiling mode. The logic of post-CHF wall heat transfer modes is shown in Figure 10.

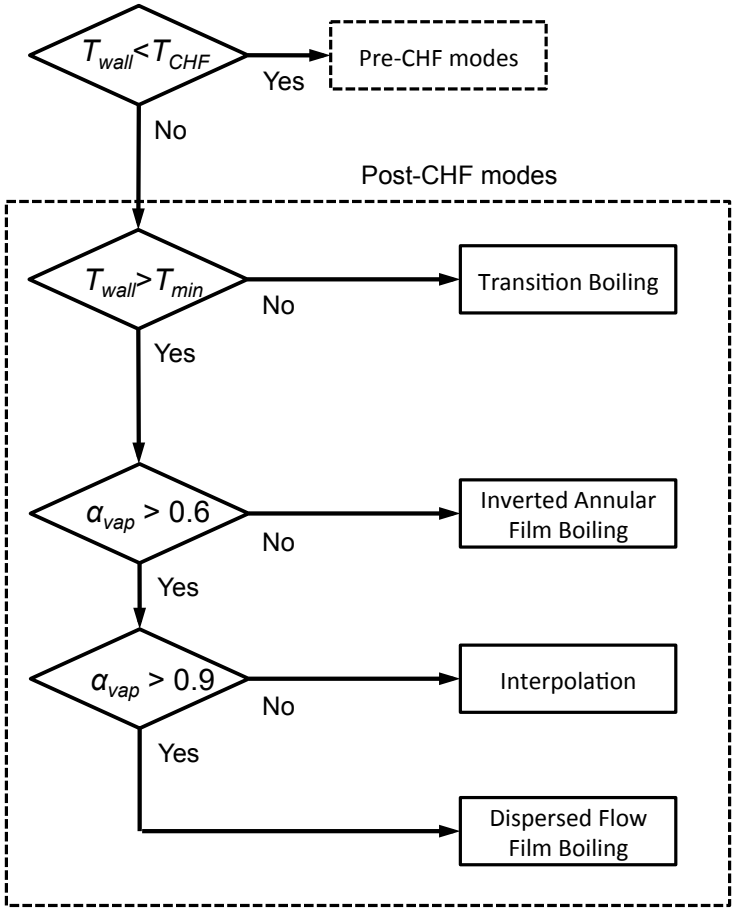

Figure 10. Logic to determine post-CHF wall heat transfer mode following TRACE [2].

7.4.2.1 Inverted Annular Film Boiling Under post-CHF condition, inverted annular film boiling is assumed to take place for void fractions smaller than 0.6. For inverted annular film boiling condition, following the TRACE discussion [2], the wall-to-liquid-phase heat transfer is approximated to have two components, one being convective heat transfer component, and the other one being radiative heat transfer component. The convective heat transfer component is simplified from the complex wall-to-vapor-phase and vapor-to-interface heat transfer relations. Following TRACE, it is modeled as

$$h_{wall,liq,\Gamma} = \frac{k_{vap}}{\delta}\mathrm{Nu}_{wall,liq}\frac{T_{wall}-T_{sat}}{T_{wall}-T_{liq}}\,, \tag{778}$$

in which δ is the vapor film thickness, already discussed in Section 7.2. For tube geometry, Equation (629) is used, and for rod bundle geometry, Equation (630) is used. For tube geometry, the Nusselt number is modeled as

$$\mathrm{Nu}_{wall,liq} = \max\left[0, 0.268(\delta^*)^{0.77} - 0.34\right]\,, \tag{779}$$

and for rod bundle geometry, this number is increased with a 30% enhancement:

$$\mathrm{Nu}_{wall,liq} = \max\left\{0, 1.3\left[0.268(\delta^*)^{0.77} - 0.34\right]\right\}\,. \tag{780}$$

The non-dimensional film thickness is given by

$$\delta^* = \delta\left(\frac{\rho_{vap}g\Delta\rho}{\mu_{vap}^2}\right)^{1/3}\,, \tag{781}$$

with $\Delta\rho = \rho_{liq} - \rho_{vap}$. Following TRACE, the wall-to-liquid radiation heat transfer coefficient is modeled as

$$h_{\text{wall},liq,rad} = \frac{q''_{\text{wall},liq,rad}}{T_{wall}-T_{liq}} = \frac{\sigma_{SB}(T_{wall}^2+T_{sat}^2)(T_{wall}+T_{sat})}{\frac{1}{\epsilon_{liq}\sqrt{1-\alpha_{vap}}}+\left(\frac{1}{\epsilon_{wall}}-1\right)}\,, \tag{782}$$

in which σ_{SB} is the Stefan-Boltzmann constant, $5.670\times10^{-8}\,\mathrm{Wm^{-2}K^{-4}}$, and ϵ_{liq} and ϵ_{wall} are the liquid and the wall emissivities, respectively. As suggested by the TRACE manual [2], by default, ϵ_{liq} is set to be 0.96, and ϵ_{wall} is set to be 0.7. Finally, the total wall to fluid heat transfer coefficient is

$$h_{\text{wall},\ell} = h_{\text{wall},liq,\Gamma} + h_{\text{wall},liq,rad}\,. \tag{783}$$

For wall-to-vapor heat transfer coefficient, following the TRACE manual, a simple model is used:

$$h_{\text{wall},\ell} = 2\frac{k_{vap}}{\delta}\,. \tag{784}$$

7.4.2.2 Dispersed Flow Film Boiling As in TRACE [2], under post-CHF conditions, dispersed flow film boiling is assumed to take place for void fraction greater than 0.9. For dispersed flow film boiling condition, heat transfer between the wall and the two phases is modeled in three components:

$$q''_{\text{wall}} = q''_{\text{wall},liq,rad} + q''_{\text{wall},vap,conv} + q''_{\text{wall},vap,rad} \,, \tag{785}$$

in which the three terms on the right hand side are the following, respectively: wall-to-liquid-phase radiation heat flux, wall-to-vapor-phase convective heat flux, and wall-to-vapor-phase radiation heat flux.

7.4.2.2.1 Wall-to-Vapor-Phase Convective Heat Transfer For dispersed flow film boiling, wall-to-vapor-phase convective heat transfer is modeled with the single-phase wall-to-vapor heat transfer with a two-phase enhancement factor:

$$q''_{\text{wall},vap,conv} = \frac{k_{vap}}{D_h} \text{Nu}_{wall,vap,conv} \Psi_{2\Phi} (T_{wall} - T_{vap}) \,, \tag{786}$$

in which $\text{Nu}_{wall,vap,conv}$ is the single-phase wall-to-vapor convective heat transfer Nusselt number, and $\Psi_{2\Phi}$ is the two-phase enhancement factor. The single-phase wall-to-vapor convective heat transfer is similar to that of single-phase liquid wall heat transfer with several modifications. Following TRACE [2], the Nusselt number is taken as the larger one between a laminar flow value and a turbulent flow value:

$$\text{Nu}_{wall,vap,conv} = \max(\text{Nu}_{\text{lam}}, \text{Nu}_{\text{turb}}) \,. \tag{787}$$

TRACE [2] suggests that a superposition method to be used for the laminar flow value:

$$\text{Nu}_{\text{lam}} = \left(\text{Nu}^3_{\text{lam,FC}} + \text{Nu}^3_{\text{lam,NC}} \right)^{1/3} \,, \tag{788}$$

in which the subscripts FC and NC stand for forced convection and natural convection, respectively.

For natural circulation condition, the laminar flow Nusselt number is modeled as

$$\text{Nu}_{\text{lam,NC}} = 0.13 \left[\frac{g \beta_{vap} (T_{wall} - T_{vap}) D_h^3}{\nu_{vap}^2} \text{Pr}_{vap} \right]^{1/3} \,. \tag{789}$$

For forced convection condition, the laminar flow Nusselt number is first calculated based on its geometry, and then corrected with a logarithmic ramp-up factor:

$$\text{Nu}_{\text{lam,FC}} = w f_{\text{FC}} \text{Nu}'_{\text{lam,FC}} \,, \tag{790}$$

in which wf_{FC} is the ramp-up factor, defined as

$$wf_{\text{FC}} = \frac{2 - \log_{10}(\text{Ri})}{2}\,, \tag{791}$$

with $\text{Ri} = \text{Gr}_D/\text{Re}_D^2$ being the Richardson number, and

$$\text{Gr}_D = \frac{g\beta_{vap}(T_{wall} - T_{vap})D_h^3}{\nu_{vap}^2}\,, \tag{792}$$

$$\text{Re}_D = \frac{\rho_{vap}u_{vap}D_h}{\mu_{vap}} \tag{793}$$

The Richardson number is limited to be in between 1 and 100. The geometry-based laminar flow Nusselt number, $\text{Nu}_{\text{lam,FC}}^{'}$, is given as a constant, 4.36, for tube geometry. For rod bundle geometry, the same correlations used in single-phase liquid flow conditions are used, i.e., Equation (751).

For turbulent flow of forced convection condition, the Nusselt number, Nu_{turb}, is calculated in a multi-step manner.

$$\text{Nu}_{\text{turb}} = \text{Nu}_{\text{turb}}^{'} f_{\text{wall}} f_{entr}\,, \tag{794}$$

in which $\text{Nu}_{\text{turb}}^{'}$ is the geometry-based turbulence flow Nusselt number. For tube geometry, the Gnielinski correlation, Equation (741), is used, and for rod bundle geometry, the El-Genk turbulence correlation, Equation (755), is used. The wall temperature correction factor, f_{wall}, is defined as

$$f_{\text{wall}} = \left(\frac{T_{wall}}{T_{vap}}\right)^n\,, \tag{795}$$

with

$$n = -\left[\log_{10}\left(\frac{T_{wall}}{T_{vap}}\right)\right]^{1/4} + 0.3\,. \tag{796}$$

For the condition that $T_{wall} < T_{vap}$, n is set to be -0.36. The entrance effect is taken into consideration with the entrance correction factor, f_{entr}, that is defined as

$$f_{entr} = 1 + \frac{2.4254}{(L/D_h)^{0.676}}\,, \tag{797}$$

in which L is the distance to the entrance, and the value of L/D_h is limited to be greater than 3.

Following TRACE [2], the two-phase enhancement factor, $\Psi_{2\Phi}$, is modeled as

$$\Psi_{2\Phi} = \left[1 + 25\frac{(1-\alpha_{vap})\mathrm{Gr}_{2\Phi}}{\mathrm{Re}_{vap}^2}\right]^{1/2}, \tag{798}$$

with

$$\mathrm{Gr}_{2\Phi} = \frac{\rho_{vap} g \Delta\rho D_h^3}{\mu_{vap}^2}, \tag{799}$$

with $\Delta\rho = \rho_{liq} - \rho_{vap}$, and

$$\mathrm{Re}_{vap} = \frac{\rho_{vap} u_{vap} D_h}{\mu_{vap}}. \tag{800}$$

The value of the two-phase enhancement factor is limited to be smaller than 5.

7.4.2.2.2 Wall-to-Liquid/Vapor-Phase Radiation Heat Transfer Following TRACE, the wall-to-liquid/vapor-phase radiation heat transfers are modeled as

$$q''_{\text{wall},liq,rad} = F_{wall,liq}\sigma_{SB}(T_{wall}^4 - T_{liq}^4), \tag{801}$$

and

$$q''_{\text{wall},vap,rad} = F_{wall,vap}\sigma_{SB}(T_{wall}^4 - T_{vap}^4), \tag{802}$$

where σ_{SB} is the Stefan-Boltzmann constant, and $F_{wall,liq}$ and $F_{wall,vap}$ are the gray body factors:

$$F_{wall,liq} = \frac{1}{R_2\left(1 + \frac{R_3}{R_1} + \frac{R_3}{R_2}\right)}, \tag{803}$$

and

$$F_{wall,vap} = \frac{1}{R_1\left(1 + \frac{R_3}{R_1} + \frac{R_3}{R_2}\right)}, \tag{804}$$

where

$$R_1 = \frac{1-\epsilon_{vap}}{\epsilon_{vap}(1-\epsilon_{vap}\epsilon_{liq})}, \tag{805}$$

$$R_2 = \frac{1-\epsilon_{liq}}{\epsilon_{liq}(1-\epsilon_{vap}\epsilon_{liq})}, \tag{806}$$

and

$$R_3 = \frac{1}{1-\epsilon_{vap}\epsilon_{liq}} + \frac{1-\epsilon_{wall}}{\epsilon_{wall}}. \tag{807}$$

For emissivity of the wall, ϵ_{wall}, the same constant value, 0.7, is used. For emissivities of the liquid phase and the vapor phase, the formula used in TRACE [111] is used:

$$\epsilon_{liq} = 1 - exp\left[-1.11\frac{\alpha_{liq}}{d_{drop}}L_{beam}\right] , \tag{808}$$

and

$$\epsilon_{vap} = \frac{\sum_{i=1}^{6}\eta_i}{1000\sigma_{SB}T_{vap}^4} , \tag{809}$$

with d_{drop} as the droplet size, given by Equation (636), and

$$\eta_i = c_1\frac{w_{avg,i}^3}{\exp(c_2 w_{avg}/T_{vap}) - 1}\cdot\{1 - \exp\left[-\min(100, k_i u)\right]\}\cdot dw_i . \tag{810}$$

The wave number difference, dw_i, and average wave number, w_{avg}, are given as

$$dw_i = w_{max,i} - w_{min,i} , \tag{811}$$

and

$$w_{avg} = \frac{w_{max,i} + w_{min,i}}{2} , \tag{812}$$

with

$$w_{min,1-6} = \{195.5, 1283.0, 3399.0, 5043.0, 6942.0, 8468.0\} , \tag{813}$$

and

$$w_{max,1-6} = \{804.5, 1892.0, 4008.0, 5652.0, 7551.0, 9077.0\} . \tag{814}$$

Parameter, k_i, is given as

$$k_i = \frac{300\alpha_i}{T_{vap}} , \tag{815}$$

in which α_i is the absorption coefficient, and

$$\alpha_{1-6} = \{0.0959, 0.2874, 0.2069, 0.0166, 0.0136, 0.00053\} . \tag{816}$$

The parameter u is given as

$$u = 9.869\times10^{-4} p_{vap} L_{beam} , \tag{817}$$

and the radiation path-length in steam, L_{beam}, is given as

$$L_{beam} = 0.9 D_h . \tag{818}$$

The two constants are given as $c_1 = 3.747\times10^{-5}$ and $c_2 = 1.4394$. Some limitations are applied to the values of the liquid phase and vapor phase emissivities. If the radiation path-length in steam is smaller than 10^{-6} m, the values of ϵ_{vap} and ϵ_{liq} are both set to be 0. If liquid phase volume fraction is smaller than 10^{-5}, the value of ϵ_{liq} is set to be 0.

