# INL REPORT

INL/EXT-14-31366 (rev. 1)
Unlimited Release
February 2014 (rev. 1 - March 2015)

# RELAP-7 Theory Manual

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.

Approved for public release; further dissemination unlimited.

Idaho National Laboratory

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

**NOTICE:** This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

INL/EXT-14-31366 (rev. 1)
Unlimited Release
February 2014 (rev. 1 - March 2015)

# RELAP-7 Theory Manual

R. A. Berry, J. W. Peterson, H. Zhang, R. C. Martineau, H. Zhao, L. Zou, D. Andrs

# Contents


Summary . . . . . 10
1 Introduction . . . . . 13
1.1 RELAP-7 Description of Approach . . . . . 14
1.1.1 Software Framework . . . . . 14
1.2 Governing Theory . . . . . 15
1.2.1 7-Equation Two-Phase Model . . . . . 15
1.2.2 Core Heat Transfer and Reactor Kinetics . . . . . 18
1.3 Computational Approach . . . . . 19
2 Single-Phase Thermal Fluids Models . . . . . 22
2.1 Single-Phase Flow Model . . . . . 22
2.1.1 Single-Phase Flow Field Equations . . . . . 24
2.2 Single-Phase Flow Constitutive Models . . . . . 27
2.2.1 Single-Phase Flow Wall Friction Factor Model . . . . . 27
2.2.2 Single-Phase Flow Convective Heat Transfer Model . . . . . 29
2.2.2.1 Internal Pipe Flow . . . . . 31
2.2.2.2 Vertical Bundles with In-line Rods, Parallel Flow Only . . 32
2.2.3 Single-Phase Equations of State . . . . . 32
2.2.3.1 Barotropic Equation of State . . . . . 33
2.2.3.2 Isentropic Stiffened Gas Equation of State . . . . . 34
2.2.3.3 Linear Equation of State . . . . . 35
2.2.3.4 Stiffened Gas Equation of State . . . . . 36
2.2.3.5 Ideal Gas Equation of State . . . . . 40
3 Two-Phase Thermal Fluids Models . . . . . 41
3.1 Seven Equation Two-Phase Flow Model . . . . . 41
3.1.1 Ensemble Averaging . . . . . 43
3.1.2 Seven-Equation Two-Phase Flow Field Equations . . . . . 45
3.1.3 Mass Balance . . . . . 47
3.1.4 Generic Balance Equation . . . . . 48
3.1.5 Species Mass Balance . . . . . 50
3.1.6 Momentum Balance . . . . . 51
3.1.7 Energy Balance . . . . . 51
3.1.8 Entropy Inequality . . . . . 53
3.1.9 Volume Fraction Propagation Equation . . . . . 55
3.1.10 Multi-dimensional Two-Phase Governing Equations . . . . . 58
3.1.11 One-dimensional, Variable Cross-sectional Area, Seven Equation Two-phase Model . . . . . 60

3.2 Seven-Equation Two-Phase Flow Constitutive Models . . . . . 63
3.2.1 Interface Pressure and Velocity, Mechanical Relaxation Coefficients 64
3.2.2 Wall and Interface Direct Heat Transfer . . . . . 66
3.2.3 Interphase Mass Transfer . . . . . 67
3.2.4 Wall and Interphase Friction . . . . . 79
3.2.5 Stiffened Gas Equation of State for Two-phase Flows . . . . . 81
3.3 Homogeneous Equilibrium Two-Phase Flow Model (HEM) . . . . . 83
3.3.1 HEM Field Equations . . . . . 84
3.3.2 HEM Constitutive Models . . . . . 84
4 Heat Conduction Model . . . . . 86
4.1 Heat Conduction Model . . . . . 86
4.2 Material Properties . . . . . 86
4.2.1 Uranium Dioxide . . . . . 87
4.2.2 Zircaloy . . . . . 88
4.2.3 Fuel Rod Gap Gas . . . . . 88
5 Numerical Methods . . . . . 90
5.1 Spatial Discretization Algorithm . . . . . 90
5.2 Time Integration Methods . . . . . 92
5.2.1 Backward Euler . . . . . 92
5.2.2 BDF2 . . . . . 93
5.3 The PCICE Algorithm . . . . . 94
5.3.1 Explicit Predictor . . . . . 96
5.3.2 PCICE Algorithm Temporal Discretization . . . . . 96
5.3.3 Intermediate Momentum Solution . . . . . 97
5.3.4 Pressure Poisson Equation . . . . . 98
5.4 Solution Stabilization Methods . . . . . 100
5.4.1 Streamline Upwind/Petrov-Galerkin Method . . . . . 101
5.4.2 Entropy Viscosity Method . . . . . 105
5.5 Jacobian-Free Newton Krylov Solver . . . . . 114
6 Component Models . . . . . 116
6.1 Pipe . . . . . 116
6.2 PipeWithHeatStructure . . . . . 116
6.3 CoreChannel . . . . . 117
6.4 HeatExchanger . . . . . 117
6.5 Junction/Branch . . . . . 117
6.5.1 Lagrange Multiplier Based Junction Model . . . . . 117
6.5.2 Volume Branch Model . . . . . 118
6.6 Pump . . . . . 119

6.7 Turbine . . . . . 120
6.8 SeparatorDryer . . . . . 124
6.9 DownComer . . . . . 126
6.10 Valves . . . . . 127
6.11 Compressible Valve Models . . . . . 127
6.12 Wet Well Model . . . . . 129
6.13 Time Dependent Volume . . . . . 133
6.14 Time Dependent Junction . . . . . 133
6.15 SubChannel . . . . . 134
6.16 Reactor . . . . . 135
7 Reactor Kinetics Model . . . . . 136
7.1 Point Kinetics Equations . . . . . 136
7.2 Fission Product Decay Model . . . . . 137
7.3 Actinide Decay Model . . . . . 138
7.4 Transformation of Equations for Solution . . . . . 139
7.5 Reactivity Feedback Model . . . . . 140
8 Multi-Dimensional Capability and Interface with RAVEN . . . . . 142
8.1 RattleSnake . . . . . 143
8.2 Bison/MARMOT . . . . . 143
8.3 BigHorn . . . . . 144
8.4 RAVEN . . . . . 144
References . . . . . 146

# Figures


1 Diagram showing the variable-area duct used in the derivation of the governing equations. . . . . 23
2 Interface control volume (top); $T$-$p$ state space around saturation line, $T_{liq} < T_{vap}$, (bottom). . . . . 68
3 Vaporization and condensation at a liquid-vapor interface (after Moody [1]). 70
4 Turbine characteristics (credit of Saravanamuttoo, Rogers, and Cohen [2]). 122
5 $T$-$s$ diagram for a turbine. . . . . 123
6 A simplified wet well model. . . . . 130
7 Multi-physics and multi-dimensional capability coupling for RELAP-7 . . . 142

# Tables


1 Balance Equation Variable Definitions. . . . . 22
2 Constants for Courant and Friedrich's form of the isentropic stiffened gas equation of state. . . . . 35
3 Constants for the linear equation of state for $p_0 = 1$ MPa and $T_0 = 375$, 400, 425, and 450K. . . . . 37
4 Constants for the linear equation of state for $p_0 = 5$ MPa and $T_0 = 375$, 400, 425, and 450K. . . . . 38
5 Stiffened gas equation of state parameters for water and its vapor, from [3]. 39
6 State variable definitions. . . . . 45
7 Multiphase flow ensemble averages of interest. . . . . 46
8 Zircaloy thermal conductivity parameters. . . . . 88

# 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 capability and extends the analysis capability 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 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 are being carried out in order to demonstrate more refined BWR SBO analyses with more realistic geometries.

In summary, the MOOSE based RELAP-7 code development is a new 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) [4]. 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 [5] capability 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 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 [6, 7] 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 [8], 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 [9]) project or the Trilinos project [10] (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 LWRSs 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 [11], 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 [12–14]. 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 [15]. The reactor structure is coupled with the thermal fluid through energy exchange (conjugate heat transfer) employing surface convective heat transfer [16] 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 [17] 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 [8] 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 [18] (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 [19, 20] 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 [21]. If successfully integrated into the MOOSE framework, this slow transient capability would be available to 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 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.

2. 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.

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 [21]. 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.

# 2 Single-Phase Thermal Fluids Models

## 2.1 Single-Phase 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 $\rho$ is the local material mass density, $\boldsymbol{u}$ is the local material velocity, and $\psi$, $\boldsymbol{J}$, and $\phi$ 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** | $\psi$ | $\boldsymbol{J}$ | $\phi$ |
|---|---|---|---|
| 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

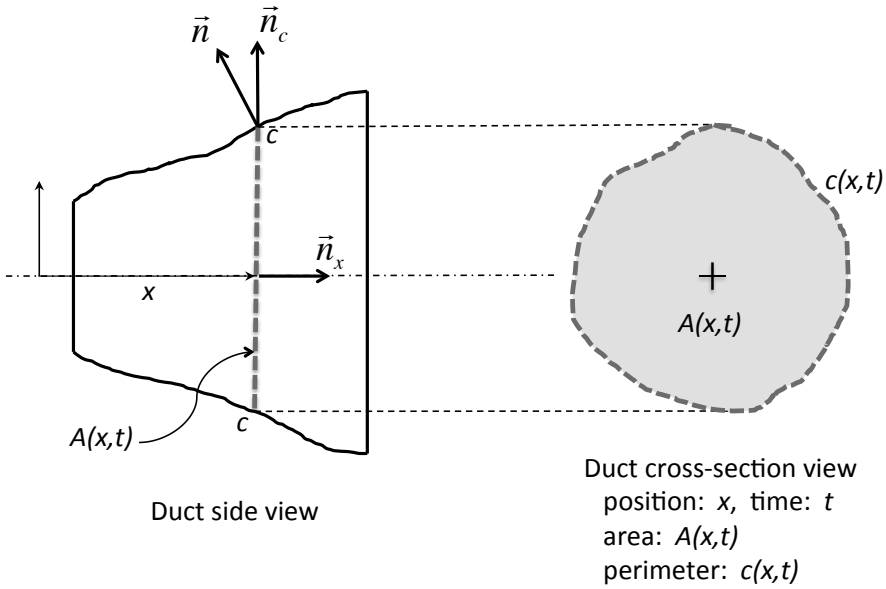


**Figure 1.** Diagram showing the variable-area duct used in the derivation of the governing equations.

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 Single-Phase Flow 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

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

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

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

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

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

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 $\Delta$ 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

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

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

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

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

where the $F_{\text{wall friction}}$ is the average duct wall shear force (friction), $Q_{\text{wall}}$ is the average heat transfer rate *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 Single-Phase Flow Constitutive Models

### 2.2.1 Single-Phase Flow Wall Friction Factor Model

The wall friction term in (21) takes the general form

$$F_{\text{wall friction}} = \frac{f}{2d_h}\rho u\,|u|\,A \tag{24}$$

where $f$ is the (Darcy) friction factor, and $d_h$ is the hydraulic diameter, defined as

$$d_h = \frac{4A}{P_{\text{wet}}} \tag{25}$$

and $P_{\text{wet}}$ is the so-called wetted perimeter of the pipe, which is defined as the "perimeter of the cross-sectional area that is wet." More accurately, it is that portion of the perimeter of the cross-sectional area for which a wall-shear stress exists. 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

$$d_h = 2r(x) = 2\sqrt{\frac{A}{\pi}} \tag{26}$$

so that (24) becomes

$$F_{\text{wall friction}} = \frac{f}{4}\rho u\,|u|\,\sqrt{\pi A} \tag{27}$$

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 [22]. 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

$$Re = \frac{\rho |u| d_h}{\mu} \tag{28}$$

where $\mu$ 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

$$f = \frac{64}{Re \Phi_S}, \qquad 64 \leq Re < 2200 \tag{29}$$

where $\Phi_S$ 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 (29), 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

$$\frac{N}{Re_{\text{min}}} - \frac{N}{Re_{\text{max}}} = 1 \tag{30}$$

yields

$$N = \frac{Re_{\text{max}} Re_{\text{min}}}{Re_{\text{max}} - Re_{\text{min}}}. \tag{31}$$

The reciprocal weighting function $w$ is then defined as

$$w = \frac{N}{Re_{\text{min}}} - \frac{N}{Re} \tag{32}$$

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

$$f = (1 - w) f_{\text{lam},Re_{\text{min}}} + w f_{\text{turb},Re_{\text{max}}}. \tag{33}$$

Formula (33) is valid for $2200 \leq Re \leq 3000$, $f_{\text{lam},Re_{\text{min}}}$ is the laminar friction factor at $Re_{\text{min}}$, and $f_{\text{turb},Re_{\text{max}}}$ is the turbulent friction factor at $Re_{\text{max}}$. The turbulent friction factor is

given by a Zigrang-Sylvester approximation [23] to the Colebrook-White correlation [24], for $Re \geq 3000$:

$$\frac{1}{\sqrt{f}} = -2\log_{10}\left\{\frac{\epsilon}{3.7D} + \frac{2.51}{Re}\left[1.14 - 2\log_{10}\left(\frac{\epsilon}{D} + \frac{21.25}{Re^{0.9}}\right)\right]\right\} \tag{34}$$

where $\epsilon$ 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 Single-Phase Flow Convective Heat Transfer Model

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

$$Q_{\text{wall}} = H_w a_w \left(T_{\text{wall}} - T\right) A \tag{35}$$

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:

$$a_w \equiv \lim_{\Delta x \to 0} \frac{\text{wetted area of pipe section of length } \Delta x}{\text{volume of pipe section of length } \Delta x} \tag{36}$$

For a constant-area pipe with radius $r$ and circular cross-section, formula (36) yields

$$a_w = \lim_{\Delta x \to 0} \frac{2\pi r \Delta x}{\pi r^2 \Delta x} = \frac{2}{r} \tag{37}$$

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 (35) we can set

$$a_w A \Delta x \equiv \text{“projected area of a pipe segment of length } \Delta x\text{”} \tag{38}$$

and then take the limit as $\Delta x \to 0$. The right-hand side of (38) 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 $\Delta x$ implies that (38) can be written as:

$$a_w A \Delta x = \pi \left(2r + \frac{\partial r}{\partial x}\Delta x\right) \Delta x \sqrt{1 + \left(\frac{\partial r}{\partial x}\right)^2} \tag{39}$$

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

$$a_w A = 2\pi r \sqrt{1 + \left(\frac{\partial r}{\partial x}\right)^2} = \sqrt{4\pi A + \left(\frac{\partial A}{\partial x}\right)^2} \tag{40}$$

where (40) arises upon substitution of the cross-sectional area formula for a circle. Note also that we recover

$$a_w A = 2\pi r \tag{41}$$

from (40) in the constant area case. The resulting wall heating term in this case is

$$Q_{\text{wall}} = H_w \left(T - T_{\text{wall}}\right) \left[4\pi A + \left(\frac{\partial A}{\partial x}\right)^2\right]^{\frac{1}{2}} \tag{42}$$

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 (40) for polygonal cross sections other than circles. For example, for a square cross section with side length $L(x)$, the analog of (39) is

$$a_w A \Delta x = 2\left(2L + \frac{\partial L}{\partial x}\Delta x\right)\Delta x \sqrt{1 + \frac{1}{4}\left(\frac{\partial L}{\partial x}\right)^2} \tag{43}$$

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

$$a_w A = 4L \sqrt{1 + \frac{1}{4}\left(\frac{\partial L}{\partial x}\right)^2} = \sqrt{16A + \left(\frac{\partial A}{\partial x}\right)^2} \tag{44}$$

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 [25]. The convective heat transfer coefficient is determined by many factors, i.e., hydraulic geometry, fluid types, and several Buckingham $\pi$-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.

In RELAP5, many different hydraulic geometries are included, but they can be divided into two basic types: internal and external. Internal flow geometries include different shapes of pipes, parallel plates, annuli, and spheres; external flow geometries include single tube, single plate, tube bundles, and spheres. Each geometry may have different flow directions, such as vertical, horizontal, with/without cross flow, and helical. To help users communicate the flow field geometry types, RELAP5 uses a numbering system. RELAP-7 follows the same numbering system. Currently, only the two most commonly used geometries are included in RELAP-7: "101," the default geometry, for internal pipe flow, and "110," for a bundle of in-line rods with parallel flow only.

#### 2.2.2.1 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

$$Nu = \frac{H_w d_h}{k} = 4.36 \tag{45}$$

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

$$Nu = CRe^{0.8}Pr^n \tag{46}$$

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 [25]. The Churchill and Chu $Nu$-correlation,

$$Nu = \left(0.825 + \frac{0.387Ra^{\frac{1}{6}}}{\left(1 + \left(\frac{0.492}{Pr}\right)^{\frac{9}{16}}\right)^{\frac{8}{27}}}\right)^2 \tag{47}$$

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

$$Gr = \frac{\rho^2 g\beta(T_w - T)L^3}{\mu^2} \tag{48}$$

where $\beta$ 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:

$$Nu = 5.0 + 0.025 Pe^{0.8} \tag{49}$$

where $Pe = RePr$ is the Peclet number.

#### 2.2.2.2 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 [26], 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 (46) becomes

$$C = 0.023 \frac{P}{D} \tag{50}$$

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

$$Nu = 4.0 + 0.33 \left(\frac{P}{D}\right)^{3.8} \left(\frac{Pe}{100}\right)^{0.86} + 0.16 \left(\frac{P}{D}\right)^{5}. \tag{51}$$

Equation (51) 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.

### 2.2.3 Single-Phase 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

$$p = p(\rho, e) \tag{52}$$

$$T = T(\rho, e) \tag{53}$$

i.e., both the pressure and the temperature can be computed if the density and internal energy are given. Reformulations of (52) and (53) which consist of two equations relating the four quantities $p$, $T$, $\rho$, and $e$ are also acceptable and useful in practice.

The pair of equations (52) and (53) 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 [27]. 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 (52)–(53) in the present work. More will be said subsequently, when discussing selection and stabilization of "weak" solutions.

#### 2.2.3.1 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

$$\begin{aligned} p &= p_0 + a^2(\rho - \rho_0) \\ &= p_0 + a^2(U_0 - \rho_0) \end{aligned} \tag{54}$$

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

$$p_{,0} = a^2 \tag{55}$$

$$p_{,1} = 0 \tag{56}$$

$$p_{,2} = 0 \tag{57}$$

#### 2.2.3.2 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:

$$\frac{p + p_\infty}{p_0 + p_\infty} = \left(\frac{\rho}{\rho_0}\right)^\gamma \tag{58}$$

which is sometimes rearranged to read:

$$p = (p_0 + p_\infty)\left(\frac{\rho}{\rho_0}\right)^\gamma - p_\infty \tag{59}$$

where $p_\infty$, $\gamma$, and $\rho_0$ 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 $\gamma$ is used in (59), 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

$$c^2 = \frac{\partial p}{\partial \rho} = \frac{\gamma}{\rho}(p + p_\infty) \tag{60}$$

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:

$$p = (p_0 + p_\infty)\left(\frac{U_0}{\rho_0}\right)^\gamma - p_\infty \tag{61}$$

with derivatives

$$p_{,0} = \frac{\gamma}{U_0}(p + p_\infty) = c^2 \tag{62}$$

$$p_{,1} = 0 \tag{63}$$

$$p_{,2} = 0 \tag{64}$$

Finally, we note that Courant and Friedrichs [28] also discuss this equation of state in the form

$$p = A\left(\frac{\rho}{\rho_0}\right)^\gamma - B \tag{65}$$

Approximate values for the constants in (65) are given in Table 2.