7.4.2.2.3 Summary The following equations summarize the two-phase wall heat transfer coefficients for dispersed flow film boiling conditions:

$$h_{\text{wall},\ell} = \frac{q''_{\text{wall},liq,rad}}{T_{wall} - T_{liq}}\,, \tag{819}$$

and

$$h_{\text{wall},v} = \frac{q''_{\text{wall},vap,conv} + q''_{wall,vap,rad}}{T_{wall} - T_{vap}}\,, \tag{820}$$

with $q''_{\text{wall},liq,rad}$ given by Equation (801), $q''_{\text{wall},vap,conv}$ given by Equation (786), and $q''_{\text{wall},vap,rad}$ given by Equation (802).

7.4.2.3 Inverted Slug Film Boiling Under post-CHF condition, when the void fraction is between 0.6 and 0.9, wall boiling heat transfer mode is inverted slug film boiling, which is modeled as a transition region between the inverted annular film boiling and the dispersed flow boiling boiling regions. Following the TRACE manual [2], a linear interpolation is used for each phase k:

$$h_{\text{wall},k} = wf\, h_{\text{wall},k,\text{IAFB}} + (1 - wf) h_{\text{wall},k,\text{DFFB}}\,, \tag{821}$$

in which the subscripts IAFB and DFFB stand for inverted annular film boiling and dispersed flow film boiling conditions, respectively. The weighting factor is defined as

$$wf = x(2 - x)\,, \tag{822}$$

with

$$x = \frac{0.9 - \alpha_{vap}}{0.9 - 0.6}\,. \tag{823}$$

7.4.2.4 Transition Boiling Under post-CHF conditions, when the wall temperature is smaller than the minimum film boiling temperature, T_{min}, the wall boiling heat transfer mode is considered to be transition boiling. The minimum film boiling temperature, T_{min}, is given as a function of the liquid phase pressure:

$$T_{min,low}(p_{liq}) = T_{min,sat} - \frac{x \cdot 10^4}{2.82 + 1.22 \times 10^{-6} p_{liq}} \tag{824}$$

for pressure smaller than 9 MPa, and

$$T_{min,high}(p_{liq}) = \left[T_{min,low}(p_{liq} = 9 \times 10^6) - T_{sat}\right] \frac{p_{crit} - p_{liq}}{p_{crit} - 9 \times 10^6} + T_{sat} \tag{825}$$

for pressure greater than 9 MPa. The static flow quality is given by

$$x = \frac{\bar{h} - h_{liq,sat}}{h_{vap} - h_{liq,sat}} \,, \tag{826}$$

with

$$\bar{h} = \frac{\alpha_{liq}\rho_{liq}h_{liq} + \alpha_{vap}\rho_{vap}h_{vap}}{\alpha_{liq}\rho_{liq} + \alpha_{vap}\rho_{vap}} \,. \tag{827}$$

The Groeneveld-Stewart correlation for saturated water is given by

$$T_{min,sat} = 557.85 + 44.1(p_{liq} \cdot 10^{-6}) - 3.72(p_{liq} \cdot 10^{-6})^2 \,. \tag{828}$$

The wall-to-fluid heat transfer coefficients are then given as

$$h_{\text{wall},\ell} = (h_{\text{wall},\ell})_{\text{TB}} + (1 - wf_{\text{TB}})(h_{\text{wall},\ell})_{\text{FB}} \,, \tag{829}$$

and

$$h_{\text{wall},v} = (1 - wf_{\text{TB}})(h_{\text{wall},v})_{\text{FB}} \,, \tag{830}$$

with

$$(h_{\text{wall},\ell})_{\text{TB}} = \frac{wf_{\text{TB}}q''_{CHF}}{T_{wall} - T_{liq}} \,. \tag{831}$$

The two heat transfer coefficients, $(h_{\text{wall},\ell})_{\text{FB}}$ and $(h_{\text{wall},v})_{\text{FB}}$ are calculated from the film boiling heat transfer correlations. The weighting function, wf_{TB}, is modeled as

$$wf_{\text{TB}} = \sqrt{1 - \alpha_{vap}} \left(\frac{T_{wall} - T_{min}}{T_{CHF} - T_{min}} \right)^2 \,, \tag{832}$$

in which T_{min} is given by Equation (824) or (825), depending on pressure, and T_{CHF} is the wall temperature when CHF occurs. As in TRACE [2], an iterative method is implemented to solve for T_{CHF} from

$$q''_{\text{NB}}(T_{CHF}) = q''_{CHF} \,, \tag{833}$$

in which q''_{NB} is calculated from Equation (772), and q''_{CHF} is given by the 1995 AECL-IPPE CHF table as a function of pressure, mass flux, and flow quality:

$$q''_{CHF} = f(p, G, x) \,. \tag{834}$$

The details on the 1995 AECL-IPPE CHF table are not given in this document, but it can be found in the TRACE manual [2] or the original paper [112].

8 Heat Conduction Model

8.1 Heat Conduction Model

The heat conduction model calculates the temperature distributions in the solid components in the nuclear reactor system, such as the fuel, pipe walls, core barrel and core vessel, steam generator tubes, etc. It consists of a single, simplified energy balance equation, i.e., the transient heat conduction equation

$$\rho C_p \frac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) - q''' = 0 \tag{835}$$

where ρ, C_p, k are density, specific heat, and thermal diffusivity, respectively, of the solid materials. q''' is the volumetric heat source. Boundary conditions include three general types. The first type is the Dirichlet boundary condition, which provides a fixed boundary temperature

$$T_{\text{bc}} = T_0 \ . \tag{836}$$

The second type is the Neumann boundary condition, which provides a heat flux boundary condition

$$q''_{\text{bc}} = -k \frac{\partial T}{\partial \hat{n}_{\text{bc}}} = q''_0 \ . \tag{837}$$

The third type is the Robin boundary condition, which provides the convective heat transfer boundary condition

$$-k \frac{\partial T}{\partial \hat{n}_{\text{bc}}} = h_{\text{conv}} (T_{\text{coolant}} - T_{\text{bc}}) \ . \tag{838}$$

Both 1-D and 2-D solutions for the heat conduction model are available in RELAP-7.

8.2 Material Properties

Thermal properties, such as thermal conductivity k, material density ρ, and specific heat capacity C_p, for three materials are implemented in RELAP-7: uranium dioxide, the gas of the gap between the fuel rods and their cladding, and zircaloy. The implementation is consistent with values used in MATPRO [113] whenever possible. The constant room-temperature densities (ρ) are stored and are multiplied by temperature-dependent specific heat capacities (C_p) to generate the volumetric heat capacities. For all of the properties, constant values are assumed beyond the specified temperature ranges. Arbitrary low and

high values of 5 and 5000K are included to avoid problems with out-of-range material property data.

8.2.1 Uranium Dioxide

The reference density for uranium dioxide is $\rho = 10980$ kg/m^3. Its specific heat capacity information is provided in Section 2.2 of the MATPRO manual. Assuming that the material is pure UO_2 (with no PuO_2), and that the oxygen-to-metal ratio is 2.0,

$$\begin{aligned} C_p = & \frac{296.7 \times 535.285^2}{T^2 \left(\exp\left(\frac{535.285}{T}\right) - 1\right)^2} \exp\left(\frac{535.285}{T}\right) \\ & + 2.43 \times 10^{-2} T \\ & + \frac{2 \times 8.745 \times 10^7 \times 1.577 \times 10^5}{2 \times 8.3143 T^2} \exp\left(-\frac{1.577 \times 10^5}{8.3143 T}\right) . \end{aligned}$$

The uranium dioxide thermal conductivity data are taken from Section 2.3 of the MATPRO manual. The general equation for the thermal conductivity of solid fuel is

$$\begin{aligned} k = & \frac{D}{1 + T'(1 - D)} \frac{C_v}{(A + BT'')(1 + 3e_{th})} \\ & + 5.2997 \times 10^{-3} T \exp\left(\frac{-13358}{T}\right) \left[1 + 0.169 \left(\frac{13358}{T} + 2\right)^2\right] \end{aligned} \tag{839}$$

where k is thermal conductivity (W/m-K), D is the fraction of theoretical density (dimensionless); a value of 0.95 is currently assumed. A is a factor proportional to the point defect contribution to the phonon mean free path. Assuming an oxygen-to-metal ratio of 2.0, this factor is 0.339 m-s/kg-K. B is a factor proportional to the phonon-phonon scattering contribution to the phonon mean free path. Assuming no plutonium, this factor is 0.06867 m-s/kg-K. C_v is the phonon contribution to the specific heat at constant volume (J/kg-K). For pure UO_2, this is given by

$$C_v = \frac{296.7 \times 535.285^2}{T^2 \left[\exp\left(\frac{535.285}{T}\right) - 1\right]^2} \exp\left(\frac{535.285}{T}\right) \tag{840}$$

e_{th} is the linear strain term for temperatures above 300 K (dimensionless), which is given by

$$e_{th} = \frac{\Delta L}{L_0} = 1.0 \times 10^{-5} T - 3.0 \times 10^{-3} + 4.0 \times 10^{-2} \exp\left(\frac{-6.9 \times 10^{-20}}{1.38 \times 10^{-23} T}\right) \tag{841}$$

where T is fuel temperature (K). If the fuel temperature is less than 1364K, $T' = 6.5 - 0.00649T$. For temperatures greater than 1834 K, $T' = -1$. For values between these two, interpolation is employed (between these two temperatures).

8.2.2 Zircaloy

The reference density of zircaloy is 6551 kg/m^3. Its specific heat capacity is obtained by table look-up (see Table 4-2 in the MATPRO manual) with a temperature range of 300–1248 K. The zircaloy thermal conductivity is taken from Section 4.4 of the MATPRO manual. The equation used is

$$k = a_0 + a_1 T + a_2 T^2 + a_3 T^3 \tag{842}$$

for $300 < T < 2098$K, and $k = 36$ for $T \geq 2098K$. The remaining a_i parameters in (842) are given in Table 8.

Table 8. Zircaloy thermal conductivity parameters.

a_0	7.51
a_1	2.09×10^{-2}
a_2	-1.45×10^{-5}
a_3	7.67×10^{-9}

8.2.3 Fuel Rod Gap Gas

Representative gap gas properties are developed for a combination of fill and fission product gases. A 0.1066/0.1340/0.7594 mole fraction He/Kr/Xe mixture is modeled. A representative fuel rod internal pressure of 4.1 MPa is assumed to determine the gap gas density. Using the perfect gas relation and a temperature of 300 K yields $\rho = 183.06$ kg/m^3. Using the perfect gas relation, the specific heat capacity is determined to be $C_p = 186.65$ J/kg-K. From Section 12.1.1 of the MATPRO manual, the gas mixture thermal conductivity is given by

$$k_{\text{mix}} = \sum_{i=1}^{n} \left[\frac{k_i x_i}{x_i + \sum_{j=1}^{n} (1 - \delta_{ij}) \psi_{ij} x_i} \right] \tag{843}$$

where

$$\psi_{ij} = \phi_{ij}\left[1 + 2.41\frac{(M_i - M_j)(M_i - 0.142M_j)}{(M_i - M_j)^2}\right] \tag{844}$$

and

$$\phi_{ij} = \frac{\left[\left(1 + \frac{k_i}{k_j}\right)^{1/2}\left(\frac{M_i}{M_j}\right)^{1/4}\right]^2}{2^{\frac{2}{3}}\left(1 + \frac{M_i}{M_j}\right)^{1/2}} \tag{845}$$

and n is the number of components in mixture. M_i is the molecular weight of component i (kg), x_i is the mole fraction of component i, and k_i is the thermal conductivity of component i (W/m-K). The thermal conductivities of the three elements are given by $k_{He} = 2.639 \times 10^{-3}T^{0.7085}$, $k_{Kr} = 8.247 \times 10^{-5}T^{0.8363}$, $k_{Xe} = 4.351 \times 10^{-5}T^{0.8616}$. Using these equations, thermal conductivity values are provided, as a function of the mixture temperature, for temperatures from 300 to 3000K.

9 Component Models

The RELAP-7 code is an advanced system analysis tool based on components to represent the major physical processes in a reactor system. A real reactor system is very complex and contains hundreds of different physical components. It is impractical to resolve the real geometry of the entire system. Instead simplified thermal hydraulic models are used to represent (via "nodalization") the major physical components and describe the major physical processes (such as fluids flow and heat transfer). The main types of components developed in RELAP-7 are the following:

- one-dimensional (1-D) flow components,
- zero-dimensional (0-D) boundary components,
- zero-dimensional (0-D) junction components, and
- two-dimensional (2-D) heat structure components.

9.1 Pipe

Pipe is the most basic component in RELAP-7. It is a 1-D component which simulates thermal fluids flow in a pipe. Both a constant cross section area and a variable cross section area options are available for the Pipe component. The wall friction and heat transfer coefficients are either calculated through closure models or provided by user input. The pipe wall temperature can be provided as the wall heat transfer boundary condition. All the thermal fluid-dynamic models described in Chapters 2, 3, and 6 are available in the pipe component which includes the isothermal flow model, single-phase non-isothermal flow model, fully non-equilibrium 7-equation two-phase model, and the much simpler homogeneous equilibrium two-phase flow model.

9.2 Pipe Boundaries

Pipe or duct inlets and outlets, as well as pipe or duct closed ends are treated as zero-dimensional (0-D) components for setting boundary conditions. Chapter 5 describes pipe or duct boundary conditions in detail.

9.3 PipeWithHeatStructure

The PipeWithHeatStructure component simulates fluids flow in a 1-D pipe coupled with 1-D or 2-D heat conduction through the pipe wall. The adiabatic, Dirichlet, or convective boundary conditions at the outer surface of the pipe wall are available. Either a plate type or cylindrical type of heat structure can be selected. Volumetric heat source within the fluids or solid materials can be added.

9.4 CoreChannel

The CoreChannel component is a composite component designed to simulate the coolant flow and heat conduction inside a fuel rod as well as the conjugate heat transfer between the coolant and the fuel rod. In this component, the fuel rod is divided into the same number of segments as that of the coolant flow pipe elements. Each fuel rod segment is further simulated as 1-D or 2-D heat conduction model perpendicular to the fluid flow model. Both plate type fuel rod and cylindrical fuel rod type can be simulated. The

solid fuel part is able to deal with typical LWR fuel rod with complex clad/gap/fuel pellet geometries. The flow model and conjugate heat transfer model are fully coupled.

9.5 HeatTransferFromHeatStructure

This component couples together a flow channel and a heat structure via convective heat transfer exchange terms. Figure 11 gives an example where a cylindrical fuel rod heat structure (with fuel, gap, and cladding regions are present) is coupled to a 1-D flow channel. The vertical dimension is the dimension of the flow channel, which in this section will be called x, and the horizontal dimension of the heat structure will be referred to as y (in cylindrical coordinates, this is r).

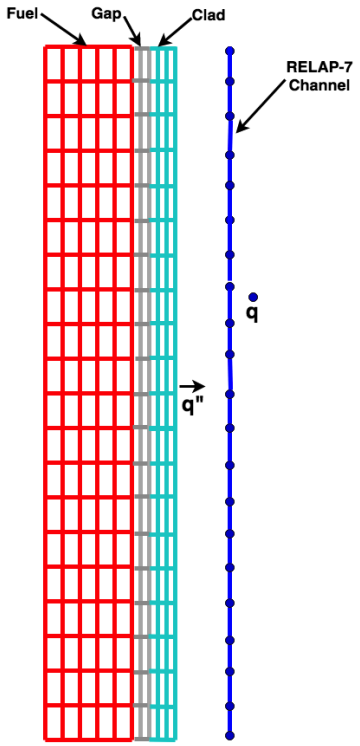

Figure 11. Heat transfer between 2-D heat structure and 1-D flow channel

On the heat transfer surface of the heat structure, the heat flux q from the heat structure, with surface temperature T_{wall}, to the flow channel on phase k, with temperature T_k, is

$$q(x) = \mathrm{h}_k(x)(T_{\text{wall}}(x) - T_k(x))\,, \tag{846}$$

where h_k is the heat transfer coefficient of phase k. Now suppose that the heat structure has a heated perimeter P_{hf} in contact with the flow channel. Then the following term appears on the right-hand side as a volumetric term of the energy equation:

$$\int_X \mathrm{h}_k(T_{\text{wall}} - T_k)P_{\text{hf}}\,\mathrm{d}X\,. \tag{847}$$

On the heat structure side, to ensure energy conservation, the term added to the RHS for each phase is the following, added:

$$\iint_A f_{\text{hf}}\mathrm{h}_k(T_k - T_{\text{wall}})\,\mathrm{d}A\,, \tag{848}$$

where f_{hf} is the ratio of the heated perimeter to the total perimeter:

$$f_{\text{hf}} \equiv \frac{P_{\text{hf}}}{P}\,. \tag{849}$$

For plate heat structure geometry, the perimeter P is simply the depth d. For cylindrical heat structure geometry, the perimeter is the circumference of the cylindrical surface in contact with the flow channel.

9.6 Junction

RELAP-7 uses a novel zero-volume, 0-D junction model designed for compressible flow in a component called `Junction`. Here, a brief description of the model is given; a full report is given in [114].

The compressible junction model employs a single strong constraint: conservation of mass at the junction:

$$\sum_{i\in\mathcal{I}_{\text{J}}} (\rho u A n)_i = 0\,, \tag{850}$$

where $\mathcal{I}_{\text{J}}$ is the set of node indices corresponding to nodes at the junction. This equation is associated with a new degree of freedom s_{J}, which represents the specific entropy at

the junction. Note that this new variable does not appear in this equation; it appears in the evaluation of the boundary fluxes for the nodes involved in the junction. As a side note, this new degree of freedom does not technically require an initial condition, as the old value is used nowhere except as the initial guess in iteration. However, most equations of state will fail if one tried to call equation of state functions with an arbitrary value of entropy such as zero, so this new variable should still be supplied with a physical guess for beginning iteration in the first time step. The simplest guess for the first time step is just to take the average of the initial entropies of the nodes connected at the junction:

$$s_{\text{J}}^{\text{guess}} = \frac{1}{n_{\mathcal{I}_{\text{J}}}} \sum_{i \in \mathcal{I}_{\text{J}}} s_i(t=0)\,. \tag{851}$$

After forming this guess before the first time step, s_{J} is solely updated via Newton updates, just like the other degrees of freedom.

Conservation of energy at the junction,

$$\sum_{i \in \mathcal{I}_{\text{J}}} (\rho u H A n)_i = 0\,, \tag{852}$$

is enforced *weakly* using the following definition for the junction specific total enthalpy:

$$H_{\text{J}} = \begin{cases} \frac{\sum_{i \in \mathcal{I}_{\text{J}}^{\text{in}}} (\rho u A H)_i}{\sum_{i \in \mathcal{I}_{\text{J}}^{\text{in}}} (\rho u A)_i} & \mathcal{I}_{\text{J}}^{\text{in}} \neq \varnothing \\ \frac{1}{n_{\mathcal{I}_{\text{J}}}} \sum_{i \in \mathcal{I}_{\text{J}}} H_i & \text{otherwise} \end{cases}, \tag{853}$$

where $\mathcal{I}_{\text{J}}^{\text{in}}$ is the subset of $\mathcal{I}_{\text{J}}$ corresponding to pipes that are *inlets* to the junction:

$$\mathcal{I}_{\text{J}}^{\text{in}} \equiv \{ i \in \mathcal{I}_{\text{J}} : u_i n_i > 0 \}\,. \tag{854}$$

The first branch of the conditional corresponds to the most common case, where there is at least one inlet to a junction. The second branch is a fail-safe, which can become necessary during iteration or when velocities are zero.

For a given junction, the contributions to the nonlinear residual for a node i, associated with one of the connected pipes, are the following:

$$r_i^{\text{mass}}(\mathbf{U}) = \rho_i^{\text{b}} u_i A_i n_i\,, \tag{855}$$

$$r_i^{\text{momentum}}(\mathbf{U}) = \left(\rho_i^{\text{b}} u_i^2 + p_i^{\text{b}}\right) A_i n_i\,, \tag{856}$$

$$r_i^{\text{energy}}(\mathbf{U}) = \rho_i^{\text{b}} u_i H_i^{\text{b}} A_i n_i\,. \tag{857}$$

The superscript "b" on density, pressure, and specific total stagnation enthalpy denotes a boundary quantity; this is a quantity may depend on more than just the corresponding interior solution value. Each of these may also depend on exterior data, namely, values from the junction. Since velocity always comes from the solution, it lacks the superscript.

The following table shows how ρ_i^{b}, p_i^{b}, and H_i^{b} are computed for the cases of inlet and outlet to the junction.

Junction Inlets		**Junction Outlets**	
$\boxed{\rho_i^{\text{b}} = \rho_i}$	(858)	$\boxed{H_i^{\text{b}} = H_{\text{J}}}$	(859)
$\tilde{h}_i = H_{\text{J}} - \frac{1}{2}u_i^2$	(860)	$p_{0,i}^{\text{b}} = p_{0,\text{J}} - K_i\left(p_{0,\text{J}} - p_i\right)$	(861)
$\boxed{p_i^{\text{b}} = p(\tilde{h}_i, s_{\text{J}})}$	(862)	$s_i^{\text{b}} = s(H_i^{\text{b}}, p_{0,i}^{\text{b}})$	(863)
$e_i^{\text{b}} = e(p_i^{\text{b}}, \rho_i^{\text{b}})$	(864)	$h_i^{\text{b}} = H_{\text{J}} - \frac{1}{2}u_i^2$	(865)
$E_i^{\text{b}} = e_i^{\text{b}} + \frac{1}{2}u_i^2$	(866)	$\boxed{p_i^{\text{b}} = p(h_i^{\text{b}}, s_i^{\text{b}})}$	(867)
$\boxed{H_i^{\text{b}} = E_i^{\text{b}} + \frac{p_i^{\text{b}}}{\rho_i^{\text{b}}}}$	(868)	$\boxed{\rho_i^{\text{b}} = \rho(p_i^{\text{b}}, s_i^{\text{b}})}$	(869)

9.7 VolumeJunction

The volume junction model is a 0-D component representing a joint/junction model with volume (inertia) effects considered. This model conserves the mass and energy among all connecting components. The governing equations of the mass and energy conservation for the `VolumeJunction` component are

$$\frac{d(\rho_{vb}V_{vb})}{dt} + \sum_{i=1}^{N}(\rho\boldsymbol{u})_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{870}$$

$$\frac{d((\rho e)_{vb}V_{vb})}{dt} + \sum_{i=1}^{N}((\rho e)_i + P_i)\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{871}$$

where ρ_{vb} and V_{vb} are the density and volume of the `VolumeJunction` component respectively. $(\rho u)_i$ is the mass flux at the connecting nodes. u_i is the fluid flow velocity at the connecting nodes. A_i is the flow area of the connecting components. P_i is the pressure at the connecting nodes. N is the number of connecting components. $(\rho e)_{vb}$ is the internal energy of the `VolumeJunction` component and $(\rho e)_i$ is the internal energy at the connecting nodes. The internal energy, instead of the total energy, is used in the energy equation since the energy changes due to the work of all the forces is difficult to capture in the `VolumeJunction` component and thus neglected (except the pressure). This assumption is valid for low speed flow applications.

The momentum conservation is more difficult to model in this 0-D component. A simplified model is used to account for various pressure losses in the `VolumeJunction` component.

$$P_i = P_{vb} + \Delta P_{acc} + s\Delta P_{form} + \Delta P_g \tag{872}$$

where the pressure loss due to acceleration is: $\Delta P_{acc} = \frac{1}{2}(\rho u^2)_{vb} - \frac{1}{2}(\rho u^2)_i$. The variable $s = 1$ if the fluids flow into the `VolumeJunction` component while $s = -1$ if the fluids flow out of the `VolumeJunction` component. The pressure loss due to the form loss is: $\Delta P_{form} = \frac{1}{2}K(\rho u^2)_i$. The pressure loss due to the gravity is $\Delta P_g = \rho_{vb}\Delta H$, and ΔH is the height difference between the elevation of the center of the `VolumeJunction` component and the elevation of the connecting components.

Note that the friction loss is neglected in this model. This is because the friction loss is dependent on the flow path, and it is very difficult (and non-physical) to model the friction loss in the 0-D component. On the other hand, the friction loss in a large volume is always very small. If the friction loss has to be considered, the form loss coefficient can be adjusted to account for it.

The above simplifications of modeling the momentum conservation works well as long as the pressure propagation is much faster than the fluid transport, which is true for incompressible flows and low speed compressible flows.

9.8 Pump

The simplified pump model is based on three assumptions:

- quasi-steady state,
- incompressible flow,

- and 100% pump efficiency.

Currently, the RELAP-7 pump designed as one 0-D junction component which provides:

- one BC for upstream pipe: pressure
- two BCs for downstream pipe: pressure and total energy.