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

| | SI | Imperial |
|---|---|---|
| $\rho_0$ | 999.8 kg/m$^3$ | 1.94 slug/ft$^3$ |
| $\gamma$ | 7 | |
| $A$ | $3.04076 \times 10^8$ Pa | 3001 atm |
| $B$ | $3.03975 \times 10^8$ Pa | 3000 atm |

#### 2.2.3.3 Linear Equation of State

A more general "linear" equation of state (a straightforward extension of (54)) which takes into account variations in temperature as well as density, is given by

$$p = p_0 + K_\rho(\rho - \rho_0) + K_T(T - T_0) \tag{66}$$

$$e = e_0 + c_v(T - T_0). \tag{67}$$

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 (66) and (67) 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, (66) and (67) can be written as

$$p = p_0 + K_\rho(U_0 - \rho_0) + \frac{K_T}{c_v}\left(\frac{U_2}{U_0} - \frac{U_1^2}{2U_0^2} - e_0\right) \tag{68}$$

$$T = T_0 + \frac{1}{c_v}\left(\frac{U_2}{U_0} - \frac{U_1^2}{2U_0^2} - e_0\right), \tag{69}$$

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

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

$$p_{,0} = K_\rho + \frac{K_T}{c_v U_0}\left(\frac{U_1^2}{U_0^2} - \frac{U_2}{U_0}\right) = K_\rho + \frac{K_T}{c_v \rho}\left(u^2 - E\right) \tag{70}$$

$$p_{,1} = -\frac{K_T U_1}{c_v U_0^2} = -\frac{K_T u}{c_v \rho} \tag{71}$$

$$p_{,2} = \frac{K_T}{c_v U_0} = \frac{K_T}{c_v \rho}\ . \tag{72}$$

The derivatives of $T$ with respect to the conserved variables are

$$T_{,0} = \frac{1}{c_v U_0}\left(\frac{U_1^2}{U_0^2} - \frac{U_2}{U_0}\right) = \frac{1}{c_v \rho}\left(u^2 - E\right) \tag{73}$$

$$T_{,1} = -\frac{U_1}{c_v U_0^2} = -\frac{u}{c_v \rho} \tag{74}$$

$$T_{,2} = \frac{1}{c_v U_0} = \frac{1}{c_v \rho}\ . \tag{75}$$

For completeness, the density is given as a function of pressure and temperature, and the temperature as a function of pressure and density, for the linear equation of state:

$$\rho = \rho_0 + \frac{p - p_0}{K_\rho} - \frac{K_T}{K_\rho}(T - T_0) \tag{76}$$

$$T = T_0 + \frac{p - p_0}{K_T} - \frac{K_\rho}{K_T}(\rho - \rho_0)\,. \tag{77}$$

#### 2.2.3.4 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. [3])

$$p(\rho, e) = (\gamma - 1)\rho(e - q) - \gamma p_\infty \tag{78}$$

where $p$, $\rho$, $e$, and $q$ are the pressure, density, internal energy, and the binding energy of the fluid considered. The parameters $\gamma$, $q$, and $p_\infty$ are the constants (coefficients) of each fluid. The parameter $q$ defines the zero point for the internal energy, which will be

**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 |
| $K_\rho$ | $2.1202 \times 10^6$ | Pa-m$^3$/kg |
| $\rho_0$ | 957.43 | kg/m$^3$ |
| $K_T$ | $1.5394 \times 10^6$ | Pa/K |
| $T_0$ | 375 | K |
| $c_v$ | $4.22 \times 10^3$ | J/kg-K |
| $e_0$ | $4.27 \times 10^5$ | J/kg |

| $T = 400$K | | |
|---|---|---|
| $p_0$ | $10^6$ | Pa |
| $K_\rho$ | $1.9474 \times 10^6$ | Pa-m$^3$/kg |
| $\rho_0$ | 937.87 | kg/m$^3$ |
| $K_T$ | $1.6497 \times 10^6$ | Pa/K |
| $T_0$ | 400 | K |
| $c_v$ | $4.22 \times 10^3$ | J/kg-K |
| $e_0$ | $5.32 \times 10^5$ | J/kg |

| $T = 425$K | | |
|---|---|---|
| $p_0$ | $10^6$ | Pa |
| $K_\rho$ | $1.7702 \times 10^6$ | Pa-m$^3$/kg |
| $\rho_0$ | 915.56 | kg/m$^3$ |
| $K_T$ | $1.6643 \times 10^6$ | Pa/K |
| $T_0$ | 425 | K |
| $c_v$ | $4.22 \times 10^3$ | J/kg-K |
| $e_0$ | $6.39 \times 10^5$ | J/kg |

| $T = 450$K | | |
|---|---|---|
| $p_0$ | $10^6$ | Pa |
| $K_\rho$ | $1.5552 \times 10^6$ | Pa-m$^3$/kg |
| $\rho_0$ | 890.39 | kg/m$^3$ |
| $K_T$ | $1.6303 \times 10^6$ | Pa/K |
| $T_0$ | 450 | K |
| $c_v$ | $4.22 \times 10^3$ | J/kg-K |
| $e_0$ | $7.48 \times 10^5$ | J/kg |

relevant later when phase transitions are involved with two-phase flows. The parameter $p_\infty$ 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 (78) 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. [3], 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.

**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 | | |
|---|---|---|
| $p_0$ | $5 \times 10^6$ | Pa |
| $K_\rho$ | $2.1202 \times 10^6$ | Pa-m$^3$/kg |
| $\rho_0$ | 959.31 | kg/m$^3$ |
| $K_T$ | $1.5559 \times 10^6$ | Pa/K |
| $T_0$ | 375 | K |
| $c_v$ | $4.26 \times 10^3$ | J/kg-K |
| $e_0$ | $4.25 \times 10^5$ | J/kg |

| $T = 400$K | | |
|---|---|---|
| $p_0$ | $5 \times 10^6$ | Pa |
| $K_\rho$ | $1.9474 \times 10^6$ | Pa-m$^3$/kg |
| $\rho_0$ | 939.91 | kg/m$^3$ |
| $K_T$ | $1.6406 \times 10^6$ | Pa/K |
| $T_0$ | 400 | K |
| $c_v$ | $4.26 \times 10^3$ | J/kg-K |
| $e_0$ | $5.31 \times 10^5$ | J/kg |

| $T = 425$K | | |
|---|---|---|
| $p_0$ | $5 \times 10^6$ | Pa |
| $K_\rho$ | $1.7702 \times 10^6$ | Pa-m$^3$/kg |
| $\rho_0$ | 917.83 | kg/m$^3$ |
| $K_T$ | $1.6659 \times 10^6$ | Pa/K |
| $T_0$ | 425 | K |
| $c_v$ | $4.26 \times 10^3$ | J/kg-K |
| $e_0$ | $6.37 \times 10^5$ | J/kg |

| $T = 450$K | | |
|---|---|---|
| $p_0$ | $5 \times 10^6$ | Pa |
| $K_\rho$ | $1.5552 \times 10^6$ | Pa-m$^3$/kg |
| $\rho_0$ | 892.99 | kg/m$^3$ |
| $K_T$ | $1.6370 \times 10^6$ | Pa/K |
| $T_0$ | 450 | K |
| $c_v$ | $4.26 \times 10^3$ | J/kg-K |
| $e_0$ | $7.46 \times 10^5$ | J/kg |

The pressure law, equation (78), 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:

$$e(p, \rho) = \frac{p + \gamma p_\infty}{(\gamma - 1)\rho} + q \tag{79}$$

$$\rho(p, T) = \frac{p + p_\infty}{(\gamma - 1)c_v T} \tag{80}$$

$$h(T) = \gamma\, c_v T + q \tag{81}$$

$$g(p, T) = (\gamma c_v - q')T - c_v T \ln \frac{T^\gamma}{(p + p_\infty)^{(\gamma-1)}} + q \tag{82}$$

where $T$, $h$, and $g$ are the temperature, enthalpy, and Gibbs free enthalpy, respectively, of the fluid considered. In this system, equation (80) 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 [3] are given in Table 5. These parameter values appear to yield reasonable approximations over a temperature range from 298 to 473 K [3]. Equation (81) can also

**Table 5.** Stiffened gas equation of state parameters for water and its vapor, from [3].

| **Water** | $\gamma$ | $q$ (J kg$^{-1}$) | $q'$ (J kg$^{-1}$ K$^{-1}$) | $p_\infty$ (Pa) | $c_v$ (J kg$^{-1}$ K$^{-1}$) |
|---|---|---|---|---|---|
| Liquid | 2.35 | $-1167 \times 10^3$ | 0 | $10^9$ | 1816 |
| Vapor | 1.43 | $2030 \times 10^3$ | $-23 \times 10^3$ | 0 | 1040 |

be written as

$$h = c_p T + q \tag{83}$$

if we define $c_p = \gamma c_v$. Combining (79) and (80) also allows us to write the temperature as

$$T = \frac{1}{c_v}\left(e - q - \frac{p_\infty}{\rho}\right). \tag{84}$$

In terms of conserved variables, the pressure is given by

$$p = (\gamma - 1)\left(U_2 - \frac{U_1^2}{2U_0} - U_0 q\right) - \gamma p_\infty. \tag{85}$$

The derivatives of $p$ with respect to the conserved variables are

$$p_{,0} = (\gamma - 1)\left(\frac{1}{2}\frac{U_1^2}{U_0^2} - q\right) = (\gamma - 1)\left(\frac{1}{2}u^2 - q\right) \tag{86}$$

$$p_{,1} = (\gamma - 1)\left(-\frac{U_1}{U_0}\right) = (\gamma - 1)(-u) \tag{87}$$

$$p_{,2} = \gamma - 1. \tag{88}$$

In terms of conserved variables, the temperature is given by

$$T = \frac{1}{c_v}\left(\frac{U_2}{U_0} - \frac{U_1^2}{2U_0^2} - q - \frac{p_\infty}{U_0}\right). \tag{89}$$

The derivatives of $T$ with respect to the conserved variables are

$$T_{,0} = \frac{1}{c_v U_0^2}\left(p_\infty + \frac{U_1^2}{U_0} - U_2\right) = \frac{1}{c_v \rho^2}\left(p_\infty + \rho u^2 - \rho E\right) \tag{90}$$

$$T_{,1} = -\frac{U_1}{c_v U_0^2} = -\frac{u}{c_v \rho} \tag{91}$$

$$T_{,2} = \frac{1}{c_v U_0} = \frac{1}{c_v \rho}\ . \tag{92}$$

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

$$\begin{aligned} c^2 &= \frac{p}{\rho^2}(\gamma - 1)\rho + (\gamma - 1)(e - q) \\ &= \gamma\left(\frac{p + p_\infty}{\rho}\right). \end{aligned} \tag{93}$$

#### 2.2.3.5 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

$$p = (\gamma - 1)\rho e \tag{94}$$

$$T = \frac{e}{c_v} \tag{95}$$

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 2.2.3.4, 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 2.2.3.4, and the derivatives listed therein.

# 3 Two-Phase Thermal Fluids Models

## 3.1 Seven Equation Two-Phase 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" [29,30] and "time averaging" [31]. 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 [32], 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 repre-

sentative 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 [33].

### 3.1.1 Ensemble Averaging

A general method is presented here, based on the ensemble averaging concept [33–37] 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

$$\langle Q_0 \rangle(\boldsymbol{x}, t) = \frac{1}{N_R} \sum_{r=1}^{N_R} Q_{0,r}(\boldsymbol{x}, t) \tag{96}$$

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 $N_R$

$$\begin{aligned}\langle Q_0\rangle(\boldsymbol{x},t) &= \frac{1}{N_R}\sum_{r=1}^{N_R} N(\boldsymbol{x},t,\Gamma)Q_0(\Gamma)\\ &= \sum_{r=1}^{N_\Gamma}\frac{N(\boldsymbol{x},t,\Gamma)}{N_R}Q_0(\Gamma)\\ &= \int_{\text{all }\Gamma} Q_0(\Gamma)f(\boldsymbol{x},t,\Gamma)\ \mathrm{d}\Gamma \end{aligned} \tag{97}$$

where $f(\boldsymbol{x},t,\Gamma) = \frac{N(\boldsymbol{x},t,\Gamma)}{N_R}$ is the probability of the state $\Gamma$ 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

$$\Gamma = \begin{bmatrix} \rho_0, \boldsymbol{u}_0, h_0, p_0, \tau_0, \ldots \\ \rho_0^1, \boldsymbol{u}_0^1, h_0^1, \rho_0^2, \boldsymbol{u}_0^2, h_0^2, \ldots \\ X_1, X_2, \ldots \end{bmatrix} \tag{98}$$

where the various symbols used in (98) are described in Table 6, and

$$\rho_0 = \sum_s \rho_0^s \tag{99}$$

$$\rho_0\boldsymbol{u}_0 = \sum_s \rho_0^s\boldsymbol{u}_0^s \tag{100}$$

$$\rho_0 h_0 = \sum_s \rho_0^s h_0^s\ . \tag{101}$$

Other properties may also appear in the above thermodynamic/kinematic state such as the phase or material temperature, $\theta_0$, the phase or material specific internal energy, $e_0$, and the phase or material specific entropy, $s_0$.

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

**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 |
| $\rho_0$ | 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 |
| $\rho_0^s$ | Species partial density |
| $\boldsymbol{u}_0^s$ | Species velocity |
| $h_0^s$ | Species partial enthalpy |

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 [34], while others prefer working with intrinsic densities, e.g. Drew and Passman [33]. 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.

### 3.1.2 Seven-Equation Two-Phase Flow Field Equations

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

$$\dot{\rho}_0 = -\rho_0 \nabla \cdot \boldsymbol{u}_0 \tag{102}$$

$$\dot{\rho}_0^s = -\rho_0^s \nabla \cdot \boldsymbol{u}_0 - \nabla \cdot \rho_0^s (\boldsymbol{u}_0^s - \boldsymbol{u}_0) + \dot{r}_0^s \tag{103}$$

$$\rho_0 \dot{\boldsymbol{u}}_0 = \nabla \cdot \boldsymbol{T}_0 + \rho_0 \boldsymbol{g} \tag{104}$$

$$\rho_0 \dot{E}_0 = \nabla \cdot (\boldsymbol{T}_0 \cdot \boldsymbol{u}_0) + \nabla \cdot \boldsymbol{q}_0 + \rho_0 \boldsymbol{g} \cdot \boldsymbol{u}_0 + \rho_0 \varepsilon_0 \tag{105}$$

$$\rho_0 \dot{s}_0 \geqslant \frac{\rho_0 \varepsilon_0}{\theta_0} - \nabla \cdot \left( \frac{\boldsymbol{q}_0}{\theta_0} \right). \tag{106}$$

**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 |

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

$$\dot{Q}_0 \equiv \frac{\partial Q_0}{\partial t} + \boldsymbol{u}_0 \cdot \nabla Q_0 \ . \tag{107}$$

Let the total variation of $f$ in the phase space $(\boldsymbol{x}, t, \Gamma)$ be given by [34]

$$\frac{\partial f}{\partial t} + \boldsymbol{u}_0 \cdot \nabla f + \dot{\Gamma} \frac{\partial f}{\partial \Gamma} = \frac{df}{dt} = 0 \tag{108}$$

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 [34], here corrected) that the resulting equation is

$$\frac{\partial}{\partial t} \langle Q_0 \rangle + \nabla \cdot \langle Q_0 \boldsymbol{u}_0 \rangle = \langle \dot{Q}_0 + Q_0 \nabla \cdot \boldsymbol{u}_0 \rangle \ . \tag{109}$$

This result is called the *moment evolution equation* and the details of its derivation are given in [13, 14]. 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 (109) results in

$$\frac{\partial \langle X_k \rho_0 \rangle}{\partial t} + \nabla \cdot \langle X_k \rho_0 \boldsymbol{u}_0 \rangle = \left\langle \dot{X}_k \rho_0 + X_k (\dot{\rho}_0 + \rho_0 \nabla \cdot \boldsymbol{u}_0) \right\rangle . \tag{110}$$

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

$$\frac{\partial \alpha_k \rho_k}{\partial t} + \nabla \cdot \alpha_k \rho_k \boldsymbol{u}_k = \left\langle \dot{X}_k \rho_0 \right\rangle . \tag{111}$$

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 [38]. 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

$$\dot{X}_k = \frac{\partial X_k}{\partial t} + \boldsymbol{u}_0 \cdot \nabla X_k \tag{112}$$

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,

$$\frac{\partial X_k}{\partial t} + \boldsymbol{u}_{int} \cdot \nabla X_k = 0 \tag{113}$$

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

$$\left\langle \dot{X}_k \rho_0 \right\rangle = \langle \rho_0 (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) \cdot \nabla X_k \rangle \tag{114}$$

and the averaged mass balance equation becomes

$$\begin{aligned} \frac{\partial \alpha_k \rho_k}{\partial t} + \nabla \cdot \alpha_k \rho_k \boldsymbol{u}_k &= \langle \rho_0 (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) \cdot \nabla X_k \rangle \\ &\equiv \Omega_k^{mass} \ . \end{aligned} \tag{115}$$

Because $\nabla X_k$ 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 [39, 40], $\nabla X_k$ 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 $\Omega_k^{mass}$ 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

$$\sum_{k=1}^{no.\,of\,phases} \Omega_k^{mass} = 0 \ . \tag{116}$$

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

$$A_k = -\left\langle \hat{\boldsymbol{n}}_k \cdot \nabla X_k \right\rangle \tag{117}$$

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

$$\frac{\partial \rho_0 \psi_0}{\partial t} + \nabla \cdot \rho_0 \psi_0 \boldsymbol{u}_0 = \nabla \cdot J_0 + \rho_0 g_0 \tag{118}$$

or

$$\rho_0 \dot{\psi}_0 = \frac{d(\rho_0 \psi_0)}{dt} + (\rho_0 \psi_0) \nabla \cdot \boldsymbol{u}_0 = \nabla \cdot J_0 + \rho_0 g_0 \ . \tag{119}$$

Equations (118) and (119) hold at each point where sufficient smoothness occurs for the derivatives to be taken, otherwise at simple discontinuities its generic jump balance condition

$$[\![\rho_0 \psi_0 (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) + J_0]\!] \cdot \hat{\boldsymbol{n}} = m \tag{120}$$

holds, where $\psi_0$ 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 $\psi_0$. 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.

$$\Gamma = \begin{bmatrix} \rho_0, \boldsymbol{u}_0, \psi_0, J_0, \ldots \\ X_1, X_2, \ldots \end{bmatrix} \ . \tag{121}$$

Let us also define averages of these quantities as

$$\psi_k \equiv \frac{\langle X_k \rho_0 \psi_0 \rangle}{\alpha_k \rho_k} \tag{122}$$

$$J_k \equiv \frac{\langle X_k J_0 \rangle}{\alpha_k} \tag{123}$$

$$g_k \equiv \frac{\langle X_k \rho_0 g_0 \rangle}{\alpha_k \rho_k} \ . \tag{124}$$

Letting $Q_0 = X_k \rho_0 \psi_0$ in (109) gives

$$\begin{aligned} \frac{\partial \langle X_k \rho_0 \psi_0 \rangle}{\partial t} + \nabla \cdot \langle X_k \rho_0 \psi_0 \boldsymbol{u}_0 \rangle &= \nabla \cdot \langle X_k J_0 \rangle + \langle X_k \rho_0 g_0 \rangle \\ &\quad + \langle [\rho_0 \psi_0 (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) - J_0] \cdot \nabla X_k \rangle \ . \end{aligned} \tag{125}$$

Introducing the fluctuating velocity

$$\boldsymbol{u}'_k \equiv \boldsymbol{u}_0 - \boldsymbol{u}_k \tag{126}$$

into this expression finally results in

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

where $\boldsymbol{J}_k^{Fluct} = -\frac{\langle X_k \rho_0 \psi_0 \boldsymbol{u}'_k \rangle}{\alpha_k}$ is the flux of $\psi$ due to fluctuations in the phase $k$ velocity, $\psi_k^{int}$ is the effective value of $\psi$ that is transferred to phase $k$ from the other phases due to mass transfer, or phase change, and $\Omega_k^{\psi}$ is a flux of $\psi$ 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

$$\sum_{k=1}^{no.\,of\,phases} \Omega_k^{mass} \psi_k^{int} + \Omega_k^{\psi} = M \tag{127}$$

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

$$\frac{\partial \rho_0^s}{\partial t} + \nabla \cdot \rho_0^s \boldsymbol{u}_0^s = \dot{r}^s \tag{128}$$

where $\rho_0^s$ 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

$$\frac{\partial \rho_0^s}{\partial t} + \nabla \cdot \rho_0^s \boldsymbol{u}_0 = \nabla \cdot \rho_0^s (\boldsymbol{u}_0 - \boldsymbol{u}_0^s) + \dot{r}^s \tag{129}$$

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 (118) with the assignments of

$$\psi_0 = \frac{\rho_0^s}{\rho_0}, \qquad J_0 = \rho_0 \frac{\rho_0^s}{\rho_0} (\boldsymbol{u}_0 - \boldsymbol{u}_0^s), \qquad g_0 = \frac{\dot{r}^s}{\rho_0} \ . \tag{130}$$

Thus the averaged species mass balance equation takes the form

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

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

$$\begin{aligned} \frac{\partial \alpha_k \rho_k^s}{\partial t} + \nabla \cdot \alpha_k \rho_k^s \boldsymbol{u}_k = & \nabla \cdot \langle X_k \rho_0^s (\boldsymbol{u}_0 - \boldsymbol{u}_0^s) \rangle \\ & - \nabla \cdot \langle X_k \rho_0^s \boldsymbol{u}_k' \rangle \\ & + \langle \rho_0^s (\boldsymbol{u}_0 - \boldsymbol{u}_{int}) \cdot \nabla X_k \rangle \\ & - \langle \rho_0^s (\boldsymbol{u}_0 - \boldsymbol{u}_0^s) \cdot \nabla X_k \rangle \\ & + \dot{R}_k^s \end{aligned} \tag{132}$$

where the terms on the right-hand side of (132) 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

$$\psi_0 = \boldsymbol{u}_0, \qquad J_0 = \boldsymbol{T}_0, \qquad g_0 = \boldsymbol{g}_0 \tag{133}$$

to give:

$$\begin{aligned}\frac{\partial \alpha_k \rho_k \boldsymbol{u}_k}{\partial t} + \nabla \cdot \alpha_k \rho_k \boldsymbol{u}_k \otimes \boldsymbol{u}_k = \nabla \cdot \alpha_k (\boldsymbol{T}_k + \boldsymbol{T}_k^{Fluct}) + \alpha_k \rho_k \boldsymbol{g}_k \\ + \boldsymbol{\Omega}_k^{mom} + \boldsymbol{u}_k^{int} \Omega_k^{mass}\end{aligned} \tag{134}$$

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

$$\boldsymbol{T}_k^{Fluct} \equiv -\frac{\langle X_k \rho_0 \boldsymbol{u}_k' \otimes \boldsymbol{u}_k' \rangle}{\alpha_k} \tag{135}$$

$$\boldsymbol{\Omega}_k^{mom} \equiv -\langle \boldsymbol{T}_0 \cdot \nabla X_k \rangle \ . \tag{136}$$

The averaged interfacial momentum balance constraint (jump condition) is

$$\boldsymbol{\gamma} = \sum_{k=1}^{no.\,of\,phases} \boldsymbol{\Omega}_k^{mom} + \boldsymbol{u}_k^{int} \Omega_k^{mass} \tag{137}$$

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

### 3.1.7 Energy Balance

The assignment of

$$\psi_0 = E_0 = e_0 + \frac{1}{2} \boldsymbol{u}_0 \cdot \boldsymbol{u}_0 \tag{138}$$

$$J_0 = \boldsymbol{T_0} \cdot \boldsymbol{u}_0 + \boldsymbol{q}_0 \tag{139}$$

$$g_0 = \boldsymbol{g}_0 \cdot \boldsymbol{u}_0 + \varepsilon_0 \tag{140}$$

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

$$\begin{aligned}
&\frac{\partial}{\partial t}\alpha_k\rho_k\left(e_k+\frac{1}{2}\boldsymbol{u}_k\cdot\boldsymbol{u}_k+e_k^{Fluct}\right)+\nabla\cdot\alpha_k\rho_k\boldsymbol{u}_k\left(e_k+\frac{1}{2}\boldsymbol{u}_k\cdot\boldsymbol{u}_k+e_k^{Fluct}\right)\\
&\quad=\nabla\cdot[\alpha_k(\boldsymbol{T}_k+\boldsymbol{T}_k^{Fluct})\cdot\boldsymbol{u}_k]\\
&\quad-\nabla\cdot\alpha_k(\boldsymbol{q}_k+\boldsymbol{q}_k^{Fluct})+\alpha_k\rho_k(\varepsilon_k+\boldsymbol{g}_k\cdot\boldsymbol{u}_k)\\
&\quad+\Omega_k^{energy}+\boldsymbol{\Omega}_k^{mom}\cdot\boldsymbol{u}_k^{int}+\Omega_k^{mass}\left(e_k^{int}+\frac{1}{2}\boldsymbol{u}_k^{int}\cdot\boldsymbol{u}_k^{int}\right)
\end{aligned}\tag{141}$$

where

$$e_k^{Fluct}\equiv\frac{1}{2}\frac{\langle X_k\rho_0\boldsymbol{u}_k'\cdot\boldsymbol{u}_k'\rangle}{\alpha_k\rho_k}\tag{142}$$

is the fluctuation kinetic energy,

$$\boldsymbol{q}_k^{Fluct}\equiv\frac{\langle X_k\rho_0\boldsymbol{u}_k'e_k'\rangle}{\alpha_k}+\frac{\langle X_k\boldsymbol{T}_0\cdot\boldsymbol{u}_k'\rangle}{\alpha_k}+\frac{1}{2}\frac{\langle X_k\rho_0\boldsymbol{u}_k'(\boldsymbol{u}_k'\cdot\boldsymbol{u}_k')\rangle}{\alpha_k}\tag{143}$$

is the fluctuation energy flux,

$$\varepsilon_k\equiv\frac{\langle X_k\rho_0\varepsilon_0\rangle}{\alpha_k\rho_k}\tag{144}$$

is the energy source,

$$\Omega_k^{energy}\equiv\langle\boldsymbol{q}_0\cdot\nabla X_k\rangle\tag{145}$$

is the interfacial heat source, and

$$\boldsymbol{\Omega}_k^{mom}\cdot\boldsymbol{u}_k^{int}\equiv-\langle\boldsymbol{T}_0\cdot\boldsymbol{u}_0\cdot\nabla X_k\rangle\tag{146}$$

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

$$\sum_{k=1}^{no.\,of\,phases}\Omega_k^{energy}+\boldsymbol{\Omega}_k^{mom}\cdot\boldsymbol{u}_k^{int}+\Omega_k^{mass}\left(e_k^{int}+\frac{1}{2}\boldsymbol{u}_k^{int}\cdot\boldsymbol{u}_k^{int}\right)=\xi\tag{147}$$

where $\xi$ 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. [41])

$$\begin{aligned} \alpha_k \rho_k \frac{\partial e_k^{Fluct}}{\partial t} + \alpha_k \rho_k \boldsymbol{u}_k \cdot \nabla e_k^{Fluct} &= \alpha_k \boldsymbol{T}_k^{Fluct} : \nabla \boldsymbol{u}_k \\ &- \nabla \cdot \langle X_k \rho_0 \frac{\boldsymbol{u}_k' \cdot \boldsymbol{u}_k'}{2} \boldsymbol{u}_k' \rangle \\ &+ \langle X_k \boldsymbol{u}_k' \cdot (\nabla \cdot \boldsymbol{T}_0 + \rho_0 \boldsymbol{g}_0) \rangle \ . \end{aligned} \tag{148}$$

This equation exhibits some similarity to the equation of evolution of the fluctuational kinetic energy in a single-phase turbulent fluid [42]. 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 [35].

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 [35].

### 3.1.8 Entropy Inequality

The local form of the entropy inequality (106), 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

$$\psi_0 = s_0, \qquad J_0 = -\frac{\boldsymbol{q}_0}{\theta_0}, \qquad g_0 = \frac{\varepsilon_0}{\theta_0} \tag{149}$$

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

$$\frac{\partial \alpha_k \rho_k s_k}{\partial t} + \nabla \cdot \alpha_k \rho_k s_k \boldsymbol{u}_k \geq \nabla \cdot \alpha_k (\boldsymbol{\Phi}_k + \boldsymbol{\Phi}_k^{Fluct}) + \alpha_k \rho_k S_k + \Omega_k^{entropy} + \Omega_k^{mass} s_k^{int} \tag{150}$$

where

$$\boldsymbol{\Phi}_k \equiv -\frac{\left\langle X_k \frac{\boldsymbol{q}_0}{\theta_0} \right\rangle}{\alpha_k} \tag{151}$$

is the entropy flux,

$$\boldsymbol{\Phi}_k^{Fluct} \equiv -\frac{\langle X_k \rho_0 s_k' \boldsymbol{u}_k' \rangle}{\alpha_k} \tag{152}$$

is the fluctuation entropy flux,

$$S_k \equiv \frac{\left\langle X_k \frac{\rho_0 \varepsilon_0}{\theta_0} \right\rangle}{\alpha_k \rho_k} \tag{153}$$

is a volumetric entropy source, and

$$\Omega_k^{entropy} \equiv \left\langle \frac{\boldsymbol{q}_0}{\theta_0} \cdot \nabla X_k \right\rangle \tag{154}$$

is an interfacial entropy source. This entropy inequality corresponds to what Drew and Passman [33] call the *microscopic entropy inequality*. A *macroscopic entropy inequality* can be obtained by summing inequalities (150) over all of the phases or materials present in the mixture (for details, see Truesdell [43] 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

$$\sum_{k=1}^{no.\,of\,phases} \Omega_k^{entropy} + \Omega_k^{mass} s_k^{int} \geq 0 \ . \tag{155}$$

### 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 $X_k$,

$$\frac{\partial X_k}{\partial t} + \boldsymbol{u}_{int} \cdot \nabla X_k = 0 \tag{156}$$

this equation is averaged to give

$$\left\langle \frac{\partial X_k}{\partial t} + \boldsymbol{u}_{int} \cdot \nabla X_k \right\rangle = \frac{\partial \alpha_k}{\partial t} + \langle \boldsymbol{u}_{int} \cdot \nabla X_k \rangle = 0 \ . \tag{157}$$

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

$$\begin{aligned} \frac{\partial \alpha_k}{\partial t} + \langle \boldsymbol{u}_{int} \cdot \nabla X_k \rangle &= \frac{\partial \alpha_k}{\partial t} + \langle (\boldsymbol{u}_I + \boldsymbol{u}_I') \cdot \nabla X_k \rangle \\ &= \frac{\partial \alpha_k}{\partial t} + \boldsymbol{u}_I \cdot \nabla \alpha_k + \langle \boldsymbol{u}_I' \cdot \nabla X_k \rangle \\ &= \frac{\partial \alpha_k}{\partial t} + \boldsymbol{u}_I \cdot \nabla \alpha_k - \Omega_k^{vol} \\ &= 0 \end{aligned} \tag{158}$$

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

$$\frac{\partial \alpha_k}{\partial t} + \boldsymbol{u}_I \cdot \nabla \alpha_k = \Omega_k^{vol} \ . \tag{159}$$

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 [29, 44], first order (in time) as above with "viscous damping" effects [45–50], and zeroth order (in time) which amounts to a steady-state version balancing the microstructural forces operating within the $\Omega_k^{vol}$ function [33, 51–56]. 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 $\Omega_k^{vol}$, 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

$$V_1 + V_2 = V \ . \tag{160}$$

The constituent phases have material density $\rho_1$ and $\rho_2$, respectively, so

$$\begin{aligned} V &= V_1 + V_2 \\ &= \frac{m_1}{\rho_1} + \frac{m_2}{\rho_2} \end{aligned} \tag{161}$$

or

$$\begin{aligned} 1 &= \frac{V_1}{V} + \frac{V_2}{V} \\ &= \alpha_1 + \alpha_2 \\ &= \frac{m_1}{V\rho_1} + \frac{m_2}{V\rho_2} \end{aligned} \tag{162}$$

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

$$\begin{aligned} p_1 &= f_1(\rho_1, e_1) \\ &= f_1\left(\frac{m_1}{V_1}, e_1\right) \end{aligned} \tag{163}$$

$$\begin{aligned} p_2 &= f_2(\rho_2, e_2) \\ &= f_2\left(\frac{m_2}{V_2}, e_2\right) . \end{aligned} \tag{164}$$

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

$$p = f_1\left(\frac{m_1}{V_1^*},\, e_1\right) = f_2\left(\frac{m_2}{V_2^*},\, e_2\right) . \tag{165}$$

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

$$\alpha_1^e = \frac{V_1^*}{V}, \qquad \alpha_2^e = \frac{V_2^*}{V} . \tag{166}$$

Alternatively, equations (163) and (164) can be rewritten as

$$\begin{aligned} p_1 &= f_1(\rho_1,\, e_1) \\ &= f_1\left(\frac{m_1}{\alpha_1 V},\, e_1\right) \end{aligned} \tag{167}$$

$$\begin{aligned} p_2 &= f_2(\rho_2,\, e_2) \\ &= f_2\left(\frac{m_2}{\alpha_2 V},\, e_2\right) = f_2\left(\frac{m_2}{(1-\alpha_1)V},\, e_2\right) \end{aligned} \tag{168}$$

and equivalently, $\alpha_1$ 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

$$\frac{d\alpha_1}{dt} = \frac{p_1 - p_2}{\tau} . \tag{169}$$

If $\alpha_1$ is compressed too much (such that $p_1 > p_2$) then $\alpha_1$ will increase with time (i.e. relax) letting $p_1$ reduce while $\alpha_2$ 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, $\tau$, 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

$$\frac{d}{dt}\left[(\text{microinertia}) \times \frac{d\alpha_1}{dt}\right] + (\text{compression viscosity}) \times \frac{d\alpha_1}{dt} = (\text{microstructural forces}) \\ = F \ . \tag{170}$$

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 [57] with

$$F = \begin{cases} \alpha_s \alpha_g (p_s - p_g - \beta_s) & , \quad p_s - \beta_s > 0 \\ -\alpha_s \alpha_g p_g & , \quad p_s - \beta_s \leq 0 \end{cases} \tag{171}$$

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

$$\frac{d\alpha_1}{dt} = \frac{\alpha_1 \alpha_2 (p_1 - p_2)}{\mu} \tag{172}$$

where for this example $\mu$ 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 (115), (134), and (141) respectively, as well as the volume fraction evolution equation (159) with simple pressure driving force. For the liquid ("$liq$" subscript) and vapor ("$vap$" subscript) phases, we have

$$\frac{\partial\left(\alpha\rho\right)_{liq}}{\partial t}+\nabla\cdot\left(\alpha\rho\boldsymbol{u}\right)_{liq}=\Omega_{liq}^{mass} \tag{173}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho\boldsymbol{u}\right)_{liq}}{\partial t}+\nabla\cdot\left(\alpha\rho\boldsymbol{u}\otimes\boldsymbol{u}+\alpha p\boldsymbol{I}\right)_{liq}&=p_{int}\nabla\alpha_{liq}+\lambda(\boldsymbol{u}_{vap}-\boldsymbol{u}_{liq})\\&\quad+\left(\alpha\rho\right)_{liq}\boldsymbol{g}+\boldsymbol{u}_{int}\Omega_{liq}^{mass}\end{aligned} \tag{174}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{liq}}{\partial t}+\nabla\cdot\left[\alpha(\rho E+p)\boldsymbol{u}\right]_{liq}&=p_{int}\boldsymbol{u}_{int}\cdot\nabla\alpha_{liq}+\lambda\boldsymbol{u}'_{int}(\boldsymbol{u}_{vap}-\boldsymbol{u}_{liq})\\&\quad-\mu p'_{int}(p_{liq}-p_{vap})+E_{int}\Omega_{liq}^{mass}+Q_{liq}\end{aligned} \tag{175}$$

$$\frac{\partial\alpha_{liq}}{\partial t}+\nabla\alpha_{liq}\cdot\boldsymbol{u}_{int}=\mu(p_{liq}-p_{vap})+\frac{\Omega_{liq}^{mass}}{\rho_{int}} \tag{176}$$

$$\frac{\partial\left(\alpha\rho\right)_{vap}}{\partial t}+\nabla\cdot\left(\alpha\rho\boldsymbol{u}\right)_{vap}=-\Omega_{liq}^{mass} \tag{177}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho\boldsymbol{u}\right)_{vap}}{\partial t}+\nabla\cdot\left(\alpha\rho\boldsymbol{u}\otimes\boldsymbol{u}+\alpha p\boldsymbol{I}\right)_{vap}&=p_{int}\nabla\alpha_{vap}-\lambda(\boldsymbol{u}_{vap}-\boldsymbol{u}_{liq})\\&\quad+\left(\alpha\rho\right)_{vap}\boldsymbol{g}-\boldsymbol{u}_{int}\Omega_{liq}^{mass}\end{aligned} \tag{178}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{vap}}{\partial t}+\nabla\cdot\left[\alpha(\rho E+p)\boldsymbol{u}\right]_{vap}&=p_{int}\boldsymbol{u}_{int}\cdot\nabla\alpha_{vap}-\lambda\boldsymbol{u}'_{int}(\boldsymbol{u}_{vap}-\boldsymbol{u}_{liq})\\&\quad+\mu p'_{int}(p_{liq}-p_{vap})-E_{int}\Omega_{liq}^{mass}-Q_{liq}\end{aligned} \tag{179}$$

$$\frac{\partial\alpha_{vap}}{\partial t}+\nabla\alpha_{vap}\cdot\boldsymbol{u}_{int}=-\mu(p_{liq}-p_{vap})-\frac{\Omega_{liq}^{mass}}{\rho_{int}} \tag{180}$$

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 (176) or (180) to be replaced by the algebraic relation

$$\alpha_{vap} = 1 - \alpha_{liq} \ . \tag{181}$$

In this relaxation model, $\mu$ 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 $\tau$). Now in this new form, $\mu$ will be more generally called the *pressure relaxation coefficient* or function and similarly $\lambda$ 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 [58] 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:

$$\frac{\partial\left(\alpha\rho\right)_{liq}A}{\partial t}+\frac{\partial\left(\alpha\rho u\right)_{liq}A}{\partial x}=-\Gamma A_{int}A \tag{182}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho u\right)_{liq}A}{\partial t}+\frac{\partial\alpha_{liq}A\left(\rho u^2+p\right)_{liq}}{\partial x}&=p_{int}A\frac{\partial\alpha_{liq}}{\partial x}+p_{liq}\alpha_{liq}\frac{\partial A}{\partial x}\\&+A\lambda(u_{vap}-u_{liq})\\&-\Gamma A_{int}u_{int}A\\&-F_{\text{wall friction},liq}-F_{\text{friction},vap}\\&+\left(\alpha\rho\right)_{liq}A\boldsymbol{g}\cdot\hat{\boldsymbol{n}}_{axis}\end{aligned} \tag{183}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{liq}A}{\partial t}+\frac{\partial\alpha_{liq}u_{liq}A\left(\rho E+p\right)_{liq}}{\partial x}&=p_{int}u_{int}A\frac{\partial\alpha_{liq}}{\partial x}-\bar{p}_{int}A\mu(p_{liq}-p_{vap})\\&+\bar{u}_{int}A\lambda(u_{vap}-u_{liq})\\&+\Gamma A_{int}\left(\frac{p_{int}}{\rho_{int}}-H_{liq,int}\right)A\\&+Q_{int,liq}+Q_{\text{wall},liq}\end{aligned} \tag{184}$$

$$\frac{\partial\alpha_{liq}A}{\partial t}+u_{int}A\frac{\partial\alpha_{liq}}{\partial x}=A\mu(p_{liq}-p_{vap})-\frac{\Gamma A_{int}A}{\rho_{int}} \tag{185}$$

for the liquid phase, and

$$\frac{\partial\left(\alpha\rho\right)_{vap}A}{\partial t}+\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial x}=\Gamma A_{int}A \tag{186}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}A\left(\rho u^2+p\right)_{vap}}{\partial x}&=p_{int}A\frac{\partial\alpha_{vap}}{\partial x}+p_{vap}\alpha_{vap}\frac{\partial A}{\partial x}\\&+A\lambda(u_{liq}-u_{vap})\\&+\Gamma A_{int}u_{int}A\\&-F_{\text{wall friction},vap}-F_{\text{friction},liq}\\&+\left(\alpha\rho\right)_{vap}A\boldsymbol{g}\cdot\hat{\boldsymbol{n}}_{axis}\end{aligned} \tag{187}$$

$$\frac{\partial\left(\alpha\rho E\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}u_{vap}A\left(\rho E+p\right)_{vap}}{\partial x}=p_{int}u_{int}A\frac{\partial\alpha_{vap}}{\partial x}-\bar{p}_{int}A\mu(p_{vap}-p_{liq})$$
$$+\bar{u}_{int}A\lambda(u_{liq}-u_{vap})$$
$$-\Gamma A_{int}\left(\frac{p_{int}}{\rho_{int}}-H_{vap,int}\right)A$$
$$+Q_{int,vap}+Q_{\text{wall},vap} \tag{188}$$

$$\frac{\partial\alpha_{vap}A}{\partial t}+u_{int}A\frac{\partial\alpha_{vap}}{\partial x}=A\mu(p_{vap}-p_{liq})+\frac{\Gamma A_{int}A}{\rho_{int}} \tag{189}$$

for the vapor phase. As before, it is noted that for two-phase flow, either of the differential relations (185) or (189) may be replaced with the algebraic relation

$$\alpha_{vap}=1-\alpha_{liq} \tag{190}$$

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

In equations (182)–(189), $\Gamma$ 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 $F_{\text{wall friction},k}$ 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 $k'$. Similarly, $Q_{int,k}$ is the direct heat transfer from the interface to phase $k$ and $Q_{\text{wall},k}$ is the direct heat transfer from the wall to phase $k$.