Only one scalar variable –pump pressure p_J is defined as the unknown for the pump model, which uses the mass balance as the nonlinear equation:

$$(\rho u)_1 A_1 \hat{n}_1 + (\rho u)_2 A_2 \hat{n}_2 = 0\,, \tag{873}$$

where ρu is the momentum for the connecting pipes, A the cross-section area, and $\hat{n}$ is the direction normal ($\hat{n} = 1$ for the inlet and $\hat{n} = -1$ for the outlet). It is assuming that internal energy does not change through a pump, so

$$e_{out} = e_{in} = e_J\,. \tag{874}$$

Pressures at inlet and outlet are calculated with incompressible flow Bernoulli's equation. It is also assumed that the pump work is added to the fluid only in the entrance segment and the loss in the exit segment is negligible. For normal flow

$$p_1 = (p_J + \frac{1}{2}\rho_J u_J^2) - \rho_1 g H - \frac{1}{2}\rho_1 u_1^2 \tag{875}$$

$$p_2 = (p_J + \frac{1}{2}\rho_J u_J^2) - \frac{1}{2}\rho_2 u_2^2 \tag{876}$$

where

$$\rho_J = \rho(e_J, p_J) \tag{877}$$

$$u_J = \frac{\rho_1 u_1 A_1}{\rho_J A_J} \tag{878}$$

g is the gravity constant and H is the pump head. H can be set as an input parameter which can be changed through the control system to simulate dynamic process such as coast-down, or H can be calculated by coupling with a shaft work, i.e., provided by a turbine,

$$H = \frac{\dot{W}_t}{\rho_1 u_1 A_1 g} \tag{879}$$

where $\dot{W}_t$ is the turbine shaft power. Downstream total energy is calculated by

$$\rho E_{bc} = \rho_{bc} \left(e_J + \frac{1}{2} u_{bc}^2 \right) . \tag{880}$$

For reverse flow, the pump is treated as a resistance junction. The reverse form loss coefficients for inlet (K_1) and outlet (K_2) are given by the user. The pressures at inlet and outlet for reverse flow conditions are

$$p_1 = (p_J + \frac{1}{2}\rho_J u_J^2) - \frac{1}{2}(1 + K_1)\rho_1 u_1^2 \tag{881}$$

$$p_2 = (p_J + \frac{1}{2}\rho_J u_J^2) - \frac{1}{2}(1 - K_2)\rho_2 u_2^2 . \tag{882}$$

The pump can also be simulated as a time dependent junction with given mass flow rate as a function of time.

9.9 Turbine

A turbine is a device that converts energy contained in high-pressure and high-temperature fluid into mechanical work. The complicated configuration of a turbine precludes a complete first-principle model, at least for the purpose of system transient calculations. In RELAP5 [23], quasi-steady state mass, momentum, and energy conservation equations are used for flow across a turbine stage. However, several questionable assumptions, such as constant density across the turbine blade stage, are used to derive the momentum equation. For a complex curved flow path, it is almost impossible to derive an accurate 0-D momentum equation. The force between the junction solid wall and the fluid is unknown due to the lack of geometric definition in 0-D and no simple assumptions can be made. This is why Bernoulli's equation (or mechanical energy equation) is used instead for 0-D junction models in current reactor safety system codes such as RELAP5 [23], TRAC [115], and TRACE [2]. However, for compressible flow in a turbine, Bernoulli's equation for isentropic compressible flow is identical to the total energy conservation equation. Hence, the Bernoulli's equation cannot be used for momentum.

Lacking an equation for momentum, we instead use turbine characteristics curves for momentum, which is based on actual dynamical turbine performance data. Turbine characteristics curves reflect the complex relationships of the non-dimensional turbine mass flow rate and turbine efficiency with pressure ratio and the non-dimensional rotational speed. Fig. 12 shows one example of turbine characteristics curves [3]. In the figure, subscript 03

indicates the upstream stagnation condition, subscript 04 indicates the downstream stagnation condition and N is the rotational speed. Note that the curves dynamically capture the choking behavior. To further simplify the curves, a couple of assumptions are made: (1) Turbine thermal efficiency is constant, and (2) Non-dimensional mass flow rate is not a function of non-dimensional rotational speed (by noting that all the curves for different rotational speeds tend to collapse together). With these assumptions, an equation for rotational speed is not needed, and only one characteristics curve for mass flow rate is sufficient to establish the equation for momentum.

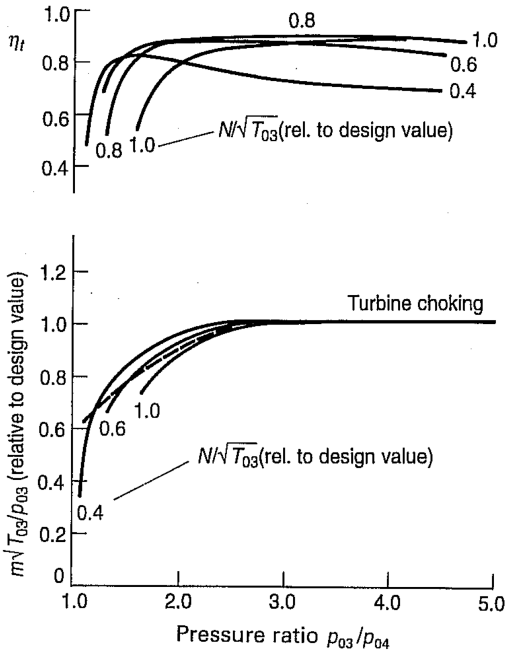

Figure 12. Turbine characteristics (credit of Saravanamuttoo, Rogers, and Cohen [3]).

Based upon the aforementioned discussion, we developed a new simple turbine component model as a junction without volume. Thermal inertia in the solid structures and fluid is ignored, similar to that in RELAP-5. Fig. 13 shows the T-s diagram for a thermodynamic process in a turbine. Point 1 represents inlet static condition and point 2 represents outlet static condition; point 2s is the end point for a reversible process; point 01, 02, and 02s represent the stagnation conditions corresponding to points 1, 2, and 02, respectively.

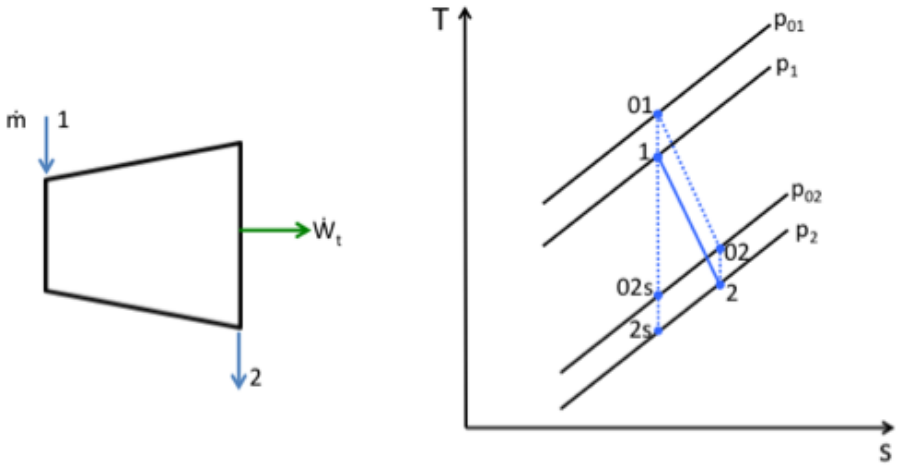

Figure 13. T-s diagram for a turbine.

Because a quasi-steady state turbine is a 0-D component which provides

- one BC for inlet pipe: p_1 (inlet pressure)
- two BCs for outlet pipe: p_2 (outlet pressure), ρ_2 (outlet density)
- turbine shaft power: $\dot{W}_t$

four equations are needed to close the system. The first one is mass conservation Eq. (873). The mass flow rate is calculated as $\dot{m} = \rho u A$.

As discussed before, turbine characteristics are used for the momentum equation. Assuming constant thermal efficiency and ignoring rotational speed effect, we have

$$\frac{\frac{\dot{m}\sqrt{T_{01}}}{p_{01}}}{\frac{\dot{m}_{max}\sqrt{T_{01r}}}{p_{01r}}} = f(p_{01}/p_{02})\,. \tag{883}$$

The subscript r denotes nominal design reference value and 0 denotes stagnation condition. $\dot{m}_{max}$ is the nominal maximum design mass flow rate through the turbine. The turbine characteristic curve $f(p_{01}/p_{02})$ should come from turbine vendors. According to

reference [116], the curve for a HP (High Pressure) steam turbine is defined as

$$f(p_{01}/p_{02}) = \sqrt{1 - \left(\frac{p_{02}}{p_{01}}\right)^2} \ . \tag{884}$$

This equation matches the real test data very well. However, this curve is not valid when the pressure ratio is equal or less than 1. Therefore, a similar smooth curve is used

$$f(p_{01}/p_{02}) = \tanh\left(\beta\left(\frac{p_{02}}{p_{01}} - 1\right)\right) \tag{885}$$

where β is a constant and is calculated by the following formula

$$\tanh\left(\beta\left(\frac{p_{01r}}{p_{012}} - 1\right)\right) = \frac{\dot{m}_r}{\dot{m}_{max}} \ . \tag{886}$$

Therefore, β is determined by the design pressure ratio and the ratio of nominal mass flow rate at design point with the maximal mass flow rate. The energy equation for turbine is

$$\eta = \frac{h_{01} - h_{02}}{h_{01} - h_{02s}} \tag{887}$$

where η is the turbine thermal efficiency, and h the enthalpy. Fig. 13 shows the location of the thermodynamic states on a T-s diagram.

Turbine shaft work is calculated by

$$\dot{W}_t = \dot{m}(h_{01} - h_{02}) \ . \tag{888}$$

Eqs. (873), (883) and (887) are used to solve for p_1, p_2, and ρ_2, and Eq. (888) is used to compute turbine power. To derive stagnation states, recognize that

$$h_0 = h + \frac{1}{2}u^2 \tag{889}$$

where u is the velocity. Then assuming an isentropic process, from the static state, (h_1, p_1), the stagnation state (h_{01}, p_{01}) may be found. For ideal gas, the following equations hold (pages 54 to 56, ref [3])

$$p_0 = p\left(1 + \frac{\gamma - 1}{\gamma}\frac{\rho u^2}{2p}\right)^{\gamma/(\gamma-1)} \tag{890}$$

$$T_0 = T\left(\frac{p_0}{p}\right)^{(\gamma-1)/\gamma} \tag{891}$$

where γ is the ratio of specific heats. h_1, u_1, and T_1 are obtained from turbine inlet pipe as coupled variables and p_1 is a scalar variable unknown. h_{01} is calculated according to Eq. (889). p_{01} is calculated according to Eq. (890). T_{01} is calculated according to Eq. (891).

The pressure p_2 and density ρ_2 are scalar variable unknowns. According to an EOS (equation of state) relationship, h_2 is evaluated

$$h_2 = h\left(p_2, \rho_2\right) \tag{892}$$

u_2 is obtained from turbine outlet pipe as a coupled variable. h_{02} is calculated according to Eq. (889). p_{02} is calculated according to Eq. (890). To derive h_{02s}, we need two thermodynamic states at 2s. Note $p_{2s} = p_2$. We can obtain the density at 2s by following the isentropic line from point 1 (see Fig. 13)

$$\frac{\rho_{2s}}{\rho_1} = \left(\frac{p_2}{p_1}\right)^{1/\gamma} . \tag{893}$$

According to the EOS relationship, h_{2s} is evaluated with p_2 and ρ_{2s}. Now h_{02s} can be calculated according to Eq. (889). When the stagnation pressure at the inlet is less than the stagnation pressure at the outlet, the turbine is treated as a closed valve. Major physical parameters for the turbine model include thermal efficiency, nominal mass flow rate, design pressure ratio, and design stagnation inlet temperature and pressure.

9.10 SeparatorDryer

Boiling Water Reactors (BWRs) use a steam separator to increase the quality of steam prior to generation of mechanical energy in the turbine. A steam separator component is based on the principle of centrifugal separation, where the liquid/gas phase separation occurs as a mixture of water and steam flows upward in a vortex motion within vertical separator tubes. Therefore, the outflows of the steam separator are a flow of steam from the top exit and a flow of liquid water from the discharge to the bulk water surrounding the separator barrel. Typically, the quality of the steam at the outlet of the separator is at least 90%. In addition, steam dryers are used to further increase the quality of steam to ensure that the steam is dry.

In RELAP-7 the separator dryer component is developed to model both the steam separators and moisture dryers together. Currently only an ideal separation model with perfect steam separation has been implemented into RELAP-7. The mechanistic separator

and dryer models will be implemented in the future. The steam SeparatorDryer component has one inlet and two outlets. Each connection has a form loss coefficient K, which generally accounts for pressure loss due to expansion/contraction, mixing, and friction.

The conservation equations of mass and energy for the SeparatorDryer model are the following:

$$V\frac{d\rho_{sd}}{dt} + \sum_{i=1}^{3}(\rho \boldsymbol{u})_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{894}$$

$$V\frac{d(\rho e)_{sd}}{dt} + \sum_{i=1}^{3}((\rho e)_i + P_i)\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{895}$$

where ρ_{sd} and $(\rho e)_{sd}$ are the density and internal energy of the SeparatorDryer component respectively. V is the volume of the SeparatorDryer component. $(\rho u)_i$ is the mass flux at the connecting nodes. u_i is the velocity at the connecting nodes. A_i is the flow area of the connecting component. $(\rho e)_i$ is the internal energy of the connecting nodes. P_i is the pressure at the connecting nodes.

An incomplete form of the momentum equation is used to account for the various pressure losses in the SeparatorDryer component:

$$P_i = P_{sd} + \Delta P_{acc} + s\Delta P_{form} + \Delta P_g \tag{896}$$

where $s = 1$ if fluids flow into SeparatorDryer and $s = -1$ if fluids flow out of the SeparatorDryer. P_{sd} is the reference pressure of the SeparatorDryer which is taken as the value in the center of SeparatorDryer. The pressure loss due to acceleration is: $\Delta P_{acc} = \frac{1}{2}(\rho u^2)_{sd} - \frac{1}{2}(\rho u^2)_i$. The pressure loss due to the form loss is: $\Delta P_{form} = \frac{1}{2}K(\rho u^2)_i$. The pressure loss due to the gravity is $\Delta P_g = \rho_{sd}\Delta H$, and ΔH is the height difference between the elevation of the connecting pipe and the center elevation of the SeparatorDryer component.