Equation system (182)–(189) is the basic system solved with RELAP-7. The system was implemented within the MOOSE computational framework following a series of logically-complete steps [59] 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 [52].

Therefore the assumption of $p_{vap} = p_{liq}$ is very restrictive when derivatives are involved. As pointed out by Boure and Delhaye [60], 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 sense of Hadamard[3]. 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 [64]. 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 [65, 66], 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 Seven-Equation Two-Phase Flow 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)

[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 [61] one does not always know whether these effects are important and under what conditions they are important.

[3]The mathematician Jacques Hadamard [62] 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. [63].

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, $k$-$\epsilon$ turbulence model.

Because the 7-equation two-phase model's most unique features are reflected in the 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 [49], $p_{int}$ 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

$$p_{int} = \sum_{k=1,2} \alpha_k p_k \tag{191}$$

$$u_{int} = \frac{\sum_{k=1,2} \alpha_k \rho_k u_k}{\sum_{k=1,2} \alpha_k \rho_k} \ . \tag{192}$$

In this early research, mechanical relaxation parameters $\mu$ and $\lambda$ were also specified in a, more or less, ad hoc manner. Abgrall and Saurel [67] 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

[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.

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 $p_{int}$, $u_{int}$, $\mu$, and $\lambda$ 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 [14, 68] to

$$p_{int} = \bar{p}_{int} + \frac{Z_{liq} Z_{vap}}{Z_{liq} + Z_{vap}} \text{sgn}\left(\frac{\partial \alpha_{liq}}{\partial x}\right)(u_{vap} - u_{liq}) \tag{193}$$

$$\bar{p}_{int} = \frac{Z_{vap} p_{liq} + Z_{liq} p_{vap}}{Z_{liq} + Z_{vap}} \tag{194}$$

$$u_{int} = \bar{u}_{int} + \text{sgn}\left(\frac{\partial \alpha_{liq}}{\partial x}\right) \frac{p_{vap} - p_{liq}}{Z_{liq} + Z_{vap}} \tag{195}$$

$$\bar{u}_{int} = \frac{Z_{liq} u_{liq} + Z_{vap} u_{vap}}{Z_{liq} + Z_{vap}} \tag{196}$$

$$\lambda = \frac{1}{2} \mu Z_{liq} Z_{vap} \tag{197}$$

$$\mu = \frac{A_{int}}{Z_{liq} + Z_{vap}} \tag{198}$$

where $\lambda$ is the velocity relaxation coefficient function, $\mu$ is the pressure relaxation coefficient function, $Z_k = \rho_k c_k$, $(k = liq, vap)$, 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. [12].

Remark (1): From this specification of $\lambda$ and $\mu$ 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 [69]. 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 $\mu$ (pressure) and $\lambda$ (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 $\mu$ 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 [12, 70].

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 (193) to (198) 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. That is,

$$Q_{\text{wall},k} = H_{w,k} a_w \left(T_{\text{wall}} - T_k\right) \alpha_k A \tag{199}$$

for phase $k = (liq, vap)$, where $H_{w,k}$ is the wall convective wall heat transfer coefficient associated with phase $k$. 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

$$Q_{int,k} = h_{T,k} \left( T_{int} - T_k \right) A_{int} A \tag{200}$$

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

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

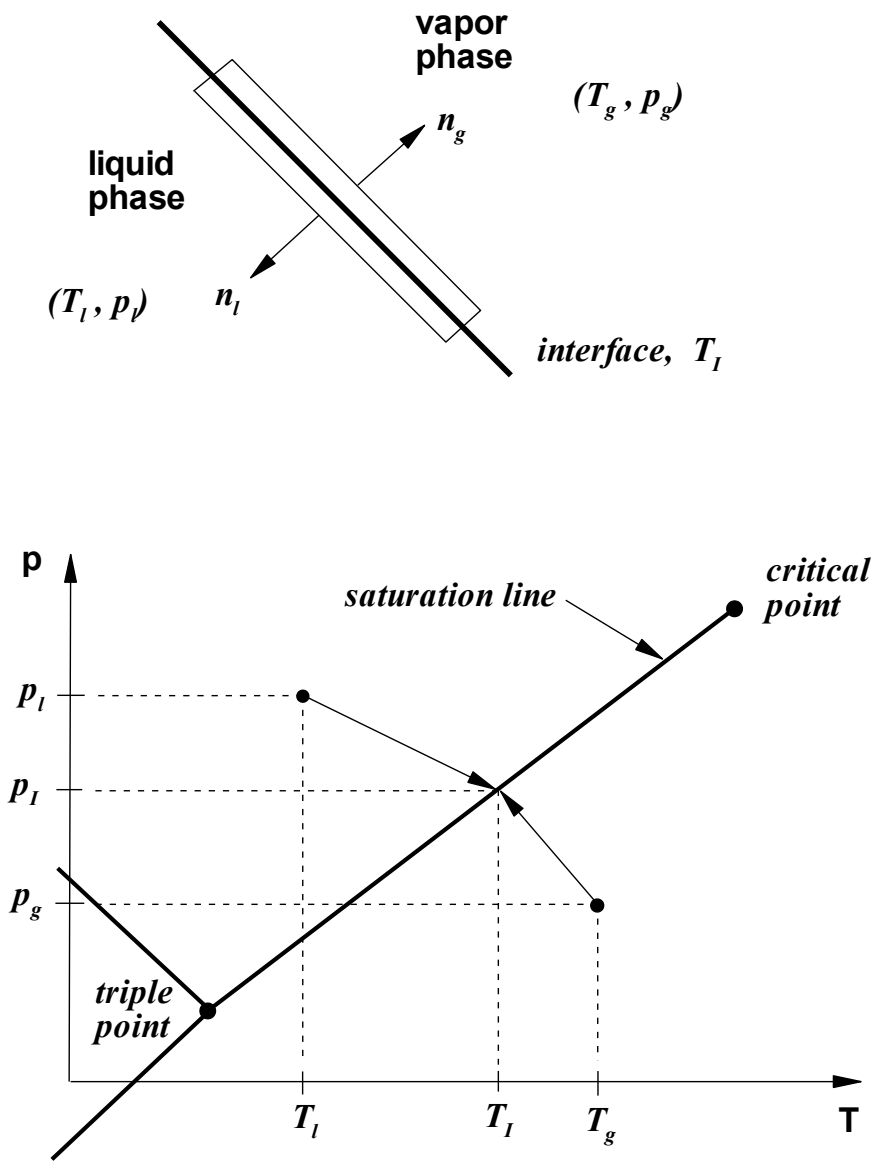


**Figure 2.** Interface control volume (top); $T$-$p$ state space around saturation line, $T_{liq} < T_{vap}$, (bottom).

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 integrated over a "pill-box" control volume containing an interface (see e.g. Kuo [71]), as shown in the upper graphic of Figure 2, to obtain the algebraic "jump conditions"

$$\rho_{liq,int}\boldsymbol{u}_{liq,int}\cdot\hat{\boldsymbol{n}}_{liq}+\rho_{vap,int}\boldsymbol{u}_{vap,int}\cdot\hat{\boldsymbol{n}}_{vap}=0 \tag{201}$$

or

$$\Gamma_{liq}+\Gamma_{vap}=0 \tag{202}$$

for mass, and

$$\Gamma_{liq}h_{liq,int}+\boldsymbol{q}_{liq,int}\cdot\hat{\boldsymbol{n}}_{liq}+\Gamma_{vap}h_{vap,int}+\boldsymbol{q}_{vap,int}\cdot\hat{\boldsymbol{n}}_{vap}=0 \tag{203}$$

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

$$\begin{aligned}\boldsymbol{q}_{k,int}\cdot\hat{\boldsymbol{n}}_k &= -k_k\nabla T_{k,int}\cdot\hat{\boldsymbol{n}}_k\\ &= h_{T,k}\left(T_{int}-T_k\right)\end{aligned} \tag{204}$$

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

$$\begin{aligned}\Gamma=\Gamma_{vap} &= \frac{h_{T,liq}\left(T_{liq}-T_{int}\right)+h_{T,vap}\left(T_{vap}-T_{int}\right)}{h_{vap,int}-h_{liq,int}}\\ &= \frac{h_{T,liq}\left(T_{liq}-T_{int}\right)+h_{T,vap}\left(T_{vap}-T_{int}\right)}{L_v\left(T_{int}\right)}\end{aligned} \tag{205}$$

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 $T_{liq}<T_{vap}$).

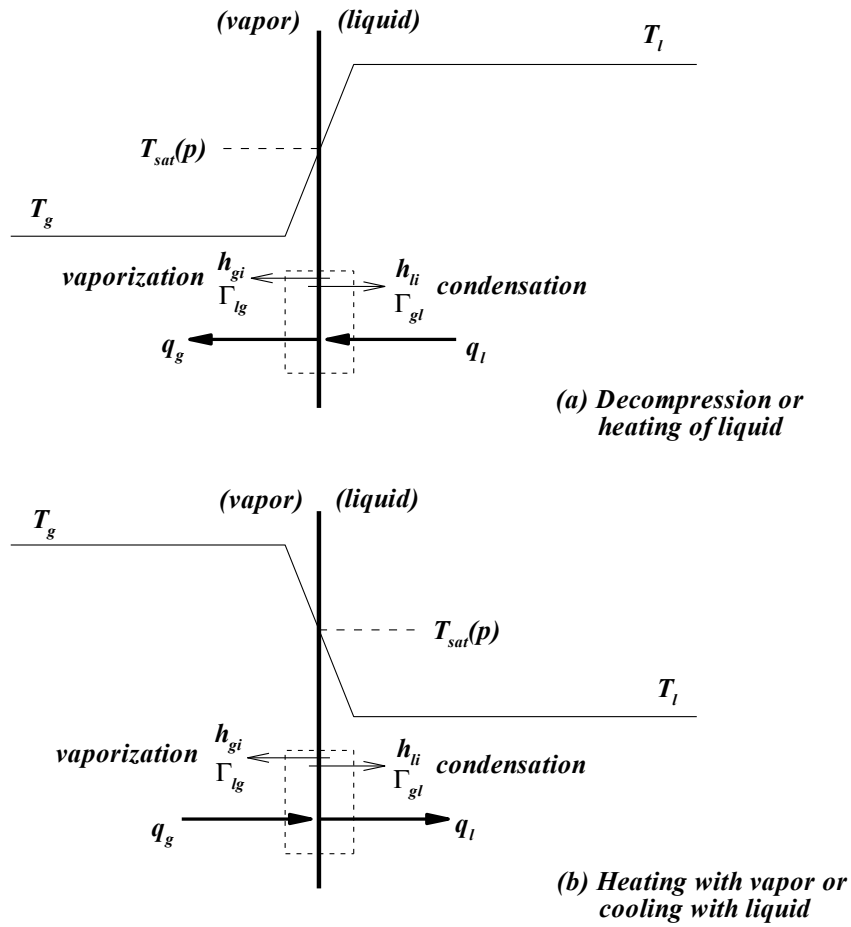


**Figure 3.** Vaporization and condensation at a liquid-vapor interface (after Moody [1]).

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 $q_{liq}$ 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:

$$H_{k,int} = h_{k,int} + \frac{1}{2}u_{int}^2 \tag{206}$$

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 $\rho_{int} = \rho_{liq,sat}(p_{int})$.

## Simple Interface Heat- and Mass-Transfer Model

A simple model described here which is a more complete version of the one described in [12]. That model always assumed droplets in a vapor regardless of volume fraction. Here it will be assumed that the phase topology is bubbles in a liquid for $\alpha_{vap}$ less than some specified value, $\alpha_{vap,A} = 0.3$, and that the phase topology is droplets in a vapor for $\alpha_{vap}$ greater than another specifed value, $\alpha_{vap,B} = 0.7$. For vapor volume fractions $\alpha_{vap}$ between $\alpha_{vap,A}$ and $\alpha_{vap,B}$ a linear interpolation of properties corresponding to those two vapor volume fraction points is performed.

Interfacial specific area and cut-off values are first computed/specifed, e.g.:

$$A_{int,max} = 1500.\ (user\ input) \tag{207}$$
$$A_{int} = A_{int,max}[4\alpha_{liq}(1-\alpha_{liq})] - offset \tag{208}$$
$$offset = 0.5 \tag{209}$$
$$\alpha_{vap,A} = 0.3 \tag{210}$$
$$\alpha_{vap,B} = 0.7 \tag{211}$$
$$A_{int,A} = A_{int,max}[4\alpha_{liq,A}(1-\alpha_{liq,A})] - offset \tag{212}$$
$$A_{int,B} = A_{int,max}[4\alpha_{liq,B}(1-\alpha_{liq,B})] - offset. \tag{213}$$

The interface saturation temperature corresponding to $p_{int}$, $T_{int} = T_{sat}(p_{int})$, is obtained from the equations of state along with phasic enthalpies $h_{liq,sat}$ and $h_{vap,sat}$ at $T_{sat}$, and heat of vaporization $L_{vap}(T_{sat})$ at $T_{sat}$. The interface density is taken to be the saturated liquid density at $p_i$; e.g. for the stiffened gas EOS:

$$\rho_{int} = \frac{p_{int} + p_{\infty,liq}}{(\gamma_{liq}-1)c_{v,liq}T_{int}}. \tag{214}$$

Thermal conductivities $k_{liq}$ and $k_{vap}$ along with dynamic (or ordinary) viscosity $\nu_{liq}$ and $\nu_{vap}$ are assumed constant. For example:

$$k_{liq} = 0.5 \tag{215}$$
$$k_{vap} = 0.026 \tag{216}$$
$$\nu_{liq} = 281.8 * 10^{-6} \tag{217}$$
$$\nu_{vap} = 134.4 * 10^{-7}. \tag{218}$$

The bubble and droplet sizes and their values corresponding to vapor volume fraction

cut-off values, $\alpha_{vap,A}$ and $\alpha_{vap,B}$, are:

$$r_{bub} = \frac{3\alpha_{vap}}{A_{int}} \tag{219}$$

$$r_{bub,A} = \frac{3\alpha_{vap,A}}{A_{int,A}} \tag{220}$$

$$r_{drop} = \frac{3\alpha_{liq}}{A_{int}} = \frac{3(1-\alpha_{vap})}{A_{int}} \tag{221}$$

$$r_{drop,B} = \frac{3\alpha_{liq,B}}{A_{int,B}} = \frac{3(1-\alpha_{vap,B})}{A_{int,B}} \tag{222}$$

$$d_{bub} = 2r_{bub} \tag{223}$$

$$d_{bub,A} = 2r_{bub,A} \tag{224}$$

$$d_{drop} = 2r_{drop} \tag{225}$$

$$d_{drop,B} = 2r_{drop,B}. \tag{226}$$

The convection coefficients for the interiors of bubbles and droplets and their values corresponding to vapor volume fraction cut-off values, $\alpha_{vap,A}$ and $\alpha_{vap,B}$, are taken as:

$$h_{conv,bub} = \frac{5k_{vap}}{r_{bub}} \tag{227}$$

$$h_{conv,bub,A} = \frac{5k_{vap}}{r_{bub,A}} \tag{228}$$

$$h_{conv,drop} = \frac{5k_{liq}}{r_{drop}} \tag{229}$$

$$h_{conv,drop,B} = \frac{5k_{liq}}{r_{drop,B}}. \tag{230}$$

These convective coefficient values correspond to conduction values [12].

The phasic isobaric specific heats are determined from the corresponding phasic isochoric specific heats as

$$c_{p,liq} = \gamma_{liq} c_{v,liq} \tag{231}$$

$$c_{p,vap} = \gamma_{vap} c_{v,vap}. \tag{232}$$

Reynolds numbers for bubbles, droplets, and their respective cut-off values are

$$Re_1 = \frac{\rho_{liq} d_{bub} |u_{liq} - u_{vap}|}{\nu_{liq}} \tag{233}$$

$$Re_{1A} = \frac{\rho_{liq} d_{bub,A} |u_{liq} - u_{vap}|}{\nu_{liq}} \tag{234}$$

$$Re_2 = \frac{\rho_{vap} d_{drop} |u_{vap} - u_{liq}|}{\nu_{vap}} \tag{235}$$

$$Re_{2B} = \frac{\rho_{vap} d_{drop,B} |u_{vap} - u_{liq}|}{\nu_{vap}}. \tag{236}$$

Prandtl numbers of bubbles and droplets are

$$Pr_1 = \frac{\nu_{liq} c_{p,liq}}{k_{liq}} \tag{237}$$

$$Pr_2 = \frac{\nu_{vap} c_{p,vap}}{k_{vap}}. \tag{238}$$

The Nusselt numbers of bubbles, droplets, and their respective cut-off values are determined from

$$Nu_1 = 2 + 0.6 Re_1^{0.5} Pr_1^{0.33} \tag{239}$$

$$Nu_{1A} = 2 + 0.6 Re_{1A}^{0.5} Pr_1^{0.33} \tag{240}$$

$$Nu_2 = 2 + 0.6 Re_2^{0.5} Pr_2^{0.33} \tag{241}$$

$$Nu_{2B} = 2 + 0.6 Re_{2B}^{0.5} Pr_2^{0.33}. \tag{242}$$

Finally, the convective heat transfer coefficients for the carrier phases and their values corresponding to vapor volume fraction cut-off values, $\alpha_{vap,A}$ and $\alpha_{vap,B}$, are determined as:

$$conv_1 = \frac{k_{liq} Nu_1}{d_{bub}} \tag{243}$$

$$conv_{1A} = \frac{k_{liq} Nu_{1A}}{d_{bub,A}} \tag{244}$$

$$conv_2 = \frac{k_{vap} Nu_2}{d_{drop}} \tag{245}$$

$$conv_{2B} = \frac{k_{vap} Nu_{2B}}{d_{drop,B}}. \tag{246}$$

The determination of the final interface convective heat transfer coefficients can be summarized as follows:

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

$$h_{conv,liq} = conv_1 \tag{247}$$
$$h_{conv,vap} = h_{conv,bub} \tag{248}$$

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

$$h_{conv,liq} = h_{conv,drop} \tag{249}$$
$$h_{conv,vap} = conv_2 \tag{250}$$

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

$$h_{conv,liq} = conv_{1A} + \frac{(h_{conv,drop,B} - conv_{1A})(\alpha_{vap} - \alpha_{vap,A})}{\alpha_{vap,B} - \alpha_{vap,A}} \tag{251}$$
$$h_{conv,vap} = h_{conv,bub,A} + \frac{(conv_{2B} - h_{conv,bub,A})(\alpha_{vap} - \alpha_{vap,A})}{\alpha_{vap,B} - \alpha_{vap,A}}. \tag{252}$$

The total saturated phasic enthalpies are constructed as

$$H_{liq,sat} = h_{liq,sat} + 0.5v_{int}^2 \tag{253}$$
$$H_{vap,sat} = h_{vap,sat} + 0.5v_{int}^2 \tag{254}$$

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

$$L_{tot}(T_{sat}) = H_{vap,sat} - H_{liq,sat}. \tag{255}$$

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

$$\Gamma_{int,vap} = \frac{h_{conv,liq}(T_{liq} - T_{int}) + h_{conv,vap}(T_{vap} - T_{int})}{L_{vap}(T_{sat})}. \tag{256}$$

Note, additional vapor will be generated at the wall, $\Gamma_{wall,vap}$, due to local wall boiling such that $\Gamma_{vap}A = \Gamma_{int,vap}(A_{int})A + \Gamma_{wall,vap}$. Such wall boiling model will be described subsequently.