9.11 Valve

The current valve component developed in RELAP-7 is a simplified model to simulate the fundamental functions (i.e. open and close) of generic valves. The valve component is a junction type of components and it connects one pipe on each side. The valve is initiated

with a given user input (i.e., fully open or fully closed). It then starts to react (i.e., close or open) and is triggered either by a preset user given trigger time or by a trigger event, which requires the RAVEN code control logic. In its opening status, either fully open or partially open, it serves as a regular flow junction with form losses. In its fully closed status, the connected two pipes are physically isolated. The current valve model also includes the gradually open/close capability similar to a motor driven valve to simulate the physical behavior of a valve open/close procedure. It also has the benefit of avoiding spurious numerical oscillations that are caused by an instantaneous open/close procedure. Additional, specific valve components to be developed in the future (e.g., gate valve and check valve) are planned to enhance the RELAP-7 capabilities for engineering analysis.

9.12 CompressibleValve

The valve model introduced in the previous section is for low speed nearly incompressible flow cases. For reactor safety simulations, there are cases where high speed compressible flow models are needed. One such example is a safety/relief valve (SRV), which either is activated by passive setting points such as pressure (safety valve mode) or by active control actions through an electric motor or compressed air (relief mode). Normally, a SRV would discharge pure gas or steam. However, there are transients in a LWR that can involve the discharge of two-phase mixture or pure liquid through a SRV [117]. As an initial version of simplified SRV model, only steam/gas is considered. Since the SRV always has the minimal cross section area along the release line, it is assumed that choking always happens in the throat of the SRV. To further simplify the model, is is further assumed that choking will happen whenever the valve is open. Also, the steam/gas is currently treated as an ideal gas.

The Compressible Valve is designed as a single 0-D junction component which provides:

- one BC for upstream pipe: pressure (p_i)
- two BCs for downstream pipe: momentum $(\rho u)_o$ and total enthalpy (H_o).

Therefore three equations are needed to close the system. First consider the case when the valve is open. The p_i unknown will correspond to the mass conservation:

$$(\rho u)_1 A_1 \hat{n}_1 - (\rho u)_o A_2 \hat{n}_2 = 0 \tag{897}$$

where $(\rho u)_1$ is the coupled momentum for the connecting inlet pipe end, A the cross-section area, and $\hat{n}$ direction normal ($\hat{n}_1 = 1$ for the inlet and $\hat{n}_2 = -1$ for the outlet). The $(\rho u)_o$ unknown corresponds to the following equation for the choked condition

$$(\rho u)_o A_2 \hat{n}_2 - \dot{m}_c = 0 \tag{898}$$

where $\dot{m}_c$ is the critical mass flow rate calculated by the equation for isentropic ideal gas flow [118]

$$\dot{m}_c = A_t(\rho u)_c = A_t \left(\gamma p_c \rho_c\right)^{1/2} \tag{899}$$

where A_t is the cross-section area at the valve throat, which can be controlled by the valve action, i.e., from 0 to the fully open area. The critical pressure p_c and the critical density ρ_c are determined by

$$\frac{p_c}{p_{i0}} = \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} \tag{900}$$

$$\frac{\rho_c}{\rho_{i0}} = \left(\frac{2}{\gamma+1}\right)^{\frac{1}{\gamma-1}} \tag{901}$$

where the subscript $i0$ indicates the stagnation condition for the inlet. For non-ideal choked flow (not to be confused with non-ideal gas) through a valve, m_c can be modified by multiplying the valve coefficient C_v [118] which is defined as the ratio of real mass flow rate over the ideal mass flow rate. The valve coefficient model will be included in the near future.

For ideal gas and isentropic flow, the steady state mass flow rate is calculated as

$$\dot{m}_{sub} = A_2 \left\{ 2\left(\frac{\gamma}{\gamma-1}\right) p_{i0}\rho_{i0} \left(\frac{p_2}{p_{i0}}\right)^{\frac{2}{\gamma}} \left[1 - \left(\frac{p_2}{p_{0i}}\right)^{\frac{\gamma-1}{\gamma}}\right] \right\}^{\frac{1}{2}} . \tag{902}$$

By comparing the subsonic mass flow rate and the choking mass flow rate, we can determine whether choking happens. When $\dot{m}_{sub} \geq \dot{m}_c$, Eq. (898) is used for momentum; Otherwise subsonic flow momentum equation is used:

$$(\rho u)_o A_2 \hat{n}_2 - \dot{m}_{sub} = 0. \tag{903}$$

The H_o unknown will correspond to the energy conservation:

$$(\rho u)_1 A_1 \hat{n}_1 H_1 - (\rho u)_o A_2 \hat{n}_2 H_o = 0 \ . \tag{904}$$

When the valve is fully closed, the following equations are used for p_i, $(\rho u)_o$, and H_o, respectively

$$p_i - p_1 = 0 \tag{905}$$

$$(\rho u)_o - (\rho u)_2 = 0 \tag{906}$$

$$H_o - H_2 = 0\ . \tag{907}$$

p_1, $(\rho u)_2$, and H_2 are coupled variables from the connecting pipe ends. The pipe end BCs are treated as solid wall conditions when the valve is fully closed. Subsonic compressible flow model, valve coefficient model, stiffened gas model, and two-phase critical flow model will be included in a later version.

9.13 WetWell

The wet well refers to the suppression chamber of a BWR reactor, which is composed of water space and gas space. The 0-D wet well model simulates both spaces. Fig. 14 shows the schematic of the simplified model. Major assumptions include: (1) the suppression pool is well mixed; (2) the kinetic energy in both spaces is ignored, therefore the water space pressure follows a hydrostatic distribution; (3) no mass transfer between water and gas space; (4) gas space is filled with 100% nitrogen gas; (5) the geometry of the wet well is rectangular; and (6) no steam venting from dry well to the suppression pool. The wet well model developed with these assumptions is adequate to simulate slow transients such as extended station black-out transients. However, the current model is not suitable for LOCA analysis. With these assumptions, mass and energy balance equations apply for both gas and water spaces. By assuming one pressure for the gas space, another equation for the water level is obtained. The mass conservation equation for the gas space is

$$\frac{\mathrm{d}m_g}{\mathrm{d}t} = -\dot{m}_v \tag{908}$$

where m_g is the gas mass and $\dot{m}_v$ is the venting mass flow rate to the dry well which is obtained from the connected pipe controlled by the vacuum breaker.

Energy conservation equation for the gas space is

$$\frac{\mathrm{d}(me)_g}{\mathrm{d}t} = A_c\alpha\left(T_w - T_g\right) - \dot{m}_v H_v \tag{909}$$

where $(me)_g$ is the total internal energy (also total energy since kinetic energy is assumed to be 0) for the gas space, A_c the average cross section area for the wet well, α the effective

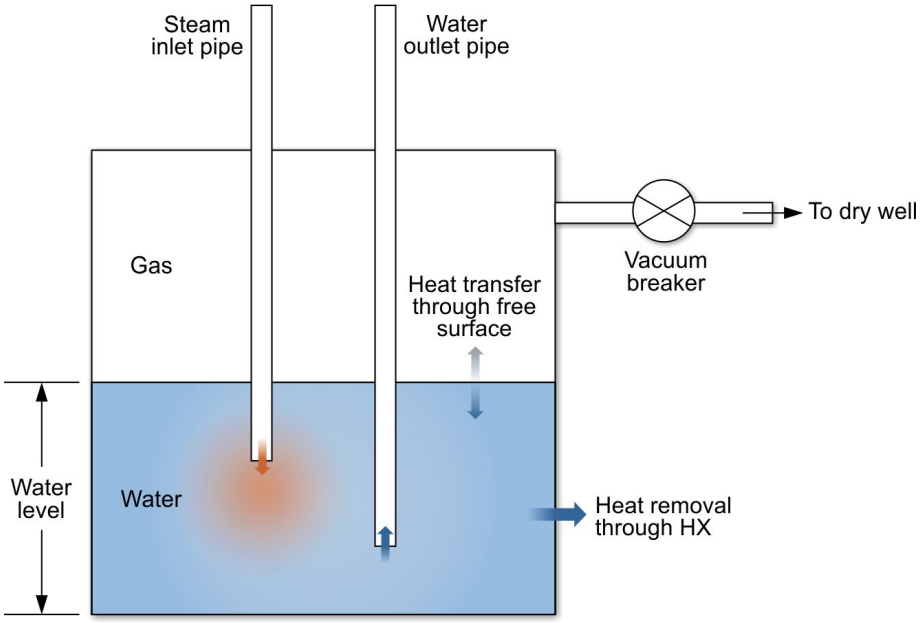

Figure 14. A simplified wet well model.

heat transfer coefficient given by user input, T_w and T_g are temperatures for water and gas, respectively. H_v is the total enthalpy from upstream. The small pressure work due to the change of the volume is ignored since the change of water volume is slow and small due to its tremendous volume. The gravity change inside the volume is ignored due to low density.

The mass conservation equation for the water space is

$$\frac{\mathrm{d}m_w}{\mathrm{d}t} = \dot{m}_{in} - \dot{m}_{out} \tag{910}$$

where m_w is the total mass of water, $\dot{m}_{in}$ is the inlet steam mass flow rate and obtained from the connected steam pipe, and $\dot{m}_{out}$ is the outlet water mass flow rate

$$\dot{m}_{out} = (\rho u)_{out} A_{out} \tag{911}$$

where $(\rho u)_{out}$ the outlet momentum which is coupled from the connected water pipe, and A_{out} the pipe cross section area.

The total energy conservation equation for the water space is

$$\begin{aligned}\frac{\mathrm{d}(me)_w}{\mathrm{d}t} &= \dot{m}_{in}\left(H_{in} + (z_i - 0.5L_w)g\right) \\ &- \dot{m}_{out}\left(H_{out} + (z_o - 0.5L_w)g\right) \\ &- A_c\alpha(T_w - T_g) - \dot{q}\end{aligned} \tag{912}$$

where $(me)_w$ is the total internal energy for the water space, H_{in} is the total enthalpy coupled from the connecting steam pipe, z_i is the inlet steam pipe end elevation relative to the pool bottom, L_w is the pool water level, z_o, of the outlet water pipe end elevation relative to the pool bottom, $\dot{q}$ is the active heat removal rate from the immersed heat exchanger, and H_{out} is the total enthalpy for the outlet water pipe which can be calculated for outflow as

$$H_{out} = \frac{\rho e_w + p_w(\rho_w, 0, \rho e_w)}{\rho_w} + \frac{1}{2}u_{out}^2 \ . \tag{913}$$

u_{out} is the exit speed and is obtained from coupled water pipe end. The methods to calculate the average water density ρ_w and specific volume energy ρe_w will be introduced shortly. For inflow condition, H_{out} will be coupled from the pipe end. In Eq. (912), it is assumed that the gravity center is at the half depth of the water pool. Reference pressure in the water space is defined at the middle elevation of the pool

$$p_w = p_g + \frac{1}{2}L_w\rho_w g \tag{914}$$

where p_w is the reference water pressure and p_g the gas pressure. Pressure and temperature are calculated from EOS relationships. The momentum of gas and water are assumed to be 0. Therefore, the total energy is

$$\rho E_t = \rho e = \rho\frac{me}{m} \ . \tag{915}$$

In the code implementation of the wet well model, m_g, $(me)_g$, m_w, $(me)_w$, and L_w are designated as the primary variables to be solve for, with corresponding equations (908), (909), (910), (912), and (914). Another set of auxiliary variables is defined to close the system, which include gas density ρ_g and water density ρ_w. Gas density is calculated according to

$$\rho_g = \frac{m_g}{A_c(L_t - L_w)} \tag{916}$$

where L_t is the total effective height of the wet well. Similarly, the average water density is calculated according to

$$\rho_w = \frac{m_w}{A_c L_w} \ . \tag{917}$$

Initial conditions for the primary variables are calculated according to the initial water level $L_w(0)$, gas pressure $p_g(0)$, and gas temperature $T_g(0)$. Boundary conditions for three connecting pipes are set similarly as for the reverse pump BCs. For example, the inlet steam pipe needs one BC p_{in}

$$p_{in} + \frac{1}{2}(\rho u^2)_{in} = p_i + K_{in}\frac{1}{2}(\rho u^2)_{in} \tag{918}$$

where K_{in} is the form loss coefficient and p_i is the water pressure at the elevation of inlet steam pipe end

$$p_i = p_w + (0.5L_w - z_i)\rho_w g \ . \tag{919}$$

The other two pipe's BCs are set in a similar manner.

9.14 Subchannel

A fully coupled subchannel model for the single-phase has been implemented into RELAP-7. The single-phase subchannel model includes four balance equations: mass, energy, axial momentum, and lateral momentum. The mass balance equation for the subchannel i is

$$A_i\frac{\partial \rho_i}{\partial t} + \frac{\partial(\rho_i u_i A_i)}{\partial x} + \sum_{j\in K(i)} w_{i,j} = 0 \tag{920}$$

where i is the index of subchannel i. A_i is the flow area for subchannel i. j is the index of a subchannel which is adjacent to subchannel i. $K(i)$ is the set of lateral interfaces (gaps) on the boundary of subchannel i. $w_{i,j} = \rho u_l s_k$ is the mass flow rate per unit length in the lateral direction across the gap k between subchannels i and j. s_k is the width of gap k.

The axial momentum balance for subchannel i is

$$A_i\frac{\partial \rho_i u_i}{\partial t} + \frac{\partial(\rho_i u_i u_i A_i)}{\partial x} + A_i\frac{\partial P_i}{\partial x} + A_i g\rho_i + \frac{1}{2}\left(\frac{f}{D_h} + K_i'\right)\rho_i u_i|u_i|A_i + \sum_{j\in K(i)} w_{i,j}u^* + \sum_{j\in K(i)} w_{i,j}^t(u_i - u_j) = 0 \tag{921}$$

where f is the wall friction coefficient, D_h is the subchannel hydraulic diameter, and K_i' is the form loss coefficient. u^* is the lateral donor axial velocity at gap face k. If the flow is into the subchannel i, then $u^* = u_j$, otherwise, $u^* = u_i$. $w_{i,j}^t$ is the turbulent mixing mass flow rate per unit length in the lateral direction at gap face k. $w_{i,j}^t$ is the fluctuating

crossflow which is related to the eddy diffusely ϵ_t, by $w^t_{i,j} = \epsilon_t \rho_i \frac{s_k}{l_k}$. In the current RELAP-7 implementation, $w^t_{i,j}$ is calculated as $w^t_{i,j} = \beta s_k \bar{G}$, where β is the turbulent mixing parameter and $\bar{G}$ is the average mass flux in the adjacent subchannels.