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*

$$+ \Gamma_{int,vap}(A_{int})(\frac{p_{int}}{\rho_{int}} - H_{liq,sat})A + (A_{int})h_{conv,liq}(T_{int} - T_{liq})A \tag{257}$$

*Vapor energy equation terms*

$$- \Gamma_{int,vap}(A_{int})(\frac{p_{int}}{\rho_{int}} - H_{vap,sat})A + (A_{int})h_{conv,vap}(T_{int} - T_{vap})A. \tag{258}$$

Simple Wall Boiling

A simple 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 $\beta$ depends upon the wall temperature and the saturated liquid temperature, i.e. $\beta(T_{wall}, T_{sat,liq})$. Rendering this into equation form,

$$\begin{aligned} Q_{wall,total} &= Q_{wall,vap} + Q_{wall,liq} \\ &= Q_{wall,vap} + Q_{wall,liq,conv} + Q_{wall,liq,boil} \end{aligned} \tag{259}$$

where

$$\begin{aligned} Q_{wall,vap} &= h_{wall,vap}(T_{wall} - T_{vap})(1-\kappa)a_{wall}A \\ &= h_{wall,vap}(T_{wall} - T_{vap})(1-\kappa)P_{hf} \end{aligned} \tag{260}$$

$$
\begin{aligned}
Q_{wall,liq} &= h_{wall,liq}(T_{wall} - T_{liq})\,\kappa\,P_{hf} \\
&= Q_{wall,liq,conv} + Q_{wall,liq,boil} \\
&= \beta Q_{wall,liq} + (1-\beta)Q_{wall,liq}
\end{aligned} \tag{261}
$$

so

$$
Q_{wall,liq,conv} = \beta h_{wall,liq}(T_{wall} - T_{liq})\,\kappa\,P_{hf} \tag{262}
$$

$$
Q_{wall,liq,boil} = (1-\beta) h_{wall,liq}(T_{wall} - T_{liq})\,\kappa\,P_{hf} \tag{263}
$$

The rate of vapor generated by boiling at the wall due to wall heat flux is then

$$
\begin{aligned}
\Gamma_{wall,vap} &= \frac{Q_{wall,liq,boil}}{c_{p,liq}(T_{sat,liq} - T_{liq}) + L(T_{sat})} \\
&= \frac{(1-\beta)h_{wall,liq}(T_{wall} - T_{liq})\,\kappa\,P_{hf}}{c_{p,liq}(T_{sat,liq} - T_{liq}) + L(T_{sat})}.
\end{aligned} \tag{264}
$$

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:

$$
\Gamma_{vap}A = \Gamma_{int,vap}A_{int}A + \Gamma_{wall,vap}.
$$

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

Here $\beta$ is unity 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.

$$
\begin{aligned}
T_{wall} \le T_{sat,liq} \quad &\Rightarrow \quad \beta = 1 \\
T_{wall} > T_{sat,liq} \quad &\Rightarrow \quad \beta = exp[-0.25(T_{solid,wall} - T_{sat,liq})].
\end{aligned} \tag{265}
$$

In the above $\kappa$ 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 $0.01 < \alpha_{liq} < 1.0$.

This results in the phasic balance equations of mass, momentum, and total energy along with volume fraction evolution:

$$\frac{\partial\left(\alpha\rho\right)_{liq}A}{\partial t}+\frac{\partial\left(\alpha\rho u\right)_{liq}A}{\partial x}=-\Gamma_{int,vap}A_{int}A-\Gamma_{wall,vap} \tag{266}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho u\right)_{liq}A}{\partial t}+\frac{\partial\alpha_{liq}A\left(\rho u^2+p\right)_{liq}}{\partial x}&=p_{int}A\frac{\partial\alpha_{liq}}{\partial x}+p_{liq}\alpha_{liq}\frac{\partial A}{\partial x}\\&+A\lambda(u_{vap}-u_{liq})\\&-\Gamma_{int,vap}A_{int}u_{int}A-\Gamma_{wall,vap}u_{int}\\&-F_{\text{wall friction},liq}-F_{\text{friction},vap}\\&+\left(\alpha\rho\right)_{liq}A\boldsymbol{g}\cdot\hat{n}_{axis}\end{aligned} \tag{267}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{liq}A}{\partial t}+\frac{\partial\alpha_{liq}u_{liq}A\left(\rho E+p\right)_{liq}}{\partial x}&=p_{int}u_{int}A\frac{\partial\alpha_{liq}}{\partial x}-\bar{p}_{int}A\mu(p_{liq}-p_{vap})\\&+\bar{u}_{int}A\lambda(u_{vap}-u_{liq})\\&+\Gamma_{int,vap}A_{int}\left(\frac{p_{int}}{\rho_{int}}-H_{liq,int}\right)A\\&+A_{int}h_{conv,liq}(T_{int}-T_{liq})A\\&+h_{wall,liq,conv}(T_{wall}-T_{liq})\,\kappa\,P_{hf}\\&-\Gamma_{wall,vap}\left(-\frac{p_{int}}{\rho_{int}}+h_{vap,int}+\frac{u_{int}^2}{2}\right)\\&+\left(\alpha\rho u\right)_{liq}A\boldsymbol{g}\cdot\hat{n}_{axis}\end{aligned} \tag{268}$$

$$\frac{\partial\alpha_{liq}A}{\partial t}+u_{int}A\frac{\partial\alpha_{liq}}{\partial x}=A\mu(p_{liq}-p_{vap})-\frac{\Gamma_{int,vap}A_{int}A}{\rho_{int}}-\frac{\Gamma_{wall,vap}}{\rho_{int}} \tag{269}$$

for the liquid phase, and

$$\frac{\partial\left(\alpha\rho\right)_{vap}A}{\partial t}+\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial x}=\Gamma_{int,vap}A_{int}A+\Gamma_{wall,vap} \tag{270}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}A\left(\rho u^2+p\right)_{vap}}{\partial x}&=p_{int}A\frac{\partial\alpha_{vap}}{\partial x}+p_{vap}\alpha_{vap}\frac{\partial A}{\partial x}\\&+A\lambda(u_{liq}-u_{vap})\\&+\Gamma_{int,vap}A_{int}u_{int}A+\Gamma_{wall,vap}u_{int}\\&-F_{\text{wall friction},vap}-F_{\text{friction},liq}\\&+\left(\alpha\rho\right)_{vap}A\boldsymbol{g}\cdot\hat{n}_{axis}\end{aligned} \tag{271}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}u_{vap}A\left(\rho E+p\right)_{vap}}{\partial x}&=p_{int}u_{int}A\frac{\partial\alpha_{vap}}{\partial x}-\bar{p}_{int}A\mu(p_{vap}-p_{liq})\\&+\bar{u}_{int}A\lambda(u_{liq}-u_{vap})\\&-\Gamma_{int,vap}A_{int}\left(\frac{p_{int}}{\rho_{int}}-H_{vap,int}\right)A\\&+A_{int}h_{conv,vap}(T_{int}-T_{vap})A\\&+h_{wall,vap,conv}(T_{wall}-T_{vap})(1-\kappa)P_{hf}\\&+\Gamma_{wall,vap}\left(-\frac{p_{int}}{\rho_{int}}+h_{vap,int}+\frac{u_{int}^2}{2}\right)\\&+\left(\alpha\rho u\right)_{vap}A\boldsymbol{g}\cdot\hat{n}_{axis}\end{aligned} \tag{272}$$

$$\frac{\partial\alpha_{vap}A}{\partial t}+u_{int}A\frac{\partial\alpha_{vap}}{\partial x}=A\mu(p_{vap}-p_{liq})+\frac{\Gamma_{int,vap}A_{int}A}{\rho_{int}}+\frac{\Gamma_{wall,vap}}{\rho_{int}} \tag{273}$$

for the vapor phase.

### 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 reduced by the fraction of the wall area which the phase occupies. Thus

$$F_{\text{wall friction},k} = \frac{f_k}{2d_h}\rho_k u_k \left|u_k\right| \alpha_k A \tag{274}$$

for phases $k = (liq, vap)$, where $f_k$ is the wall friction factor associated with phase $k$. As discussed in Section 2.2.1, the hydraulic diameter $d_h$ depends on the shape of the cross section, and the position $x$ in the pipe.

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:

$$F_{\text{friction},k'} = f_{k,\,k'}\frac{1}{2}\rho_k(u_k - u_{int})\left|u_k - u_{int}\right| A_{int} A \tag{275}$$

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

$$F_{friction,k'} = K_{k,k'}(u_{k'} - u_k)A. \tag{276}$$

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 [72] by first determining effective bubble/droplet radius, $r_0$, as

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

$$r_0 = r_{bub} \tag{277}$$

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

$$r_0 = r_{drop} \tag{278}$$

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

$$r_0 = r_{bub,A} + \frac{(r_{drop,B} - r_{bub,A})(\alpha_{vap} - \alpha_{vap,A})}{\alpha_{vap,B} - \alpha_{vap,A}}. \tag{279}$$

Then

$$K_{k,k'} = \frac{\rho A_{int}}{8}[C_D|u_k - u_{k'}| + \frac{12\hat{\nu}}{r_0}] \tag{280}$$

where

$$\rho = \alpha_{vap}\rho_{vap} + (1-\alpha_{vap})\rho_{liq}$$

$$\hat{\nu} = \alpha_{vap}\hat{\nu}_{vap} + (1-\alpha_{vap})\hat{\nu}_{liq} \;\; (kinematic\ viscosity) \tag{281}$$

$$C_D = 0.5. \tag{282}$$

### 3.2.5 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) [3] was selected

$$p(\rho, e) = (\gamma - 1)\rho(e - q) - \gamma p_\infty \tag{283}$$

where $p$, $\rho$, $e$, and $q$ are the pressure, density, internal energy, and the binding energy of the fluid considered. The parameters $\gamma$, $q$, and $p_\infty$ 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 [3] 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:

$$e(p,\rho) = \frac{p+\gamma p_\infty}{(\gamma-1)\rho} + q \tag{284}$$

$$\rho(p,T) = \frac{p+p_\infty}{(\gamma-1)c_v T} \tag{285}$$

$$h(T) = \gamma c_v T + q \tag{286}$$

$$g(p,T) = (\gamma c_v - q')\,T - c_v T \ln \frac{T^\gamma}{(p+p_\infty)^{\gamma-1}} + q \tag{287}$$

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 [3]. 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 (286) it is readily seen that the phase $k$ specific enthalpy evaluated at the interface condition from equation (206) is

$$h_{k,int} = c_{p,k} T_{int} + q_k \tag{288}$$

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

The bulk interphase mass transfer from the liquid phase to the vapor phase $\Gamma$ 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 [3] 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 (287) yields

$$\ln\left(p + p_{\infty,vap}\right) = A + \frac{B}{T} + C\ln(T) + D\ln\left(p + p_{\infty,liq}\right) \tag{289}$$

where

$$A = \frac{c_{p,liq} - c_{p,vap} + q'_{vap} - q'_{liq}}{c_{p,vap} - c_{v,vap}} \tag{290}$$

$$B = \frac{q_{liq} - q_{vap}}{c_{p,vap} - c_{v,vap}} \tag{291}$$

$$C = \frac{c_{p,vap} - c_{p,liq}}{c_{p,vap} - c_{v,vap}} \tag{292}$$

$$D = \frac{c_{p,liq} - c_{v,liq}}{c_{p,vap} - c_{v,vap}} \ . \tag{293}$$

Relation (289) 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

$$\begin{aligned} L_v\left(T_{int}\right) &= h_{vap,int} - h_{liq,int} \\ &= h_{k,int} \\ &= \left(\gamma_{vap} c_{v,vap} T + q_{vap}\right) - \left(\gamma_{liq} c_{v,liq} T + q_{liq}\right) \ . \end{aligned} \tag{294}$$

## 3.3 Homogeneous Equilibrium Two-Phase Flow 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 preformed 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 [64], 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 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 HEM 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 $\alpha$ 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

$$\alpha = \frac{\bar{\rho} - \rho_{liq,sat}(T)}{\rho_{vap,sat}(T) - \rho_{liq,sat}(T)} \tag{295}$$

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 HEM 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:

$$\bar{\mu} = \mu_{liq}(1-\alpha) + \alpha\mu_{vap} \tag{296}$$

$$\bar{k} = k_{liq}(1-\alpha) + \alpha k_{vap} \ . \tag{297}$$

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 Heat Conduction Model

## 4.1 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

$$\rho C_p \frac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) - q''' = 0 \tag{298}$$

where $\rho$, $C_p$, $k$ 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

$$T_{\text{bc}} = T_0 \ . \tag{299}$$

The second type is the Neumann boundary condition, which provides a heat flux boundary condition

$$q''_{\text{bc}} = -k \frac{\partial T}{\partial \hat{n}_{\text{bc}}} = q''_0 \ . \tag{300}$$

The third type is the Robin boundary condition, which provides the convective heat transfer boundary condition

$$-k \frac{\partial T}{\partial \hat{n}_{\text{bc}}} = h_{\text{conv}} (T_{\text{coolant}} - T_{\text{bc}}) \ . \tag{301}$$

Both 1-D and 2-D solutions for the heat conduction model are available in RELAP-7.

## 4.2 Material Properties

Thermal properties, such as thermal conductivity $k$, material density $\rho$, 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 [73] whenever possible. The constant room-temperature densities ($\rho$) 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.

### 4.2.1 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

$$\begin{aligned} k = & \frac{D}{1 + T'(1 - D)} \frac{C_v}{(A + BT'')(1 + 3e_{th})} \\ & + 5.2997 \times 10^{-3} T \exp\left(\frac{-13358}{T}\right) \left[1 + 0.169 \left(\frac{13358}{T} + 2\right)^2\right] \end{aligned} \tag{302}$$

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

$$C_v = \frac{296.7 \times 535.285^2}{T^2 \left[\exp\left(\frac{535.285}{T}\right) - 1\right]^2} \exp\left(\frac{535.285}{T}\right) \tag{303}$$

$e_{th}$ is the linear strain term for temperatures above 300 K (dimensionless), which is given by

$$e_{th} = \frac{\Delta L}{L_0} = 1.0 \times 10^{-5} T - 3.0 \times 10^{-3} + 4.0 \times 10^{-2} \exp\left(\frac{-6.9 \times 10^{-20}}{1.38 \times 10^{-23} T}\right) \tag{304}$$

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).

### 4.2.2 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

$$k = a_0 + a_1 T + a_2 T^2 + a_3 T^3 \tag{305}$$

for $300 < T < 2098$K, and $k = 36$ for $T \geq 2098K$. The remaining $a_i$ parameters in (305) are given in Table 8.

**Table 8.** Zircaloy thermal conductivity parameters.

| | |
|---|---|
| $a_0$ | $7.51$ |
| $a_1$ | $2.09 \times 10^{-2}$ |
| $a_2$ | $-1.45 \times 10^{-5}$ |
| $a_3$ | $7.67 \times 10^{-9}$ |

### 4.2.3 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

$$k_{\text{mix}} = \sum_{i=1}^{n} \left[ \frac{k_i x_i}{x_i + \sum_{j=1}^{n} (1 - \delta_{ij}) \psi_{ij} x_i} \right] \tag{306}$$

where

$$\psi_{ij} = \phi_{ij}\left[1 + 2.41\frac{(M_i - M_j)(M_i - 0.142M_j)}{(M_i - M_j)^2}\right] \tag{307}$$

and

$$\phi_{ij} = \frac{\left[\left(1 + \frac{k_i}{k_j}\right)^{1/2}\left(\frac{M_i}{M_j}\right)^{1/4}\right]^2}{2^{\frac{2}{3}}\left(1 + \frac{M_i}{M_j}\right)^{1/2}} \tag{308}$$

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.