The lateral momentum balance for subchannel i is

$$\frac{\partial w_{i,j}}{\partial t} + \frac{\partial w_{i,j}\bar{u}}{\partial x} - \frac{s_k}{l_k}(P_i - P_j) + \frac{1}{2}\frac{s_k}{l_k} K_G \frac{|w_{i,j}|}{\bar{\rho} s_k^2} w_{i,j} = 0 \tag{922}$$

where $\bar{u} = \frac{1}{2}(u_i + u_j)$ and $\bar{\rho} = \frac{1}{2}(\rho_i + \rho_j)$. s_k is the width of lateral gap k. l_k is the distance between centroids of subchannels i and j. K_G is the lateral loss coefficient which accounts for the friction and form pressure loss caused by the area change.

The total energy balance equation for subchannel i is

$$\begin{aligned} A_i \frac{\partial \rho_i E_i}{\partial t} + \nabla \cdot (\rho_i u_i H_i A_i) + \rho_i g A_i u_i + \\ \sum_{j \in K(i)} w_{i,j} H^* + \sum_{j \in K(i)} w^t_{i,j}(H_i - H_j) + \sum_{j \in K(i)} \frac{k}{l_i}(T_i - T_j) + \\ \sum_{r \in M(i)} \phi_{i,r} h_w a_w A_i (T_i - T_{w,r}) = 0 \end{aligned} \tag{923}$$

where $H = E + \frac{P}{\rho}$ is the total enthalpy and H^* is the donor total enthalpy. k is the fluid thermal conductivity. $\phi_{i,r}$ is the heated perimeter fraction associated with the subchannel i. $M(i)$ is the set of fuel rods that surround the subchannel i. h_w is the convective heat transfer coefficient and a_w is the ratio of heat transfer surface area to the fluid volume. T_i is the fluids temperature in subchannel i and $T_{w,r}$ is the fuel rod wall temperature which is obtained from the solution of the heat conduction equation.

9.15 Reactor

The reactor component is a virtual component to allow users to specify the reactor power (i.e., steady-state power or decay heat curve) or heat source.

10 Reactor Kinetics Model

There will be two options available for the computation of the reactor power in the RELAP-7 code. The first option is the point kinetics model; this option has been implemented into RELAP-7. The second option will be a multi-dimensional neutron kinetics model. This option, which is not available yet, will be achieved through the coupling with the RattleSnake code. RattleSnake is the S_n neutron transport code being developed at the INL using the MOOSE framework. Chapter 8 has more in-depth discussions on this option.

The reactor point kinetics model is the simplest model that can be used to compute the transient behavior of the neutron fission power in a nuclear reactor. The power is computed using the space-independent, or point kinetics, approximation which assumes that power can be separated into space and time functions. This approximation is adequate for cases in which the space distribution remains nearly constant.

The point kinetics model computes both the immediate (prompt and delayed neutrons) fission power and the power from decay of fission products. The immediate power is that released at the time of fission and includes power from kinetic energy of the fission products and neutron moderation. Decay power is generated as the fission products undergo radioactive decay. The user can select the decay power model based on the RELAP-7 exact implementation of the 1979 ANSI/ANS Standard, the 1994 ANSI/ANS Standard, or the 2005 ANSI/ANS Standard.

10.1 Point Kinetics Equations

The point kinetics equations are the following:

$$\frac{dn(t)}{dt} = \frac{\rho(t) - \beta}{\Gamma} n(t) + \sum_{i=1}^{N_d} \lambda_i C_i(t) + S \tag{924}$$

$$\frac{dC_i(t)}{dt} = \frac{\beta f_i}{\Gamma} n(t) - \lambda_i C_i(t),\ i = 1, 2, \ldots, N_d \tag{925}$$

where t is time (s), n is the neutron density (neutrons/m^3), ρ is the reactivity (only the time-dependence has been indicated, however, the reactivity is dependent on other variables). β_i is the effective delayed neutron precursor yield of group i and $\beta = \sum_{i=1}^{N_d} \beta_i$ is the effective delayed neutron fraction. Γ is the prompt neutron generation time (s). λ_i is the decay constant of group i (1/s). C_i is the delayed neutron precursor concentration in group i

(nuclei/m^3). N_d is the number of delayed neutron precursor groups. $f_i = \frac{\beta_i}{\beta}$ is the fraction of delayed neutrons of group i. S is the source rate density (neutrons/m^3-s).

The neutron flux (neutrons/m^2-s) is calculated as

$$\phi(t) = n(t)v \tag{926}$$

where v is neutron velocity (m/s). The fission rate (fissions/s) $\psi(t)$ is calculated as

$$\psi(t) = V\Sigma_f\phi(t) \tag{927}$$

where V is the volume (m^3) and Σ_f is the macroscopic fission cross section (1/m). The reactor power is calculated from

$$P_f(t) = Q_f\psi(t) \tag{928}$$

where P_f is the immediate (prompt and delayed neutron) fission power (Mev/s) and Q_f is the immediate fission energy per fission (Mev/fission).

10.2 Fission Product Decay Model

The 1979, 1994, and 2005 Standards for decay power can be implemented by advancing the differential equations, which become

$$\frac{d\gamma_{\alpha j}(t)}{dt} = \frac{F_\gamma a_{\alpha j}}{\lambda_{\alpha j}} F_\alpha\psi(t) - \lambda_{\alpha j}\gamma_{\alpha j}(t) \quad j = 1, 2, \ldots, N_\alpha \tag{929}$$

where $\alpha = 1, 2, 3$ for the 1979 Standard and $\alpha = 1, 2, 3, 4$ for the 1994 and 2005 Standards. The parameters a and λ were obtained by fitting to fission decay power data. The fitting for each isotope used 23 groups ($N_\alpha = 23$). For the 1979 Standard, data are presented for three isotopes, U^{235}, U^{238}, and Pu^{239}. For the 1994 and 2005 Standards, data are presented for four isotopes, U^{235}, U^{238}, Pu^{239}, Pu^{241}. F_γ is an input factor to allow easy specification of a conservative calculation. It is usually 1.0 for best-estimate calculations. F_α is the fraction of fissions from isotope α. Summation of F_α over α is 1.0. The uncorrected decay power is calculated as

$$P'_\gamma(t) = \sum_{\alpha=1}^{N_I}\sum_{j=1}^{N_\alpha} \lambda_{\alpha j}\gamma_{\alpha j}(t) \tag{930}$$

where $N_I = 3$ for the 1979 Standard and $N_I = 4$ for the 1994 and 2005 Standards. ψ is the fission rate from all isotopes.

The 1979, 1994, and 2005 Standards use a correction factor to the energy from fission product decay to account for the effects of neutron absorption. The equation for the correction factor is the following:

$$G(t) = 1.0 + (3.24E - 6 + 5.23E - 10t)T^{0.4}\psi_g \tag{931}$$

where ψ_g is the number of fissions per initial fissile atom, T is the reactor operating time including any periods of shutdown, and t is the time since shutdown. Limits on the quantities are $1.0 \leq \psi_g \leq 3.0$, $T < 1.2614 \times 10^8$, and $t < 10^4$ seconds. The corrected decay power is given by

$$P_\gamma = G(t)P'_\gamma \ . \tag{932}$$

The RELAP-7 implementation of the 1979, 1994, and 2005 Standards is exact (i.e., not a curve fit). The data for all standards are built into the code as default data, but the user may enter different data.

10.3 Actinide Decay Model

The actinide model describes the production of U^{239}, Np^{239}, and Pu^{239} from neutron capture by U^{238} using the descriptive differential equations

$$\frac{d\gamma_U(t)}{dt} = F_U\psi(t) - \lambda_U\gamma_U(t) \tag{933}$$

$$\frac{d\gamma_N(t)}{dt} = \lambda_U\gamma_U(t) - \lambda_N\gamma_N(t) \ . \tag{934}$$

The actinide decay power is calculated as

$$P_\alpha(t) = \eta_U\lambda_U\gamma_U(t) + \eta_N\lambda_N\gamma_N(t) \ . \tag{935}$$

The quantity F_U is user-specified and is the number of atoms of U^{239} produced by neutron capture in U^{238} per fission from all isotopes. A conservative factor, if desired, should be factored into F_U. The λ and η values can be user-specified, or default values equal to those stated in the 1979, 1994, or 2005 ANS Standards can be used. The first equation describes the rate of change of atoms of U^{239}. The first term on its right hand side represents the production of U^{239}; the last term is the loss of U^{239} due to beta decay. The second equation describes the rate of change of NP^{239}. The production of Np^{239} is from the beta decay of U^{239}, and Pu^{239} is formed from the decay of Np^{239}.

10.4 Transformation of Equations for Solution

The differential equations to be advanced in time are the point kinetics equations, fission products decay equations, and actinide decay equations. Multiplying by $V\Sigma_f$ and X which is the conversion factor from MeV/s to Watts, the equations become

$$\frac{d}{dt}\left[\frac{X\psi(t)}{v}\right] = \frac{[\rho(t)-\beta]X\psi(t)}{\Gamma v} + \sum_{i=1}^{N_d}\lambda_i XV\Sigma_f C_i(t) + XV\Sigma_f S \tag{936}$$

$$\frac{d}{dt}[X\Sigma_f C_i(t)] = \frac{\beta f_i X\psi(t)}{\Gamma v} - \lambda_i XV\Sigma_f C_i(t) \quad i = 1, 2, \ldots, N_d \tag{937}$$

$$\frac{d}{dt}[X\gamma_{\alpha j}(t)] = \frac{F_\gamma a_{\alpha j} F_\alpha X\psi(t)}{\lambda_{\alpha j}} - \lambda_{\alpha j} X\gamma_{\alpha j}(t) \quad j = 1, 2, \ldots, N_\alpha \tag{938}$$

$$\frac{d}{dt}[X\gamma_U(t)] = F_U X\psi(t) - \lambda_U X\gamma_U(t) \tag{939}$$

$$\frac{d}{dt}[X\gamma_N(t)] = \lambda_U X\gamma_U(t) - \lambda_N X\gamma_N(t) \tag{940}$$

where $\alpha = 1, 2, 3$ for the 1979 Standard and $\alpha = 1, 2, 3, 4$ for the 1994 and 2005 Standards. The total power P_T is the sum of immediate fission power, corrected fission product decay, and actinide decay power, and now in units of watts is

$$P_T(t) = Q_f X\psi(t) + G(t)\sum_{\alpha=1}^{N_I}\sum_{j=1}^{N_\alpha}\lambda_{\alpha j} X\gamma_{\alpha j}(t) + \eta_U\lambda_U X\gamma_U(t) + \eta_N\lambda_N X\gamma_N(t) \tag{941}$$

where $N_I = 3$ for 1979 Standard and $N_I = 4$ for the 1994 and 2005 Standard. For solution convenience, the following substitutions are made:

$$\rho(t) = \beta r(t) \tag{942}$$

$$X\psi(t) = \psi'(t) \tag{943}$$

$$\frac{XV\Sigma_f\Gamma v S}{\beta} = S' \tag{944}$$

$$XV\Sigma_f v C_i(t) = \frac{\beta f_i}{\Gamma\lambda_i} W_i(t) \quad i = 1, 2, \ldots, N_d \tag{945}$$

$$X\gamma_{\alpha j}(t) = \frac{F_\gamma a_{\alpha j} F_\alpha}{\lambda_{\alpha j}^2} Z_{\alpha j}(t) \quad j = 1, 2, \ldots, N_\alpha \tag{946}$$

$$X\gamma_U(t) = \frac{F_U}{\lambda_U} Z_U(t) \tag{947}$$

$$X\gamma_N(t) = Z_N(t) \tag{948}$$

where $\alpha = 1, 2, 3$ for the 1979 Standard and $\alpha = 1, 2, 3, 4$ for the 1994 and 2005 Standards. The equations to be integrated are now

$$\frac{d}{dt}\psi'(t) = \frac{\beta}{\Gamma}\left[(r(t)-1)\psi'(t) + \sum_{i=1}^{N_d} f_i W_i(t) + s'\right] \tag{949}$$

$$\frac{d}{dt}W_i(t) = \lambda_i \psi'(t) - \lambda_i W_i(t) \quad i = 1, 2, \ldots, N_d \tag{950}$$

$$\frac{d}{dt}Z_{\alpha j}t = \lambda_{\alpha j}\psi'(t) - \lambda_{\alpha j} Z_{\alpha j}(t) \quad i = 1, 2, \ldots, N_d \tag{951}$$

$$\frac{d}{dt}Z_U(t) = \lambda_U \psi'(t) - \lambda_U Z_U(t) \tag{952}$$

$$\frac{d}{dt}Z_N(t) = F_U Z_U(t) - \lambda_N Z_N(t) \tag{953}$$

where $\alpha = 1, 2, 3$ for the 1979 standard and $\alpha = 1, 2, 3, 4$ for the 1994 and 2005 standards. The total power is given by

$$P_T(t) = Q_f \psi'(t) + G(t) \sum_{\alpha=1}^{N_I} \sum_{j=1}^{N_\alpha} \frac{F_\gamma a_{\alpha j} F_\alpha Z_{\alpha j}(t)}{\lambda_{\alpha j}} + F_U \eta_U Z_U(t) + \eta_N \lambda_N Z_N(t) \tag{954}$$

where $N_I = 3$ for the 1979 standard and $N_I = 4$ for the 1994 and 2005 Standards.