# 5 Numerical Methods

## 5.1 Spatial Discretization Algorithm

This section focuses on the weak statement associated to the strong form of the one-dimensional, variable cross-sectional area form of the Euler equations (20)–(22) summarized in Section 2.1.1. Start by writing the equations in "vector" form as

$$\boldsymbol{R}(\boldsymbol{U}) \equiv \frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial x} - \boldsymbol{S} = \boldsymbol{0} \tag{309}$$

where

$$\boldsymbol{U} \equiv \begin{bmatrix} \rho A \\ \rho u A \\ \rho E A \end{bmatrix}, \qquad \boldsymbol{F} \equiv \begin{bmatrix} \rho u A \\ (\rho u^2 + p)\, A \\ \rho u H A \end{bmatrix} \tag{310}$$

and $\boldsymbol{S} = \boldsymbol{S}(\boldsymbol{U})$ consists of the remaining source terms. Note that $\boldsymbol{U}$ and $\boldsymbol{F}$ are identical to their meanings in the "constant-area" equations, up to multiplication by the area, $A$. The variational statement proceeds by dotting (309) by an "admissible" vector test function $\boldsymbol{W}$ (more details of which will be given momentarily), integrating over the domain $\Omega$, and applying the divergence theorem. Solutions $\boldsymbol{U}$ are sought such that

$$\int_\Omega \left( \frac{\partial \boldsymbol{U}}{\partial t} \cdot \boldsymbol{W} - \boldsymbol{F} \cdot \frac{\partial \boldsymbol{W}}{\partial x} - \boldsymbol{S} \cdot \boldsymbol{W} \right) \mathrm{d}\Omega + \int_\Gamma (\boldsymbol{F} \cdot \boldsymbol{W})\, \hat{n}_x \, \mathrm{d}\Gamma = 0 \tag{311}$$

holds for *all* admissible $\boldsymbol{W}$. Note that the test function $\boldsymbol{W}$ is not chosen arbitrarily. In particular, we require that $\boldsymbol{W}$ come from the space of vector functions

$$\boldsymbol{W} \in \left\{ \begin{bmatrix} w \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ w \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ w \end{bmatrix} \right\} \tag{312}$$

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 (311) makes sense. The approximate problem then 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 $\{\phi_i\}$, $i = 1, \ldots, N$. We then seek $\boldsymbol{U}^h$ with components in the same space as $\mathcal{W}^h$, satisfying the boundary conditions, and such that

$$\int_\Omega \left( \frac{\partial \boldsymbol{U}^h}{\partial t} \cdot \boldsymbol{W}^h - \boldsymbol{F}^h \cdot \frac{\partial \boldsymbol{W}^h}{\partial x} - \boldsymbol{S}^h \cdot \boldsymbol{W}^h \right) \mathrm{d}\Omega + \int_\Gamma \left( \boldsymbol{F}^h \cdot \boldsymbol{W}^h \right) \hat{n}_x \, \mathrm{d}\Gamma = 0 \tag{313}$$

holds for all $\boldsymbol{W}^h$ defined analogously to (312), with components in $\mathcal{W}^h$. Note that (313) has been placed in a "continuous" setting, that is, a mesh and finite element discretization has been introduced requiring a continuous solution. Equation (313) remains a "weak" restatement of the "strong" equations (309) in the sense that derivatives of the solution and its flux need not be continuous. More will be said of this subsequently, in the upcoming section on stabilization methods. Written out in component form, and denoting the components of $\boldsymbol{U}^h$ by $U_0^h$, $U_1^h$, and $U_2^h$, (313) expands to:

$$\int_\Omega \left( \frac{\partial U_0^h}{\partial t} \phi_i - U_1^h \frac{\partial \phi_i}{\partial x} \right) \mathrm{d}\Omega + \int_\Gamma U_1^h \hat{n}_x \phi_i \, \mathrm{d}\Gamma = 0 \tag{314}$$

$$\begin{aligned} \int_\Omega \left[ \left( \frac{\partial U_1^h}{\partial t} - U_0^h g_x + \frac{f}{2d_h} U_1^h \left| \frac{U_1^h}{U_0^h} \right| - p^h \frac{\partial A}{\partial x} \right) \phi_i - \left( \frac{(U_1^h)^2}{U_0^h} + p^h A \right) \frac{\partial \phi_i}{\partial x} \right] \mathrm{d}\Omega \\ + \int_\Gamma \left( \frac{(U_1^h)^2}{U_0^h} + p^h A \right) \hat{n}_x \phi_i \, \mathrm{d}\Gamma = 0 \end{aligned} \tag{315}$$

$$\begin{aligned} \int_\Omega \left[ \left( \frac{\partial U_2^h}{\partial t} + H_w a_w (T^h - T_w) A - U_1^h g_x \right) \phi_i - U_1^h H^h \frac{\partial \phi_i}{\partial x} \right] \mathrm{d}\Omega \\ + \int_\Gamma U_1^h H^h \hat{n}_x \phi_i \, \mathrm{d}\Gamma = 0 \end{aligned} \tag{316}$$

Equations (314)–(316) must hold for $i = 1, \ldots, N$. Note that the approximate pressure, $p^h$, temperature, $T^h$, and enthalpy, $H^h$ are functions of the conserved variables $U_0^h, U_1^h, U_2^h$. As mentioned, a continuous Galerkin formulation is employed, and therefore the unknowns are expressed in the same basis used for the test functions, i.e.

$$U_0^h = \sum_j (U_0)_j \phi_j \tag{317}$$

$$U_1^h = \sum_j (U_1)_j \phi_j \tag{318}$$

$$U_2^h = \sum_j (U_2)_j \phi_j \tag{319}$$

The coefficients $(U_0)_j$, $(U_1)_j$, and $(U_2)_j$ vary in time only, and comprise the solution vector at each iteration. Note that (314)–(316) are so-called "semi-discrete" equations: they have been discretized in space, but the temporal derivatives remain in continuous form. In Section 5.2 we discuss the various time discretization methods employed in RELAP-7. Furthermore, it is well-known that a continuous Galerkin discretization of this set of

hyperbolic equations is equivalent to a central difference method for a certain choice of integration rule, and therefore will exhibit oscillatory instabilities unless some artificial diffusion is added to stabilize the method. In Section 5.4, we discuss the SUPG (5.4.1) and entropy viscosity (5.4.2) stabilization schemes used in the present work.

## 5.2 Time Integration Methods

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 [74] is a well-known, first-order, A-stable implicit time integration method. Given a generic semi-discrete equation in a form similar to (314)–(316),

$$\int_\Omega \left( \frac{\partial u^h}{\partial t} + G(u^h) \right) \phi_i \, \mathrm{d}\Omega = 0 \tag{320}$$

the backward Euler method results in the temporal discretization

$$\int_\Omega \left( \frac{u^{n+1} - u^n}{\Delta t} + G(u^{n+1}) \right) \phi_i \, \mathrm{d}\Omega = 0 \tag{321}$$

where $\Delta t$ 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 (321) 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

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \tag{322}$$

yields a leading-order truncation error term of the form

$$\begin{aligned} \left. \frac{\partial u}{\partial t} \right|_{t^{n+1}} &= \frac{u^{n+1} - u^n}{\Delta t} + \frac{\Delta t}{2} \left. \frac{\partial^2 u}{\partial t^2} \right|_{t^{n+1}} + \mathcal{O}(\Delta t^2) \\ &= \frac{u^{n+1} - u^n}{\Delta t} + \frac{a^2 \Delta t}{2} \left. \frac{\partial^2 u}{\partial x^2} \right|_{t^{n+1}} + \mathcal{O}(\Delta t^2) \end{aligned} \tag{323}$$

where (323) follows from differentiating the continuous equation (322) with respect to time:

$$\frac{\partial^2 u}{\partial t^2} = -a\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial x}\right) = -a\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial t}\right) = -a\frac{\partial}{\partial x}\left(-a\frac{\partial u}{\partial x}\right) = a^2\frac{\partial^2 u}{\partial x^2} \ . \tag{324}$$

Rearranging terms in (323) and adding $a\frac{\partial u}{\partial x}$ to both sides allows us to write

$$\frac{u^{n+1} - u^n}{\Delta t} + a\frac{\partial u}{\partial x} = \frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} - \frac{a^2\Delta t}{2}\frac{\partial^2 u}{\partial x^2} + \mathcal{O}(\Delta t^2) \tag{325}$$

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 (325) 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 (322). 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 [75] 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 $\Delta t$, the BDF2 method applied to the ordinary differential equation

$$\frac{\partial u}{\partial t} = f(t, u) \tag{326}$$

$$u(t = 0) = u_0 \tag{327}$$

yields the update step:

$$u^{n+1} = \frac{4}{3}u^n - \frac{1}{3}u^{n-1} + \frac{2}{3}\Delta t f(u^{n+1}, t^{n+1}) \tag{328}$$

Dividing through by $\frac{2}{3}\Delta t$, equation (328) can be alternatively written as

$$\frac{\frac{3}{2}u^{n+1} - 2u^n + \frac{1}{2}u^{n-1}}{\Delta t} = f(u^{n+1}, t^{n+1}) \tag{329}$$

The left-hand side of (329) 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 (321) to derive a fully-discrete system of equations:

$$\int_\Omega \left( \frac{\frac{3}{2}u^{n+1} - 2u^n + \frac{1}{2}u^{n-1}}{\Delta t} + G(u^{n+1}) \right) \phi_i \, \mathrm{d}\Omega = 0 \tag{330}$$

based on the semi-discrete equations (314)–(316).

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:

$$\int_\Omega \partial_t u \, \phi_i = \int_\Omega \left( \omega_0 u^{n+1} + \omega_1 u^n + \omega_2 u^{n-1} \right) \phi_i \, , \tag{331}$$

with

$$\omega_0 = \frac{2\Delta t^{n+1} + \Delta t^n}{\Delta t^{n+1} \left( \Delta t^{n+1} + \Delta t^n \right)} \tag{332}$$

$$\omega_1 = -\frac{\Delta t^{n+1} + \Delta t^n}{\Delta t^{n+1} \Delta t^n} \tag{333}$$

$$\omega_2 = \frac{\Delta t^{n+1}}{\Delta t^n \left( \Delta t^{n+1} + \Delta t^n \right)} \tag{334}$$

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 The PCICE Algorithm

The PCICE-FEM scheme [19, 20, 76, 77] is a finite element method (FEM) spatial discretization of the Pressure-Corrected Implicit Continuous-fluid Eulerian (PCICE) algo-

rithm. The PCICE algorithm defines the temporal discretization and hydrodynamic coupling procedure for the PCICE-FEM scheme. It is an advanced semi-implicit, mass-momentum coupled pressure-based scheme. The governing hydrodynamic equations for this scheme (for single phase) are the conservative form of the balance of momentum equations, mass conservation equation, and the total energy equation. An operator splitting process is performed between explicit and implicit operators of the semi-implicit equations to render the PCICE-FEM scheme in the class of predictor-corrector schemes. With the complete set of semi-implicit governing equations in the PCICE-FEM scheme cast in this form, an explicit predictor step and two semi-implicit pressure-correction steps with the elliptic pressure Poisson solution are performed to implicitly couple the momentum, density, and pressure. The result of this predictor- corrector formulation is that the pressure Poisson equation in the PCICE-FEM scheme is provided with sufficient internal energy information to avoid an iterative scheme.

To develop an all-speed/all-fluid simulation capability, the conservative form of governing equations are required. For reader convenience, the single-phase, variable area one-dimensional Euler equations with friction, wall-heating, and gravity forcing terms are recalled here by

$$\frac{\partial(A\rho)}{\partial t} + \frac{\partial(A\rho u)}{\partial x} = 0 \tag{335}$$

$$\frac{\partial(A\rho u)}{\partial t} + \frac{\partial A(\rho u^2 + p)}{\partial x} = p\frac{\partial A}{\partial x} + D_f + (A\rho)g_x \tag{336}$$

$$\frac{\partial(A\rho E)}{\partial t} + \frac{\partial(A\rho u)H}{\partial x} = H_w + (A\rho u)g_x \tag{337}$$

where $A\rho$, $A\rho u$, and $A\rho E$ are the conserved variables. $\rho$ is the fluid density, $u$ is the fluid velocity, $p$ is the thermodynamic pressure, $E \equiv e + \frac{1}{2}u^2$ is the specific total energy, $e$ is the internal energy, $D_f$ is the momentum friction drag, $H_w$ is the "wall" heat transfer, and $g_x$ is the strength of the gravity vector in the $x$-direction. The total enthalpy is given by

$$H = \frac{A\rho E + Ap}{A\rho}$$

These equations can be written in compact differential form as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial x} = \boldsymbol{Q} \tag{338}$$

where $U$ is the vector of the conservative variables, $\{A\rho, A\rho u, A\rho E\}$, $F$ is the convective

flux vector, and $Q$ is the source vector, given by

$$\boldsymbol{F} = \begin{bmatrix} A\rho u \\ (A\rho u)^2/A\rho + Ap \\ A\rho u H \end{bmatrix}, \qquad \boldsymbol{Q} = \begin{bmatrix} 0 \\ p\frac{\partial A}{\partial x} + D_f + (A\rho)g_x \\ H_w + (A\rho u)g_x \end{bmatrix}.$$

The PCICE algorithm is based upon the idea that both the balance of momentum and mass equations can be solved simultaneously to provide for a mathematically strong coupling between $Ap$-$A\rho u$-$A\rho$. The PCICE algorithm is basically composed of two phases, an explicit predictor with two passes through a semi-implicit corrector.

### 5.3.1 Explicit Predictor

The explicit predictor step of the PCICE algorithm is composed of a basic two-step Runge-Kutta time integration scheme, which is second-order accurate in time,

$$\boldsymbol{U}^{1/2} = \boldsymbol{U}^n - \frac{\Delta t}{2}\frac{\partial\left(\boldsymbol{F}_{proj}\right)^n}{\partial x} + \frac{\Delta t}{2}\boldsymbol{Q}^n_{proj} \tag{339}$$

$$\boldsymbol{U}^1 = \boldsymbol{U}^n - \Delta t\frac{\partial\left(\boldsymbol{F}_{proj}\right)^{1/2}}{\partial x} + \Delta t\boldsymbol{Q}^{1/2}_{proj} \tag{340}$$

where

$$\boldsymbol{F}_{proj} = \begin{bmatrix} A\rho u \\ (A\rho u)^2/A\rho \\ A\rho u(A\rho E)/A\rho \end{bmatrix}, \qquad \boldsymbol{Q}_{proj} = \begin{bmatrix} 0 \\ D_f + (A\rho)g_x \\ H_w + (A\rho u)g_x \end{bmatrix}$$

is the partial flux that ignores the pressure contribution in the explicit predictor phase of the PCICE algorithm. $U^1$ is the partial solution for the explicit predictor phase of the PCICE algorithm.

### 5.3.2 PCICE Algorithm Temporal Discretization

The target discretization for the PCICE algorithm temporal integration is based upon the second-order Crank-Nicholson scheme. For the conservation of mass,

$$A\rho^i - A\rho^n = -\frac{\Delta t}{2}\frac{\partial}{\partial x}\left[(A\rho u)^i + (A\rho u)^n\right] \tag{341}$$

the balance of momentum,

$$(A\rho u)^i - (A\rho u)^n = -\frac{\Delta t}{2}\frac{\partial}{\partial x}\left[\left(A\rho u^2\right)^{i-1} + \left(A\rho u^2\right)^n\right] - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left[A(p^i + p^n)\right] + \frac{\Delta t}{2}(p^i + p^n)\frac{\partial A}{\partial x} + \frac{\Delta t}{2}\left(Q_{\rho u}^{i-1} + Q_{\rho u}^n\right) \tag{342}$$

and the conservation of total energy,

$$(A\rho E)^i - (A\rho E)^n = -\frac{\Delta t}{2}\frac{\partial}{\partial x}\left[(A\rho u H)^i + (A\rho u H)^n\right] + \frac{\Delta t}{2}\left(Q_{\rho E}^{i-1} + Q_{\rho E}^n\right) \tag{343}$$

where

$$H^i = \frac{(A\rho E)^{i-1} + Ap^i}{A\rho^i}$$

$Q_{\rho u} = D_f + (A\rho)g_x$, and $Q_{\rho E} = H_w + (A\rho u)g_x$. In the above temporal discretization, $i$ is a correction index and $i = 1$ refers to the advanced time predictor solution, $U^1$, given by (340). The PCICE algorithm requires two semi-implicit corrections to achieve second-order accuracy in time. For the semi-implicit correction in equations (341)–(343), $i = 2, 3$.

### 5.3.3 Intermediate Momentum Solution

Directly substituting equation (342) into equation (341) will yield second order derivatives of the outer product contained in the balance of momentum convective flux terms and third-order derivatives for the divergence of the divergence of the viscous stress tensor. In order to avoid these difficult terms, an intermediate explicit momentum solution, composed of previous iterate terms in equation (342), is employed

$$S_A = (A\rho u)^n - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left[\left(A\rho u^2\right)^{i-1} + \left(A\rho u^2\right)^n\right] + \frac{\Delta t}{2}\left(Q_{\rho u}^{i-1} + Q_{\rho u}^n\right) . \tag{344}$$

Solving this intermediate step allows equation (342) to be re-written in terms of the intermediate momentum solution, $S_A$, by substituting equation (344) into equation (342), yielding

$$(A\rho u)^i = S_A - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left[A(p^i + p^n)\right] + \frac{\Delta t}{2}(p^i + p^n)\frac{\partial A}{\partial x} . \tag{345}$$

Equation (345) is the pressure correction equation for the momentum components once $p^i$ is known. As will be shown below, the explicit values of $S_A$ will be incorporated into the pressure Poisson equation, while the integral form of $S_A$ will be employed in the momentum component pressure correction for increased efficiency.

### 5.3.4 Pressure Poisson Equation

The first point to consider is what pressure variable form the pressure Poisson equation should solve. There are three obvious choices, the thermodynamic pressure, $p^i$, the change in pressure across a time step, $\delta p = p^i - p^n$, and a pressure correction variable, $p' = p^i - p^{i-1}$. We have found that a pressure correction variable provides the best performance and greatest ease in applying Dirichlet and von Neumann boundary conditions for solving with the Krylov subspace method.

Solving the pressure correction variable for $p^i$ and substituting into equation (345) yields

$$(A\rho u)^i = S_A - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left[A(p' + p^{i-1} + p^n)\right] + \frac{\Delta t}{2}(p' + p^{i-1} + p^n)\frac{\partial A}{\partial x} \ . \tag{346}$$

This form of the balance of momentum equations is mathematically identical to equation (342) and is in a form that is easily substituted into equation (341). Performing this substitution, which eliminates $(A\rho u)^i$ as an unknown, yields the basic form of the pressure Poisson equation

$$\begin{aligned} A\rho^i - A\rho^n = &-\frac{\Delta t}{2}\frac{\partial}{\partial x}\left[S_A + (A\rho u)^n\right] + \frac{\Delta t^2}{4}\frac{\partial^2}{\partial x^2}\left[A(p' + p^{i-1} + p^n)\right] \\ &- \frac{\Delta t^2}{4}\frac{\partial}{\partial x}\left[(p' + p^{i-1} + p^n)\frac{\partial A}{\partial x}\right] \ . \end{aligned} \tag{347}$$

Note that this equation is still a representation of the change in density composed of explicit convection and source terms with an implicit pressure correction. For the same reason (of efficience) that we collected explicit momentum terms into $S_A$, we now collect the explicit mass convection and pressure terms of equation (347) into a new mass variable

$$\begin{aligned} G = &-\frac{\Delta t}{2}\frac{\partial}{\partial x}\left[S_A + (A\rho u)^n\right] + \frac{\Delta t^2}{4}\frac{\partial^2}{\partial x^2}\left[A(p^{i-1} + p^n)\right] \\ &- \frac{\Delta t^2}{4}\frac{\partial}{\partial x}\left[(p^{i-1} + p^n)\frac{\partial A}{\partial x}\right] \end{aligned} \tag{348}$$

where $G$ differs from $S_A$ in that it will be used only in an integral sense. Equation (347) can now be represented in terms of $G$, yielding

$$A\rho^i - A\rho^n = G + \frac{\Delta t^2}{4} + \frac{\Delta t^2}{4}\frac{\partial^2}{\partial x^2}(Ap') - \frac{\Delta t^2}{4}\frac{\partial}{\partial x}\left(p'\frac{\partial A}{\partial x}\right) \ . \tag{349}$$

Equation (349) is the PCICE algorithm pressure correction equation for density once $p'$ is known.

Equation (349) resulted from the substitution of the balance of momentum equations into the mass conservation equation to eliminate the implicit momentum components as unknowns. This then leaves the situation where there is now one equation and two unknowns, i.e., $\rho^i$ and $p'$. To remedy this situation, we employ the equation of state to express $\rho^i$ in terms of $p'$. While the PCICE algorithm is not restricted to any specific equation of state, the algorithmic development here reflects pressure's functional dependence on density and internal energy

$$p = f(\rho, e)\,. \tag{350}$$

Differentiating equation (350) with respect to time and equating with the change in density term of equation (349),

$$A\rho^i - A\rho^n = \frac{A(p' + p^{i-1} - p^n)}{\left(\frac{\partial f}{\partial \rho}\right)_e} - \frac{\left(\frac{\partial f}{\partial e}\right)_\rho}{\left(\frac{\partial f}{\partial \rho}\right)_e} A(e^{i-1} - e^n) \tag{351}$$

yields the final form of the pressure Poisson equation for the PCICE algorithm

$$\begin{aligned}\frac{Ap'}{\left(\frac{\partial f}{\partial \rho}\right)_e} - \frac{\Delta t^2}{4}\frac{\partial^2}{\partial x^2}(Ap') - \frac{\Delta t^2}{4}\frac{\partial}{\partial x}\left(p'\frac{\partial A}{\partial x}\right) &= \frac{\left(\frac{\partial f}{\partial e}\right)_\rho}{\left(\frac{\partial f}{\partial \rho}\right)_e} A(e^{i-1} - e^n) \\ &\quad - \frac{A(p^{i-1} - p^n)}{\left(\frac{\partial f}{\partial \rho}\right)_e} + G\,.\end{aligned} \tag{352}$$

At this point, all the pressure correction equations have appeared in the development of the PCICE algorithm. With the solution of equation (352) for $p'$, the pressure correction equations, (349), (345), and (343), repeated below for conveience, are now solvable:

$$A\rho^i - A\rho^n = G + \frac{\Delta t^2}{4} + \frac{\Delta t^2}{4}\frac{\partial^2}{\partial x^2}(Ap') - \frac{\Delta t^2}{4}\frac{\partial}{\partial x}\left(p'\frac{\partial A}{\partial x}\right)\,. \tag{349}$$

$$(A\rho u)^i = S_A - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left[A(p^i + p^n)\right] + \frac{\Delta t}{2}(p^i + p^n)\frac{\partial A}{\partial x}\,. \tag{345}$$

$$(A\rho E)^i - (A\rho E)^n = -\frac{\Delta t}{2}\frac{\partial}{\partial x}\left[(A\rho uH)^i + (A\rho uH)^n\right] + \frac{\Delta t}{2}\left(Q^{i-1}_{\rho E} + Q^n_{\rho E}\right)\,. \tag{343}$$

## 5.4 Solution Stabilization Methods

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, is unstable when applied directly to hyperbolic systems of equations. It attempts to approximate potentially nonlinear discontinuous solutions with continuous, $\delta$-mollified solutions as nearly as possible with the functional space selected and element spacing chosen [78]. 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.