10.5 Reactivity Feedback Model

The reactivity feedback model implemented in RELAP-7 is the same as the separable model used for RELAP5. In the separable model, each effect is assumed to be independent of the other effects. The model assumes nonlinear feedback effects from moderator (thermal fluids) density and fuel temperature changes and linear feedback from moderator and fuel temperature changes. The separable model defining reactivity is defined as:

$$r(t) = \sum_{i=1}^{n_s} r_{si}(t) + \sum_{i=1}^{n_\rho} [W_{\rho i} R_\rho(\rho_i(t)) + a_{Mi} \Delta T_{Mi}(t)] + \sum_{i=1}^{n_F} [W_{Fi} R_F(T_{Fi}(t)) + a_{Fi} \Delta T_{Fi}(t)] \tag{955}$$

The quantities r_{si} are obtained from input tables defining n_s reactivity (scram) curves as a function of time. R_ρ is a table defining reactivity as a function of the current moderator density of fluid $\rho_i(t)$ in the thermal fluids volume i (density reactivity table). $W_{\rho i}$ is the

density volume weighting factor for volume i. $\Delta T_{Mi}(t)$ is the spatially averaged moderator fluid temperature difference between the current time t and the start of the transient for volume i. a_{Mi} is the volume fluid temperature coefficient (not including density changes) for volume i and n_ρ is the number of thermal fluids volumes in the reactor core. The quantity R_F is a table defining the Doppler reactivity as a function of the heat structure plume average fuel temperature $T_{Fi}(t)$ in the heat structure. $\Delta T_{Fi}(t)$ is the difference between the current time t and the start of the transient. W_{Fi} and a_{Fi} are the fuel temperature heat structure weighting factor and the heat structure fuel temperature coefficient, respectively, for heat structure i. Finally, n_F is the number of fuel volumes in a reactor core.

Boron feedback is not provided, but will be added in a later version. The separable model can be used if boron changes are quite small and the reactor is near critical about only one state point.

References

[1] F. J. Moody, *Introduction to Unsteady Thermofluid Mechanics*. New York, U.S.A.: John Wiley & Sons, Inc., 1990.

[2] US NRC, "TRACE V5. 0 Theory Manual, "Field Equations, Solution Methods, and Physical Models"," tech. rep., US NRC. draft.

[3] H. I. H. Saravanamuttoo *et al.*, *Gas Turbine Theory, 5th edition*. Pearson Education Limited, 2001.

[4] O. L. Métayer, J. Massoni, and R. Saurel, "Elaborating equations of state of a liquid and its vapor for two-phase flow models," *Int. J. Thermal Sciences (in French)*, vol. 43, pp. 265–276, Mar. 2004. `http://dx.doi.org/10.1016/j.ijthermalsci.2003.09.002`.

[5] D. Gaston, C. Newman, G. Hansen, and D. Lebrun-Grandié, "MOOSE: A parallel computational framework for coupled systems of nonlinear equations," *Nuclear Engineering and Design*, vol. 239, pp. 1768–1778, Oct. 2009. `http://dx.doi.org/10.1016/j.nucengdes.2009.05.021`.

[6] R. L. Moore, S. M. Sloan, R. R. Schultz, and G. E. Wilson, "RELAP5/MOD3 Code Manual, Volume 1: Code Structure, System Models, and Solution Methods," Tech. Rep. NUREG/CR-5335, INEL-95/0174, Idaho National Engineering Laboratory, June 1995. `http://pbadupws.nrc.gov/docs/ML1103/ML110380259.pdf`.

[7] D. A. Knoll and D. E. Keyes, "Jacobian-free Newton-Krylov methods: a survey of approaches and applications," *Journal of Computational Physics*, vol. 193, pp. 357–397, Jan. 2004. `http://dx.doi.org/10.1016/j.jcp.2003.08.010`.

[8] P. N. Brown and A. C. Hindmarsh, "Matrix-free methods for stiff systems of ODEs," *SIAM J. Numer. Anal.*, vol. 23, pp. 610–638, June 1986. `http://www.jstor.org/stable/2157527`.

[9] B. S. Kirk, J. W. Peterson, R. H. Stogner, and G. F. Carey, "`libMesh`: A C++ Library for Parallel Adaptive Mesh Refinement/Coarsening Simulations," *Engineering with Computers*, vol. 22, no. 3–4, pp. 237–254, 2006. `http://dx.doi.org/10.1007/s00366-006-0049-3`.

[10] S. Balay, W. D. Gropp, L. Curfman-McInnes, and B. F. Smith, "Efficient management of parallelism in object oriented numerical software libraries," in *Modern Software Tools in Scientific Computing* (E. Arge, A. M. Bruaset, and H. P. Langtangen, eds.), pp. 163–202, Birkhäuser Press, 1997.

[11] M. Heroux *et al.*, "An overview of Trilinos," Tech. Rep. SAND2003-2927, Sandia National Laboratories, 2003.

[12] A. Prosperetti and G. Tryggvason, *Computational methods for multiphase flow*. Cambridge University Press, 2007.

[13] R. A. Berry, R. Saurel, and O. LeMetayer, "The discrete equation method (DEM) for fully compressible, two-phase flows in ducts of spatially varying cross-section," *Nuclear Engineering and Design*, vol. 240, pp. 3797–3818, 2010.

[14] R. A. Berry, "Notes on well-posed ensemble averaged conservation equations for multiphase, multi-component, and multi-material flows," Tech. Rep. INL/EXT-05-00516-modified, Idaho National Laboratory, U. S. A., 2008. Modified from INL/EXT-05-00516 (2005) and INEEL/EXT-03-01011 (2003).

[15] R. A. Berry, R. Saurel, F. Petitpas, E. Daniel, O. LeMetayer, S. Gavrilyuk, N. Dovetta, and R. C. Martineau, "Progress in the development of compressible, multiphase flow modeling capability for nuclear reactor flow applications," Tech. Rep. INL/EXT-08-15002, Idaho National Laboratory, U. S. A., 2008.

[16] H. Zhang, L. Zou, D. Andrs, H. Zhao, and R. C. Martineau, "Point Kinetics Calculations with Fully Coupled Thermal Fluids Reactivity Feedback," in *International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2013)*, (Sun Valley, Idaho, USA), May 5–9, 2013.

[17] L. Zou, J. Peterson, H. Zhao, H. Zhang, D. Andrs, and R. C. Martineau, "Solving Multi-Mesh Flow and Conjugate Heat Transfer Problems with RELAP-7," in *International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2013)*, (Sun Valley, Idaho, USA), May 5–9, 2013.

[18] Y. Wang, "Nonlinear diffusion acceleration for the multigroup transport equation discretized with S_N and continuous FEM with RattleSnake," in *International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2013)*, (Sun Valley, Idaho, USA), May 5–9, 2013.

[19] R. D. Falgout and U. M. Yang, "HYPRE: A Library of High Performance Preconditioners," in *International Conference on Computational Science*, pp. 632–641, 2002.

[20] R. C. Martineau, "The PCICE-FEM Scheme for Highly Compressible Axisymmetric Flows," *Computers & Fluids*, vol. 36, pp. 1259–1272, 2007.

[21] R. C. Martineau and R. A. Berry, "The Pressure-Corrected ICE Finite Element Method (PCICE-FEM) for Compressible Flows on Unstructured Meshes," *Journal of Computational Physics*, vol. 198, pp. 659–685, 2004.

[22] S. Y. Kadioglu, R. A. Berry, and R. C. Martineau, "A Point Implicit Intgration Technique for Slow Transient Flow Problems," *Nuclear Engineering and Design*, vol. 286, pp. 130–138, 2015.

[23] INL, "RELAP5-3D© Code Manual Volume I: Code Structure, System Models and Solution Methods, Vol. 1, revision 2.4," Tech. Rep. INEEL-EXT-98-00834, Idaho National Laboratory, June 2005.

[24] D. J. Zigrang and N. D. Sylvester, "A review of explicit friction factor equations," *Transactions of ASME, Journal of Energy Resources Technology*, vol. 107, pp. 280–283, 1985.

[25] C. F. Colebrook, "Turbulent flow in pipes with particular reference to the transition region between smooth and rough pipe laws," *Journal of Institute of Civil Engineers*, vol. 11, pp. 133–156, 1939.

[26] INL, "RELAP5-3D© Code Manual Volume IV: Models and Correlations," Tech. Rep. INEEL-EXT-98-00834, Idaho National Laboratory, June 2012.

[27] A. Y. Inayatov, "Correlation of Data on Heat Transfer Flow Parallel to Tube Bundles at Relative Tube Pitches of $1.1 < s/d < 1.6$," *Heat Transfer-Soviet Research*, vol. 7, no. 3, pp. 84–88, 1975.

[28] R. Menikoff and B. J. Plohr, "The Riemann problem for fluid flow of real materials," *Reviews of Modern Physics*, vol. 61, no. 1, pp. 75–130, 1989. `http://dx.doi.org/10.1103/RevModPhys.61.75`.

[29] R. Courant and K. Friedrichs, *Supersonic flow and shock waves*. New York: Interscience, 1948.

[30] F. Dobran, *Theory of Structured Multiphase Mixtures*. Berlin: Springer-Verlag, 1991.

[31] S. Whitaker, *The Method of Volume Averaging*. Dordrecht: Kluwer Academic, 1999.

[32] M. Ishii, *Thermo-Fluid Dynamic Theory of Two-Phase Flow*. France: Eyrolles, 1975.

[33] D. A. Drew and R. T. Lahey, "Analytical modeling of multiphase flow," in *Particulate Two-Phase Flow* (M. C. Roco, ed.), (Boston), Butterworth-Heinemann, 1993.

[34] D. A. Drew and S. L. Passman, *Theory of Multicomponent Fluids*. New York: Springer-Verlag, 1999.

[35] B. A. Kashiwa and R. M. Rauenzahn, "A multi material formalism," in *Numerical Methods in Multiphase Flow*, vol. 185, pp. 149–157, ASME, Fluids Engineering Division, 1994.

[36] D. Lhuillier, "The macroscopic modeling of multi-phase mixtures," in *Flow of Particles in Suspensions* (U. Schaflinger, ed.), New York: Springer-Verlag, 1996.

[37] J. U. Brackbill, N. L. Johnson, B. A. Kashiwa, and W. B. Vanderheyden, "Multiphase flows and particle methods," in *The Fifth Annual Conference of the Computational Fluid Dynamics Society of Canada*, (Victoria, British Columbia), 1997.

[38] B. A. Kashiwa and E. S. Gaffney, "Design basis for CFDlib," Tech. Rep. LA-UR-03-1295, Los Alamos National Laboratory, U.S.A, 2003.

[39] I. M. Gelfand and G. E. Shilov, *Generalized Functions*. New York: Academic, 1964.

[40] D. A. Drew, "Mathematical modeling of two-phase flow," *Annual Rev. Fluid Mech.*, vol. 15, pp. 261–291, 1983. `http://www.annualreviews.org/doi/pdf/10.1146/annurev.fl.15.010183.001401`.

[41] I. Kataoka and A. Serizawa, "Interfacial area concentration and its role in local instant formulation of two-phase flow," in *Transient Phenomena in Multiphase Flow* (N. H. Afgan, ed.), (New York), Hemisphere, 1988.

[42] R. I. Nigmatulin, *Dynamics of Multiphase Media*. New York: Hemisphere, 1990.

[43] D. C. Wilcox, *Turbulence Modeling for CFD*. La Canada, California: DCW Industries, 1998.

[44] C. Truesdell, *Rational Thermodynamics*. New York: Springer-Verlag, 1984.

[45] S. L. Passman, J. W. Nunziato, and E. K. Walsh, "Appendix 5c: A theory of multiphase mixtures," in *Rational Thermodynamics* (C. Truesdell, ed.), New York: Springer-Verlag, second ed., 1984. Corrected and enlarged edition of C. Truesdell's original book with adjoined appendices by multiple authors.

[46] V. H. Ransom and M. P. Scofield, "Two-pressure hydrodynamic model for two-phase separated flow," Tech. Rep. SRD-50-76, Idaho National Laboratory, U. S. A., 1976.

[47] L. Agee, R. B. Duffey, S. Banerjee, and E. D. Hughes, "Aspects of two-fluid models for two-phase flow and their numerical solution," in *Second OECD/NEA Specialists' Meeting on Transient Two-Phase Flow*, (France), CEA, 1978.

[48] H. Nguyen, "One-dimensional models for transient two-phase separated flow," in *Third CSNI Specialist Meeting on Transient Two-Phase Flow*, (Pasadena, California), 1981.

[49] D. L. Hicks, "Hyperbolic models for two-phase (or two-material) flow," Tech. Rep. SAND81-0253, Sandia National Laboratories, U. S. A., 1981.

[50] M. R. Baer and J. W. Nunziato, "A theory of deflagration-to-detonation transition (DDT) in granular explosives," *Int. J. of Multiphase Flow*, vol. 12, pp. 861–889, 1986.

[51] R. Saurel and R. Abgrall, "A multiphase Godunov method for compressible multifluid and multiphase flows," *J. of Computational Physics*, vol. 150, pp. 425–467, 1999.

[52] J. H. Stuhmiller, "The influence of interfacial pressure forces on the character of two-phase flow model equations," *Int. J. of Multiphase Flow*, vol. 3, pp. 551–560, 1977.