Currently available options of solution stabilization for RELAP-7 application include SUPG, entropy viscosity, and Lapidus methodologies. Plus, the low-order backward Euler time integration method described above is known to introduce an $\mathcal{O}(\Delta t)$ artificial viscosity through its discretization error. The main details of two schemes, the Streamline Upwind/Petrov Galerkin method (SUPG) (Section 5.4.1) and the entropy viscosity method (Section 5.4.2) are described in the following sections.

### 5.4.1 Streamline Upwind/Petrov-Galerkin Method

The Streamline Upwind/Petrov-Galerkin (SUPG) method is introduced by first writing (20)–(22) from Section 2.1 in system notation as

$$\boldsymbol{R}(\boldsymbol{V}) \equiv \frac{\partial \boldsymbol{V}}{\partial t} + \frac{\partial \boldsymbol{G}}{\partial x} - \boldsymbol{S} = \boldsymbol{0} \tag{353}$$

where

$$\boldsymbol{V} \equiv \begin{bmatrix} \rho A \\ \rho u A \\ \rho E A \end{bmatrix} \qquad \boldsymbol{G} \equiv \begin{bmatrix} \rho u A \\ (\rho u^2 + p)\, A \\ \rho u H A \end{bmatrix} \tag{354}$$

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, the weak form proceeds by dotting (353) with an admissible test function $\boldsymbol{W}$, integrating over the domain $\Omega$, and applying the divergence theorem. We then define

$$a(\boldsymbol{V}, \boldsymbol{W}) \equiv \int_\Omega \left( \frac{\partial \boldsymbol{V}}{\partial t} \cdot \boldsymbol{W} - \boldsymbol{G} \cdot \frac{\partial \boldsymbol{W}}{\partial x} - \boldsymbol{S} \cdot \boldsymbol{W} \right) \mathrm{d}\Omega + \int_\Gamma (\boldsymbol{G} \cdot \boldsymbol{W})\, \hat{n}_x \, \mathrm{d}\Gamma \tag{355}$$

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

$$\boldsymbol{U} \equiv \begin{bmatrix} \rho \\ \rho u \\ \rho E \end{bmatrix} \qquad \boldsymbol{F} \equiv \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u H \end{bmatrix} . \tag{356}$$

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

$$\frac{\partial \boldsymbol{F}}{\partial x} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}} \frac{\partial \boldsymbol{U}}{\partial x} \equiv \boldsymbol{A} \frac{\partial \boldsymbol{U}}{\partial x} \ . \tag{357}$$

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

$$\frac{\partial \boldsymbol{G}}{\partial x} = A \frac{\partial \boldsymbol{F}}{\partial x} + \frac{\partial A}{\partial x} \boldsymbol{F} \tag{358}$$

$$\boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} = \boldsymbol{A} \left( A \frac{\partial \boldsymbol{U}}{\partial x} + \frac{\partial A}{\partial x} \boldsymbol{U} \right) = A \frac{\partial \boldsymbol{F}}{\partial x} + \frac{\partial A}{\partial x} \boldsymbol{A}\boldsymbol{U} \tag{359}$$

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

$$\frac{\partial \boldsymbol{G}}{\partial x} = \boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} + (\boldsymbol{F} - \boldsymbol{A}\boldsymbol{U}) \frac{\partial A}{\partial x} \ . \tag{360}$$

Substituting (360) into (353) then gives

$$\widetilde{\boldsymbol{R}}(\boldsymbol{V}) \equiv \frac{\partial \boldsymbol{V}}{\partial t} + \boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} + (\boldsymbol{F} - \boldsymbol{A}\boldsymbol{U}) \frac{\partial A}{\partial x} - \boldsymbol{S} = \boldsymbol{0} \tag{361}$$

which is the so-called "quasi-linear" form of (353).

A few remarks about (361) 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 (361) 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

$$\boldsymbol{F} - \boldsymbol{A}\boldsymbol{U} = \begin{bmatrix} 0 \\ \hat{p} \\ u\hat{p} \end{bmatrix} \tag{362}$$

where

$$\hat{p} \equiv p - p_{,0}\rho - p_{,1}\rho u - p_{,2}\rho E \ . \tag{363}$$

For the stiffened gas equation of state, we can use the partial derivatives discussed in Section 2.2.3.4 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

$$a(\boldsymbol{V}, \boldsymbol{W}) + \sum_K \int_{\Omega_K} \boldsymbol{A}^T \frac{\partial \boldsymbol{W}}{\partial x} \cdot \boldsymbol{\tau}_{\text{SUPG}} \widetilde{\boldsymbol{R}}(\boldsymbol{V}) \, \mathrm{d}\Omega_K = 0 \tag{364}$$

for all admissible $\boldsymbol{W}$. In (364), $\boldsymbol{A}^T$ is the transpose of the flux Jacobian matrix, $\boldsymbol{\tau}_{\text{SUPG}}$ is in general a $3 \times 3$ matrix of solution-dependent stabilization parameters, and the second term of (364) is traditionally written as a sum of integrals over the finite elements $\Omega_K$ 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 (364) is said to be "consistent" in the following sense: if the true solution $\boldsymbol{V}$ (which satisfies (353) pointwise and the weak form (355)) is smooth, then it also satisfies (361), and therefore the additional stabilizing term is zero.

The "stabilizing" effects of (364) come specifically from the inviscid flux terms of the quasi-linear residual (361), i.e.

$$\begin{aligned} \int_{\Omega_K} \boldsymbol{A}^T \frac{\partial \boldsymbol{W}}{\partial x} \cdot \boldsymbol{\tau}_{\text{SUPG}} \widetilde{\boldsymbol{R}}(\boldsymbol{V}) \, \mathrm{d}\Omega_K &= \int_{\Omega_K} \boldsymbol{A}^T \frac{\partial \boldsymbol{W}}{\partial x} \cdot \boldsymbol{\tau}_{\text{SUPG}} \left( \ldots + \boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} + \ldots \right) \mathrm{d}\Omega_K \\ &= \int_{\Omega_K} \frac{\partial \boldsymbol{W}}{\partial x} \cdot \left( \boldsymbol{A} \boldsymbol{\tau}_{\text{SUPG}} \boldsymbol{A} \frac{\partial \boldsymbol{V}}{\partial x} \right) \mathrm{d}\Omega_K + \ldots \end{aligned} \tag{365}$$

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 [79] have shown that a possible form for the stabilization operator $\boldsymbol{\tau}_{\text{SUPG}}$ is

$$\boldsymbol{\tau}_{\text{SUPG}} = \frac{h}{2} |\boldsymbol{A}|^{-1} \tag{366}$$

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

$$|\boldsymbol{A}| \equiv \boldsymbol{P}|\boldsymbol{D}|\boldsymbol{P}^{-1} \tag{367}$$

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:

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 \\ p_{,0} - u^2 & p_{,1} + 2u & p_{,2} \\ u\left(p_{,0} - H\right) & up_{,1} + H & u\left(1 + p_{,2}\right) \end{bmatrix} . \tag{368}$$

The eigenvalues of the matrix defined in (368) are given by

$$\lambda_1 = u \tag{369}$$

$$\lambda_{2,3} = u + \frac{p_{,1} + up_{,2}}{2} \pm \frac{\left[4\left(p_{,0} + up_{,1} + Hp_{,2}\right) + \left(p_{,1} + up_{,2}\right)^2\right]^{1/2}}{2} \tag{370}$$

The eigenvalues (370) will be real (and hence the system will be hyperbolic) only if the term under the square root sign is $\geq 0$. It may be readily verified that, for a given equation of state, (370) reduces to $\lambda_{2,3} = u \pm c$, where $c$ is the local sound speed. In general, the form (370) 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

$$\boldsymbol{P} \equiv \begin{bmatrix} c_1 & c_3 & c_2 \\ \lambda_1 c_1 & \lambda_2 c_3 & \lambda_3 c_2 \\ 1 & 1 & 1 \end{bmatrix} \tag{371}$$

where

$$c_1 \equiv \frac{-p_{,2}}{p_{,0} + \lambda_1 p_{,1}} \tag{372}$$

$$c_j \equiv \frac{-\lambda_j}{d_j} \quad , \quad j = 2, 3 \tag{373}$$

$$d_j \equiv (H - u^2)(up_{,2} - \lambda_j) + u(p_{,0} - u^2) \quad , \quad j = 2, 3 \ . \tag{374}$$

Its inverse is

$$\boldsymbol{P}^{-1} \equiv \frac{1}{\det \boldsymbol{P}} \begin{bmatrix} \lambda_2 c_3 - \lambda_3 c_2 & c_2 - c_3 & c_2 c_3 (\lambda_3 - \lambda_2) \\ \lambda_3 c_2 - \lambda_1 c_1 & c_1 - c_2 & c_1 c_2 (\lambda_1 - \lambda_3) \\ \lambda_1 c_1 - \lambda_2 c_3 & c_3 - c_1 & c_1 c_3 (\lambda_2 - \lambda_1) \end{bmatrix} \tag{375}$$

where

$$\det \boldsymbol{P} \equiv c_1 (c_2 - c_3) \lambda_1 + c_3 (c_1 - c_2) \lambda_2 + c_2 (c_3 - c_1) \lambda_3 \ . \tag{376}$$

The preceding discussion provides all the information necessary to implement the SUPG scheme (364). 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.

### 5.4.2 Entropy Viscosity Method

As an available option, RELAP-7 employs a new technique, introduced recently [80–83], which requires the addition of artificial dissipation terms to the equations while ensuring that the physical entropy minimum principle remains satisfied. Additional details regarding its application to the 7-equation two-phase model and to low Mach number flows are directly based upon INL-sponsored, independent research of Delchini [84]. 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 a simplified 7-equation two-phase model, the entropy viscosity method is available for use with single-phase flow systems as well.

The simplified 7-equation two-phase model equation system is:

$$\frac{\partial}{\partial t}\left(\alpha_k A\right) + u_I A \frac{\partial \alpha_k}{\partial x} = A\mu\left(P_k - P_j\right) \tag{377}$$

$$\frac{\partial}{\partial t}\left(\alpha_k \rho_k A\right) + \frac{\partial}{\partial x}\left(\alpha_k \rho_k u_k A\right) = 0 \tag{378}$$

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_k \rho_k u_k A\right) + \frac{\partial}{\partial x}\left[\alpha_k A\left(\rho_k u_k^2 + P_k\right)\right] &= \alpha_k P_k \frac{\partial A}{\partial x} + P_I A \frac{\partial \alpha_k}{\partial x} \\ &\quad + A\lambda\left(u_j - u_k\right)\end{aligned} \tag{379}$$

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_k \rho_k E_k A\right) + \frac{\partial}{\partial x}\left[\alpha_k A u_k\left(\rho_k E_k + P_k\right)\right] &= P_I u_I A \frac{\partial \alpha_k}{\partial x} - \mu \bar{P}_I\left(P_k - P_j\right) \\ &\quad + \bar{u}_I A\lambda\left(u_j - u_k\right)\end{aligned} \tag{380}$$

where $\rho_k$, $u_k$, $E_k$ and $P_k$ are the density, the velocity, the specific total energy and the pressure of $k^{th}$ phase, respectively. The pressure and velocity relaxation parameters are denoted by $\mu$ and $\lambda$, respectively. The variables with index $(\cdot)_I$ correspond to the interfacial variables and a definition for those can be found above (and in [12]). The cross-sectional area $A$ is only function of space, so $\frac{\partial A}{\partial t} = 0$. In [12], the entropy equation is derived for each phase, by assuming that there exists a phasic entropy function $s_k$ that depends upon the density $\rho_k$ and the specific internal energy $e_k$:

$$\begin{aligned}\rho_k A\left(\frac{\partial s_k}{\partial t} + u_k \frac{\partial s_k}{\partial x}\right) &= \frac{Z_j}{Z_j + Z_k}\lambda\left(u_k - u_j\right)^2 + \frac{Z_k}{Z_j + Z_k}\mu\left(P_k - P_j\right)^2 \\ &\quad + \frac{Z_k}{Z_j + Z_k}\left|\frac{\partial \alpha}{\partial x}\right|\left[Z_j\left(u_k - u_j\right) + \operatorname{sgn}\left(\frac{\partial \alpha}{\partial x}\right)\left(P_k - P_j\right)\right]^2\end{aligned} \tag{381}$$

where $Z_j = \rho_j c_j$ and $c_j$ represents the acoustic impedance and the speed of sound of phase $j$, respectively. The symbol $\operatorname{sgn}(\cdot)$ denotes a function that returns the sign of the quantity $(\cdot)$.

From (381), it is clear that the entropy minimum principle is satisfied since the right-hand side is only composed of positive terms. In the remainder of this exposition all of the source terms (right-hand sides of (377)–(380)) are dropped in order to simplify the derivation. It will not affect the final result, provided that the definitions of the entropy residual (below) are amended to include contributions of any heat source/sink terms [85], since all of the source terms combine in a sum of positive terms when deriving the entropy function. The phase index $k$ is also dropped.

To apply the entropy viscosity method, appropriate dissipative terms are added to each

of the equations as follows

$$\frac{\partial}{\partial t}(\alpha A) + u_I A \frac{\partial \alpha}{\partial x} = \frac{\partial}{\partial x}(\alpha A l) \tag{382}$$

$$\frac{\partial}{\partial t}(\alpha \rho A) + \frac{\partial}{\partial x}(\alpha \rho u A) = \frac{\partial}{\partial x}(\alpha A f) \tag{383}$$

$$\frac{\partial}{\partial t}(\alpha \rho u A) + \frac{\partial}{\partial x}\left[\alpha A\left(\rho u^2 + P\right)\right] = \alpha P \frac{\partial A}{\partial x} + \frac{\partial}{\partial x}(\alpha A g) \tag{384}$$

$$\frac{\partial}{\partial t}(\alpha \rho E A) + \frac{\partial}{\partial x}\left[\alpha A u\left(\rho E + P\right)\right] = \frac{\partial}{\partial x}\left[\alpha A\left(h + u g\right)\right] \tag{385}$$

where $f$, $g$, $h$ and $l$ are the dissipative terms. By adding these dissipative terms to each equation, the entropy equation gets modified; extra terms will appear in the right-hand side that are a function of the dissipative terms. The sign of these new terms needs to be studied in order to conserve positivity of the right-hand side. This is achieved by the following steps:

- Recast the system of equation given in (382)–(385) in terms of the primitive variables $(\alpha, \rho, u, e)$ (we only account for the dissipative terms here).
- Derive the entropy equation by using the chain rule

$$\frac{ds}{dt} = s_\rho \frac{d\rho}{dt} + s_e \frac{de}{dt} \tag{386}$$

  where $\frac{d}{dt}$ is the material derivative. The terms $s_e$ and $s_\rho$ denote the partial derivative of the entropy $s$ with respect to $e$ and $\rho$, respectively.
- Isolate the terms of interest and choose an appropriate expression for each of the dissipative terms in order to ensure positivity of the right-hand side.

The first step consists of recasting the system of equations in terms of the primitive vari-

ables as

$$A\frac{\partial \alpha}{\partial t} + u_I A\frac{\partial \alpha}{\partial x} = \frac{\partial}{\partial x}\left(\alpha A l\right) \tag{387}$$

$$\alpha A\left(\frac{\partial \rho}{\partial t} + u\frac{\partial \rho}{\partial x}\right) + \rho\alpha\frac{\partial}{\partial x}\left(uA\right) + \Gamma = \frac{\partial}{\partial x}\left(\alpha A f\right) - \rho\frac{\partial}{\partial x}\left(\alpha A l\right) \tag{388}$$

$$\begin{aligned}\alpha\rho A\left(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial x}\left(\alpha P A\right) &= \alpha P\frac{\partial A}{\partial x} + \frac{\partial}{\partial x}\left(\alpha A g\right)\\ &\quad - u\frac{\partial}{\partial x}\left(\alpha A f\right)\end{aligned} \tag{389}$$

$$\begin{aligned}\alpha\rho A\left(\frac{\partial e}{\partial t} + u\frac{\partial e}{\partial x}\right) + \alpha P A\frac{\partial u}{\partial x} + \alpha u P\frac{\partial A}{\partial x} &= \frac{\partial}{\partial x}\left(\alpha A h\right) + \alpha A g\frac{\partial u}{\partial x}\\ &\quad + \left(\frac{u^2}{2} - e\right)\frac{\partial}{\partial x}\left(\alpha A f\right)\end{aligned} \tag{390}$$

where $\Gamma = \rho A\left(u - u_I\right)\frac{\partial \alpha}{\partial x}$ has been used. The function $\Gamma$ can be ignored since it is used to get the right-hand side of (381). For the second step, the continuity and internal energy equations can be combined using the chain rule given in (386) to obtain the entropy equation

$$\begin{aligned}\alpha\rho A\frac{ds}{dt} + \alpha\left(\rho s_\rho + P s_e\right)\left(A\frac{\partial u}{\partial x} + u\frac{\partial A}{\partial x}\right) &=\\ s_e\left[\frac{\partial}{\partial x}\left(\alpha A h\right) + \alpha A g\frac{\partial u}{\partial x} + \left(\frac{u^2}{2} - e\right)\frac{\partial}{\partial x}\left(\alpha A f\right)\right]&\\ + \rho s_\rho\left[\frac{\partial}{\partial x}\left(\alpha A f\right) - \rho\frac{\partial}{\partial x}\left(\alpha A l\right)\right]&.\end{aligned} \tag{391}$$

The last step is a little more involved. For clarity of a first approach, the entropy equation, (391), can be split into four separate terms for further consideration. The first term of the left-hand side is the Lagrangian (or material) derivative of the entropy function $s$. It does not need to be modified since its sign has to be determined by looking at the other terms. The second term in the left-hand side is usually set to zero by assuming that $\rho s_\rho + P s_e = 0$. Any other alternative would require the term $\rho^2 s_\rho + P s_e$ to be function of the velocity $u$ or its derivatives, and the cross-section $A$, in order to be able to determine its sign. In addition, any entropy function obeying to the relation $\rho s_\rho + P s_e = 0$, also

obeys the second thermodynamic law:

$$Tds = de - \frac{P}{\rho^2} d\rho$$
$$\Downarrow$$
$$s_e = \frac{1}{T} \geq 0, \quad s_\rho = -s_e \frac{P}{\rho^2} \tag{392}$$

where $T$ is the fluid temperature.