[53] J. C. Rousseau and R. L. Ferch, "A note on two-phase separated flow models," *Int. J. of Multiphase Flow*, vol. 5, pp. 489–494, 1979.

[54] S. Banerjee, "Analysis of separated flow models," Tech. Rep. NP-1442, Electric Power Research Institute, 1979. Research project 888-2.

[55] W. T. Hancox, R. L. Ferch, W. S. Liu, and R. E. Niemann, "One-dimensional models for transient gas-liquid flows in ducts," *Int. J. of Multiphase Flow*, vol. 6, pp. 25–40, 1980.

[56] S. J. Lee, K. S. Chang, and S. J. Kim, "Surface tension effect in the two-fluids equation system," *Int. J. of Heat and Mass Transfer*, vol. 41, pp. 2821–2826, 1998.

[57] M. S. Chung, W. J. Lee, and K. S. Ha, "Choked flow calculations of two-phase bubbly flow," *Numerical Heat Transfer, Part A*, vol. 42, pp. 297–305, 2002.

[58] R. A. Berry and R. L. Williamson, "A multiphase mixture model for the shock induced consolidation of metal powders," in *Shock Waves in Condensed Matter*, New York: Plenum Press, 1985. Proc. of 4th APS Topical Conf. on Shock Waves in Condensed Matter, Spokane, WA, 1985.

[59] R. J. Leveque, *Finite Volume Methods for Hyperbolic Problems*. Cambridge, UK: Cambridge University Press, 2002.

[60] R. A. Berry, "A logical progression of steps for implementation and testing of the 7-equation, two-phase model into a computational framework," in *Int. Conf. on Mathematics and Computational Methods Applied to Nuclear Science and Engineering*, (LaGrange Park, IL), American Nuclear Society, 2013.

[61] J. A. Boure and J. M. Delhaye, "General equations and two-phase flow modeling," in *Handbook of Multiphase Systems* (G. Hetsroni, ed.), Washington: Hemisphere, 1982.

[62] S. Banerjee and W. T. Hancox, "Transient thermohydraulics analysis for nuclear reactors," in *Sixth International Heat Transfer Conference, Toronto, 1978*, Washington: Hemisphere, 1978. Authors' reply during discussion session.

[63] J. Hadamard, "Sur les problemes aux derivees partielles et leur signification physique," *Princeton University Bulletin*, pp. 49–52, 1902.

[64] E. D. Hughes, R. W. Lyczkowski, J. H. McFadden, and G. F. Niederauer, "An evaluation of state-of-the-art two-velocity two-phase flow models and their applicability to nuclear reactor transient analysis," Tech. Rep. EPRI NP-143, vol. 1, 2, 3, Electric Power Research Institute, U. S. A., 1976. Research project TSA SOA 75-317, Energy Incorporated.

[65] D. Bestion, "The physical closure laws in the cathare code," *Nuclear Engineering and Design*, vol. 124, pp. 229–245, 1990.

[66] R. A. Berry, "The seven-, five-, four-, and three-equation models of compressible, two-phase flow: A fully consistent, hierarchical set." In preparation, 2010.

[67] A. K. Kapila, R. Menikoff, J. B. Bdzil, S. F. Son, and D. S. Stewart, "Two-phase modeling of deflagration-to-detonation transition in granular materials: reduced equations," *Physics of Fluids*, vol. 13, pp. 3002–3024, 2001.

[68] A. Murrone and H. Guillard, "A five equation reduced model for compressible two phase flow problems," *J. of Computational Physics*, vol. 202, pp. 664–698, 2005.

[69] R. Abgrall and R. Saurel, "Discrete equations for physical and numerical compressible multiphase mixtures," *J. of Computational Physics*, vol. 186, pp. 361–396, 2003.

[70] A. Chinnayya, E. Daniel, and R. Saurel, "Modelling detonation waves in heterogeneous energetic materials," *J. of Computational Physics*, vol. 196, pp. 490–538, 2004.

[71] R. Saurel, F. Petitpas, and R. A. Berry, "Simple and efficient relaxation methods for interfaces separating compressible fluids, cavitating flows and shocks in multiphase mixtures," *J. of Computational Physics*, vol. 228, pp. 1678–1712, 2009.

[72] J. M. Herrard and O. Hurisse, "A simple method to compute standard two-fluid models," *Int. J. of Computational Fluid Dynamics*, vol. 19, pp. 475–482, 2005.

[73] K. K. Kuo, *Principles of Combustion*. New Jersey, U.S.A.: John Wiley & Sons, Inc., second ed., 2005.

[74] C. W. Hirt and N. C. Romero, "Application of a drift-flux model to flashing in straight pipes," Tech. Rep. LA-6005-MS, Los Alamos National Laboratory, U.S.A., 1975.

[75] M. Kunick and H.-J. Kretzschmar, "Guideline on the fast calculation of steam and water properties with the spline-based table look-up method (sbtl)," tech. rep., The International Association for the Properties of Water and Steam, Moscow, Russia, 2015.

[76] M. Kunick, "Fast calculation of thermophysical properties in extensive process simulations with the spline-based table look-up method (sbtl)," tech. rep., VDI Fortschritt-Berichte, Germany, 2016. in preparation.

[77] M. Kunick, H.-J. Kretzschmar, F. di Mare, and U. Gampe, "Cfd analysis of steam turbines with the iapws standard on the spline-based table look-up method (sbtl) for the fast calculation of real fluid properties," in *Proceedings of the ASME Turbo Expo 2015: Turbine Technology Conference and Exposition*, (New York, NY), American Society of Mechanical Engineers, 2015. Montreal, Canada.

[78] H. Spath, *Two Dimensional Spline Interpolation Algorithms*. Wellesly, MA, U.S.A: A. K. Peters, 1995.

[79] J. T. Oden and J. N. Reddy, *An Introduction to the Mathematical Theory of Finite Elements*. New York: Wiley & Sons, 1976.

[80] J.-L. Guermond and R. Pasquetti, "Entropy viscosity method for higher-order approximations of conservation laws," *Lecture Notes in Computational Science and Engineering*, vol. 76, pp. 411–418, 2011.

[81] J.-L. Guermond, R. Pasquetti, and B. Popov, "Entropy viscosity method for nonlinear conservation laws," *J. of Computational Physics*, vol. 230, pp. 4248–4267, 2011.

[82] M. O. Delchini, J. C. Ragusa, and R. A. Berry, "Entropy viscosity method applied to euler equations," in *Int. Conf. on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C2013)*, (LaGrange Park, IL), American Nuclear Society, 2013. Sun Valley, ID.

[83] V. Zingan, J.-L. Guermond, J. Morel, and B. Popov, "Implementation of the entropy viscosity method with the discontinuous galerkin method," *Computational Methods in Applied Mechanics and Engineering*, vol. 253, pp. 479–490, 2013.

[84] J. K. Dukowicz, "A general, non-iterative riemann solver for godunov's method," *Journal of Computational Physics*, vol. 61, pp. 119–137, 1985.

[85] M. L. Wilkins, "Use of artificial viscosity in multimensional fluid dynamic calculations," *Journal of Computational Physics*, vol. 36, pp. 281–303, 1980.

[86] M. O. Delchini, J. C. Ragusa, and R. A. Berry, "Entropy-based viscous regularization for multi-dimensional euler equations in low-mach and transonic flows," *Computers & Fluids*, vol. 118, pp. 225–244, 2015.

[87] M. O. Delchini, "Extension of the entropy-based viscosity method to low-mach fluid flows and to seven-equation two-phase flows," tech. rep., Texas A&M University, Nuclear Engineering Department, U. S. A., 2014. Ph.D. Dissertation.

[88] R. M. Bowen, *Introduction to Continuum Mechanics for Engineers*. New York, U.S.A.: Plenum Press, 1989.

[89] M. O. Delchini, J. C. Ragusa, and R. A. Berry, "Viscous regularization for the non-equilibrium seven-equation two-phase flow model," 2016. submitted for publication, in review.

[90] T. J. R. Hughes and M. Mallet, "A new finite element formulation for computational fluid dynamics: III. the generalized streamline operator for multidimensional advective–diffusive systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 58, pp. 305–328, 1986.

[91] P. Nithiarasu, O. C. Zienkiewicz, B. V. K. S. Sai, K. Morgan, R. Codina, and M. Vasquez, "Shock capturing viscosities for the general fluid mechanics algorithm," *International Journal for Numerical Methods in Fluids*, vol. 28, pp. 1325–1353, 1998.

[92] M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz, "A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes," *Journal of Computational Physics*, vol. 227, no. 18, pp. 8209–8253, 2008.

[93] M. Dumbser and O. Zanotti, "Very high order $P_N P_M$ schemes on unstructured meshes for the resistive relativistic MHD equations," *Journal of Computational Physics*, vol. 228, no. 18, pp. 6991–7006, 2009.

[94] M. Dumbser, "Arbitrary high order $P_N P_M$ schemes on unstructured meshes for the compressible Navier–Stokes equations," *Computers & Fluids*, vol. 39, no. 1, pp. 60–76, 2010.

[95] D. S. Balsara, C. Altmann, C.-D. Munz, and M. Dumbser, "A sub-cell based indicator for troubled zones in RKDG schemes and a novel class of hybrid RKDG+HWENO schemes," *Journal of Computational Physics*, vol. 226, no. 1, pp. 586–620, 2007.

[96] H. Luo, L. Luo, R. Nourgaliev, V. A. Mousseau, and N. Dinh, "A reconstructed discontinuous Galerkin method for the compressible Navier–Stokes equations on arbitrary grids," *Journal of Computational Physics*, vol. 229, no. 19, pp. 6961–6978, 2010.

[97] H. Luo, L. Luo, A. Ali, R. Nourgaliev, and C. Cai, "A parallel, reconstructed discontinuous Galerkin method for the compressible flows on arbitrary grids," *Communications in Computational Physics*, vol. 9, no. 2, pp. 363–389, 2011.

[98] H. Luo, Y. Xia, S. Spiegel, R. Nourgaliev, and Z. Jiang, "A reconstructed discontinuous Galerkin method based on a Hierarchical WENO reconstruction for compressible flows on tetrahedral grids," *Journal of Computational Physics*, vol. 236, pp. 477–492, 2013.

[99] L. Zhang, L. Wei, H. Lixin, D. Xiaogang, and Z. Hanxin, "A class of hybrid DG/FV methods for conservation laws I: Basic formulation and one-dimensional systems," *Journal of Computational Physics*, vol. 231, no. 4, pp. 1081–1103, 2012.

[100] L. Zhang, L. Wei, H. Lixin, D. Xiaogang, and Z. Hanxin, "A class of hybrid dg/fv methods for conservation laws ii: Two-dimensional cases," *Journal of Computational Physics*, vol. 231, no. 4, pp. 1104–1120, 2012.

[101] Y. Xia, X. Liu, and H. Luo, "A finite volume method based on a WENO reconstruction for compressible flows on hybrid grids," in *52nd AIAA Aerospace Sciences Meeting*, no. 2014–0939, (National Harbor, Maryland, United States), January 2014.

[102] P. L. Roe, "Some contributions to the modelling of discontinuous flows," in *Large-Scale Computations in Fluid Mechanics*, pp. 163–193, 1985.

[103] B. Van Leer, "Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection," *Journal of computational physics*, vol. 23, no. 3, pp. 276–299, 1977.

[104] P. Batten, M. Leschziner, and U. Goldberg, "Average-state Jacobians and implicit methods for compressible viscous and turbulent flows," *Journal of computational physics*, vol. 137, no. 1, pp. 38–78, 1997.

[105] B. Einfeldt, C.-D. Munz, P. L. Roe, and B. Sjögreen, "On Godunov-type methods near low densities," *Journal of computational physics*, vol. 92, no. 2, pp. 273–295, 1991.

[106] P. L. Roe, "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, vol. 43, no. 2, pp. 357–372, 1981.

[107] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*. New York: John Wiley & Sons, 2003.

[108] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Philadelphia: SIAM, 1998.

[109] P. A. Thompson, *Compressible-Fluid Dynamics*. New Jersey, U.S.A: John Wiley & Sons, Inc., 1988.

[110] V. Guinot, *Wave Propagation in Fluids*. London, UK: ISTE Ltd, 2008.

[111] Idaho National Laboratory. Communication between INL and U.S. NRC, 2016.

[112] D. C. Groeneveld, et. al., "The 1995 look-up table for critical heat flux in tubes," *Nuclear Engineering and Design*, vol. 163, pp. 1–23, 1996.

[113] D. L. Hagrman and G. A. Reymann, "MATPRO-Version 11: A handbook of materials properties for use in the analysis of light water reactor fuel rod behavior," Tech. Rep. NUREG/CR-0497; TREE-1280, Idaho National Engineering Lab, Feb. 1979. http://dx.doi.org/10.2172/6442256.

[114] J. E. Hansel, D. Andrs, R. A. Berry, and R. C. Martineau, "Single-phase compressible junction model for RELAP-7," Tech. Rep. INL/LTD-17-43372, Idaho National Laboratory, 2017.

[115] US NRC, "TRAC-M/FORTRAN 90 (Version 3.0) Theory Manual," Tech. Rep. NUREG/CR-6724, US NRC, 2001.

[116] A. Chaibakhsh and A. Ghaffari, "Steam turbine model," *Simulation Modelling Practice and Theory*, vol. 16, pp. 1145–1162, 2008.

[117] R. T. Layhey, Jr. and F. J. Moody, *The thermal-hydraulics of a boiling water nuclear reactor*. American Nuclear Society, 1993.

[118] S. Levy, *Two-phase flow in complex systems*. John Wiley & Sons, 1999.


INL
Idaho National Laboratory