The right-hand side of (391) is more difficult to handle. It requires further assumptions in the definition of the dissipative terms $h$ and $g$. The first term of the right-hand side can be simplified by using the following expression for the dissipative terms $h$ and $g$

$$g = \rho\mu \frac{\partial u}{\partial x} + uf \tag{393}$$

$$h = \tilde{h} - \frac{u^2}{2} f \tag{394}$$

which results in

$$s_e \left[ \frac{\partial}{\partial x} (\alpha A h) + \alpha A g \frac{\partial u}{\partial x} + \left( \frac{u^2}{2} - e \right) \frac{\partial}{\partial x} (\alpha A f) \right] = s_e \left[ \frac{\partial}{\partial x} \left( \alpha A \tilde{h} \right) - e \frac{\partial}{\partial x} (\alpha A f) \right] + s_e \rho\mu \left( \frac{\partial u}{\partial x} \right)^2 \tag{395}$$

where $\mu$ is a positive viscosity coefficient, and $\tilde{h}$ is a new dissipative term. In (395), it is noted that the term $s_e \rho\mu \left( \frac{\partial}{\partial x} u \right)^2$ is always positive and does not need any further modification. Thus, it remains to determine the sign of the other term $s_e \left[ \frac{\partial}{\partial x} \left( \alpha A \tilde{h} \right) - e \frac{\partial}{\partial x} (\alpha A f) \right]$, along with $\rho s_\rho \left[ \frac{\partial}{\partial x} (\alpha A f) - \rho \frac{\partial}{\partial x} (\alpha A l) \right]$, that are now independent of the velocity $u$. Then, we define the variable $\mathcal{R}$ as

$$\mathcal{R} \equiv s_e \left[ \frac{\partial}{\partial x} \left( \alpha A \tilde{h} \right) - e \frac{\partial}{\partial x} (\alpha A f) \right] + \rho s_\rho \left[ \frac{\partial}{\partial x} (\alpha A f) - \rho \frac{\partial}{\partial x} (\alpha A l) \right] . \tag{396}$$

The reader is reminded that all of the above steps are valid for any phase of the system under consideration, i.e. each phase is considered independently of the other, and that the objective is to prove the entropy minimum principle. In order to determine the sign of $\mathcal{R}$ in (396), the dissipative term $f$ is split into two other dissipative terms: $f = f_1 + f_2$. Then,

$\mathcal{R}$ is recast as

$$\mathcal{R} = \underbrace{s_e\left(\frac{\partial}{\partial x}\left(\alpha A\tilde{h}\right) - e\left(\frac{\partial}{\partial x}\left(\alpha A f_1\right) + \frac{\partial}{\partial x}\left(\alpha A f_2\right)\right)\right) + \rho s_\rho \frac{\partial}{\partial x}\left(\alpha A f_1\right)}_{\mathcal{R}_1}$$
$$\underbrace{- e s_e \frac{\partial}{\partial x}\left(\alpha A f_2\right) + \rho s_\rho\left(\frac{\partial}{\partial x}\left(\alpha A f_2\right) - \rho\frac{\partial}{\partial x}\left(\alpha A l\right)\right)}_{\mathcal{R}_2} . \tag{397}$$

Integration by parts yields:

$$\mathcal{R} = \underbrace{\frac{\partial}{\partial x}\left[\alpha A\left(s_e\tilde{h} + \left(\rho s_\rho - e s_e\right) f_1\right)\right] - \alpha A\tilde{h}\frac{\partial}{\partial x}\left(s_e\right) - \alpha A f_1\left(\rho s_\rho - e s_e\right)}_{\mathcal{R}_1}$$
$$+ \underbrace{\frac{\partial}{\partial x}\left[\alpha A\left(\left(\rho s_\rho - e s_e\right) f_2 - \rho^2 s_\rho l\right)\right] - \alpha A f_2\frac{\partial}{\partial x}\left(\rho s_\rho - e s_e\right) + \alpha A l\frac{\partial}{\partial x}\left(\rho^2 s_\rho\right)}_{\mathcal{R}_2} . \tag{398}$$

The $\mathcal{R}_1$ term is actually identical to what is obtained in the single-phase entropy equation [86]. Positivity of this term is ensured by assuming concavity of the entropy function $s$ as a function of $\frac{1}{\rho}$ and $e$ and using the following definition for $f_1$ and $\tilde{h}$

$$\tilde{h} = \kappa\frac{\partial}{\partial x}\left(\rho e\right) \tag{399}$$

$$f_1 = \kappa\frac{\partial \rho}{\partial x} \tag{400}$$

where $\kappa$ is a positive viscosity coefficient. It remains now to determine the sign of $\mathcal{R}_2$ in (398)

$$\mathcal{R}_2 \equiv \frac{\partial}{\partial x}\left[\alpha A\left(\left(\rho s_\rho - e s_e\right) f_2 - \rho^2 s_\rho l\right)\right] - \alpha A f_2\frac{\partial}{\partial x}\left(\rho s_\rho - e s_e\right) + \alpha A l\frac{\partial}{\partial x}\left(\rho^2 s_\rho\right) .$$

The term $\mathcal{R}_2$ can be simplified by assuming that the dissipative terms $f_2$ and $l$ are related to each other by

$$f_2 = \frac{\rho^2 s_\rho}{\rho s_\rho - e s_e} l \tag{401}$$

which leads to the following expression for $\mathcal{R}_2$

$$\mathcal{R}_2 = \alpha A \rho^2 s_\rho l\left(-\frac{1}{\rho s_\rho - e s_e}\frac{\partial}{\partial x}\left(\rho s_\rho - e s_e\right) + \frac{1}{\rho^2 s_\rho}\frac{\partial}{\partial x}\left(\rho^2 s_\rho\right)\right) . \tag{402}$$

To determine the sign of (402), it is required for $l$ to be a function of the density $\rho$ and the partial derivative $s_\rho$. A *sufficient condition* is to set

$$s_\rho l \propto -\frac{1}{\rho s_\rho - e s_e}\frac{\partial}{\partial x}\left(\rho s_\rho - e s_e\right) + \frac{1}{\rho^2 s_\rho}\frac{\partial}{\partial x}\left(\rho^2 s_\rho\right) \tag{403}$$

which yields

$$\mathcal{R}_2 = \alpha A \rho \beta \left(-\frac{1}{\rho s_\rho - e s_e}\frac{\partial}{\partial x}\left(\rho s_\rho - e s_e\right) + \frac{1}{\rho^2 s_\rho}\frac{\partial}{\partial x}\left(\rho^2 s_\rho\right)\right)^2 \tag{404}$$

with

$$l = \beta \frac{1}{\rho s_\rho}\left(-\frac{1}{\rho s_\rho - e s_e}\frac{\partial}{\partial x}\left(\rho s_\rho - e s_e\right) + \frac{1}{\rho^2 s_\rho}\frac{\partial}{\partial x}\left(\rho^2 s_\rho\right)\right) \tag{405}$$

where $\beta$ is a positive viscosity coefficient. The definition of the dissipative term $l$ can be simplified further to get the final expression

$$l = \beta \frac{1}{\rho s_\rho}\left(\frac{\rho s_\rho}{\rho s_\rho - e s_e}\frac{\partial}{\partial x}\left(\frac{\rho s_\rho}{e s_e}\right) + \frac{1}{\rho}\frac{\partial \rho}{\partial x}\right) \tag{406}$$

or in a more friendly form using (392):

$$l = \beta T\left[\frac{\rho}{P + \rho e}\frac{\partial}{\partial x}\left(\frac{P}{\rho e}\right) - \frac{1}{P}\frac{\partial \rho}{\partial x}\right] . \tag{407}$$

In summary, the dissipative terms are defined as

$$l = \beta T\left[\frac{\rho}{P + \rho e}\frac{\partial}{\partial x}\left(\frac{P}{\rho e}\right) - \frac{1}{P}\frac{\partial \rho}{\partial x}\right] \tag{408}$$

$$f = \kappa \frac{\partial \rho}{\partial x} + \frac{\rho^2 s_\rho}{\rho s_\rho - e s_e} l \tag{409}$$

$$g = \rho \mu \frac{\partial u}{\partial x} + u f \tag{410}$$

$$h = \kappa \frac{\partial}{\partial x}\left(\rho e\right) - \frac{u^2}{2} f + u g \ . \tag{411}$$

At this point, some remarks are in order:

1. The dissipative term $l$ requires the definition of a new viscosity coefficient $\beta$. In theory this viscosity coefficient is independent of the other viscosity coefficients $\mu$ and $\kappa$. Its definition should account for the eigenvalue associated with the void fraction equation $u_I$. In addition, an entropy residual can be determined by analogy to Burger's equation. It is noted, however, that the eigenvalue $u_I$ can be discontinuous since its definition involves the sign of the void fraction gradient, which makes the theory more challenging. For simplicity, we ignore this aspect of the theory in this report.

2. The dissipative term $f$ is a function of $l$. Thus, all of the other dissipative terms are also functions of $l$.

3. The partial derivatives $s_e$ and $s_\rho$ can be computed using the definition provided in (392), and are functions of the thermodynamic variables: pressure, temperature and density.

4. All of the dissipative terms are chosen to be proportional to the the void fraction $\alpha$ and the cross-sectional area $A$. For instance, $\alpha A l$ is the flux of the dissipative term in the void fraction equation through the area seen by the phase $\alpha A$. When one of the phases disappears, the dissipative terms must to go to zero for consistency. On the other hand, when $\alpha$ goes to one, the single-phase equation must be recovered. This statement gives a condition for defining the viscosity coefficient $\beta$.

5. The seven equation model can be used to simulate situations in which the phases do not interact: mass and heat exchange do not occur. Then, the void fraction of each phase is expected to remain constant even though the other characteristic variables of the fluid can vary: $\frac{\partial}{\partial t}(A\alpha)+u_I A\frac{\partial \alpha}{\partial x}=0$. The dissipative term in the void fraction equation is a function of the thermodynamic variables and, thus, may cause the void fraction to change. This situation can be avoided by carefully defining the viscosity coefficient $\beta$ to be proportional to the entropy residual for $\eta(\alpha)$ that is known to be zero when the void fraction is a constant:

$$\begin{aligned}
\frac{\partial}{\partial t}(A\alpha)+u_I A\frac{\partial \alpha}{\partial x}&=0\\
\eta_\alpha\left[\frac{\partial}{\partial t}(A\alpha)+u_I A\frac{\partial \alpha}{\partial x}\right]&=0\\
A\frac{\partial \eta}{\partial t}+u_I A\frac{\partial \eta}{\partial x}&=0
\end{aligned}$$

where $\eta_\alpha$ is the derivative of $\eta$ with respect to $\alpha$, and $A$ is assumed to be space-dependent only.

It remains now to specify the coefficients $\kappa$, $\mu$, and $\beta$ in the artificial viscous terms of our balance equation system. In the current version of the method [85], $\kappa$ and $\mu$ are set for each phase as though that phase was a single phase only. Furthermore, they are set equal in each phase; that is $\kappa_k = \mu_k$ for $k = liq, vap$. The current definition includes an upper bound coefficient that will give a first order viscosity, denoted with subscript $max$, and a coefficient that will give a high-order viscosity, denoted with subscript $e$. The first-order viscosity coefficients $\kappa_{max,k}$ and $\mu_{max,k}$ are proportional respectively to the largest local, phasic eigenvalue $|u_k| + c_k$ and is equivalent to a first-order upwind scheme

$$\kappa_{max,k}(x,t) = \mu_{max,k}(x,t) = \frac{h}{2}(|u_k| + c_k) \tag{412}$$

where $h$ is the grid size. Such schemes are known to be monotone, but overly dissipative. The higher-order viscosity coefficients $\kappa_{k,e}$ and $\mu_{k,e}$ are set proportional to the entropy production that is evaluated by computing the local entropy residual $D_{e,k}$ as

$$D_{e,k}(x,t) = \frac{\partial s_k}{\partial t} + u_k \frac{\partial s_k}{\partial x} = \frac{s_{k,e}}{P_{k,e}} \underbrace{\left( \frac{d_k P_k}{dt} - c_k^2 \frac{d_k \rho_k}{dt} \right)}_{\tilde{D}_{e,k}(x,t)} \tag{413}$$

where $\frac{d_k}{dt}$ denotes the material $k$ or total-$k$ derivative, $P_{k,e}$ is the partial derivative of the phase-$k$ pressure $P_k$ with respect to the phase-$k$ internal energy $e_k$, and $s_{k,e}$ is the partial derivative of the phase-$k$ entropy with respect to the phase-$k$ internal energy $e_k$. The expression for the entropy residual has been written as a function of pressure and density. Because $D_{k,e}$ and $\tilde{D}_{k,e}$ are proportional to each other, the definition of the viscosity coefficients $\kappa_{k,e}$ and $\mu_{k,e}$ are written to depend upon $\tilde{D}_{e,k}$ as follows

$$\kappa_{k,e}(x,t) = \mu_{k,e}(x,t) = h^2 \frac{\max(|\tilde{D}_{e,k}(x,t)|,\ J)}{(1 - M_k)\rho_k c_k^2 + M_k \rho_k |u_k|^2} \tag{414}$$

where $M_k$ is the phasic Mach number. The denominator has the same dimensions as pressure and is designed to ensure consistency when dealing with low Mach number flows. The jump $J$ is chosen to be proportional to the jump of the pressure and density gradients at the interfaces

$$J_{i+1/2} = |u_k|_{i+1/2} \max \left( \left[\!\left[ \frac{\partial P_k}{\partial x} \right]\!\right]_{i+1/2}, c_k^2 \left[\!\left[ \frac{\partial \rho_k}{\partial x} \right]\!\right]_{i+1/2} \right) \tag{415}$$

where

$$[\![q]\!]_{i+1/2} \equiv |q_i - q_{i+1}| \tag{416}$$

denotes the jump in a given quantity $q$, and $i + 1/2$ denotes the interface between cells $i$ and $i + 1$.

Lastly, the $\beta$ terms must be specified. From the fourth and fifth remarks above, $\beta$ for each phase must go to zero as either of the volume fractions $\alpha_{liq}$ or $\alpha_{vap}$ approach zero. The current model uses the simple specification

$$\beta_k = \alpha_{liq}\alpha_{vap}\kappa_{k,e} \tag{417}$$

to satisfy these restrictions.

## 5.5 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 [6].

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 are first written as

$$\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0} \tag{418}$$

where $\boldsymbol{F}$ represents the nonlinear equation system and $\boldsymbol{u}$ is the solution vector. Newton's method requires an initial guess, $\boldsymbol{u}^0$, 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

$$\boldsymbol{r}^k \equiv \boldsymbol{F}(\boldsymbol{u}^k)\,. \tag{419}$$

Clearly if $\boldsymbol{u}^k$ satisfies (418) *exactly*, the $k^{th}$ residual will be zero. To update the solution vector, the following equation is solved for the update vector, $\delta\boldsymbol{u}^{k+1}$:

$$\boldsymbol{J}(\boldsymbol{u}^k)\delta\boldsymbol{u}^{k+1} = -\boldsymbol{r}^k \tag{420}$$

where $\boldsymbol{J}(\boldsymbol{u}^k)$ is the Jacobian matrix evaluated at $\boldsymbol{u}^k$. In index notation,

$$J_{ij} \equiv \frac{\partial F_i}{\partial u_j} \ . \tag{421}$$

After $\delta\boldsymbol{u}^{k+1}$ is obtained, the $(k+1)^{st}$ solution iterate is computed by

$$\boldsymbol{u}^{k+1} = \boldsymbol{u}^k + \delta\boldsymbol{u}^{k+1} \ . \tag{422}$$

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 (420) 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 (420) can be expressed as (using right preconditioning as an example),

$$\boldsymbol{J}^k\boldsymbol{P}^{-1}\left(\boldsymbol{P}\delta\boldsymbol{u}^{k+1}\right) = -\boldsymbol{r}^k \tag{423}$$

where $\boldsymbol{P}$ is the preconditioning matrix. In the approach current used in RELAP-7, an analytical Jacobian matrix is computed according to (421), and passed to the underlying numerical solver library as the matrix $\boldsymbol{P}$ for preconditioning purposes.

# 6 Component Models

The RELAP-7 code is an advanced system analysis tool based on components to represent the major physical processes in the 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). There are three main types of components developed in RELAP-7: (1) one-dimensional (1-D) components describing the geometry of the reactor system, (2) zero-dimensional (0-D) components for setting boundary conditions, and (3) 0-D components for connecting 1-D components.

## 6.1 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 fluids dynamic models described in Chapter 2 are available in the Pipe component which includes the isothermal flow model, single-phase non-isothermal flow model, fully nonequilibrium 7-equation two-phase model, and the much simpler homogeneous equilibrium two-phase flow model.

## 6.2 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.

## 6.3 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.

## 6.4 HeatExchanger

A Heat Exchanger component is a combination of two pipes with a solid wall in between. Similar to the CoreChannel model, the fluids flow model and conjugate heat transfer model are fully coupled. More complicated and realistic steam generator component will be developed in the future.

## 6.5 Junction/Branch

### 6.5.1 Lagrange Multiplier Based Junction Model

This model is implemented by the `FlowJunction` component. It uses a 1D mortar finite element method to couple together the pipes which begin/end in the junction. The mortar method is implemented using Lagrange multipliers. For the sake of simplicity, we don't give the full details of the mortar method here. Instead we list only the constraints that are enforced by method:

$$
\begin{aligned}
g_0: &\quad \sum_i \rho_i u_i A_i \hat{n}_{x_i} = 0 \\
g_1: &\quad \sum_i (p_i + \rho_i u_i^2) A_i \hat{n}_{x_i} - \frac{\sum_i \hat{n}_{x_i} A_i \sum_i p_i A_i}{\sum_i A_i} - \frac{1}{2}\mathrm{sgn}(u_i) K_i \rho_i u_i^2 A_i = 0 \\
g_2: &\quad \sum_i \rho_i H_i u_i \hat{n}_{x_i} A_i = 0 \ .
\end{aligned}
$$

For more details on the mortar method see [87].

### 6.5.2 Volume Branch Model

The volume branch 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 `VolumeBranch` component are

$$\frac{d(\rho_{vb}V_{vb})}{dt} + \sum_{i=1}^{N}(\rho\boldsymbol{u})_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{424}$$

$$\frac{d((\rho e)_{vb}V_{vb})}{dt} + \sum_{i=1}^{N}((\rho e)_i + P_i)\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{425}$$

where $\rho_{vb}$ and $V_{vb}$ are the density and volume of the `VolumeBranch` 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 `VolumeBranch` 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 `VolumeBranch` 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 `VolumeBranch` component.

$$P_i = P_{vb} + \Delta P_{acc} + s\Delta P_{form} + \Delta P_g \tag{426}$$

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 `VolumeBranch` component while $s = -1$ if the fluids flow out of the `VolumeBranch` 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 $\Delta H$ is the height difference between the elevation of the center of the `VolumeBranch` 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.

## 6.6 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:

$$(\rho u)_1 A_1 \hat{n}_1 + (\rho u)_2 A_2 \hat{n}_2 = 0\,, \tag{427}$$

where $\rho u$ 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

$$e_{out} = e_{in} = e_J\,. \tag{428}$$

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

$$p_1 = (p_J + \frac{1}{2}\rho_J u_J^2) - \rho_1 g H - \frac{1}{2}\rho_1 u_1^2 \tag{429}$$

$$p_2 = (p_J + \frac{1}{2}\rho_J u_J^2) - \frac{1}{2}\rho_2 u_2^2 \tag{430}$$

where

$$\rho_J = \rho(e_J, p_J) \tag{431}$$

$$u_J = \frac{\rho_1 u_1 A_1}{\rho_J A_J} \tag{432}$$

$g$ 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 coastdown, or $H$ can be calculated by coupling with a shaft work, i.e., provided by a turbine,

$$H = \frac{\dot{W}_t}{\rho_1 u_1 A_1 g} \tag{433}$$

where $\dot{W}_t$ is the turbine shaft power. Downstream total energy is calculated by

$$\rho E_{bc} = \rho_{bc}\left(e_J + \frac{1}{2}u_{bc}^2\right) . \tag{434}$$

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

$$p_1 = (p_J + \frac{1}{2}\rho_J u_J^2) - \frac{1}{2}(1 + K_1)\rho_1 u_1^2 \tag{435}$$

$$p_2 = (p_J + \frac{1}{2}\rho_J u_J^2) - \frac{1}{2}(1 - K_2)\rho_2 u_2^2 . \tag{436}$$

The pump can also be simulated as a time dependent junction with given mass flow rate as a function of time.

## 6.7 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 [22], 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 or branch models in current reactor safety system codes such as RELAP5 [22], TRAC [88], and TRACE [89]. 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. 4 shows one example of turbine characteristics curves [2]. 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.

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. 5 shows the $T$-$s$ diagram for a thermodynamical 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.

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), $\rho_2$ (outlet density)
- turbine shaft power: $\dot{W}_t$

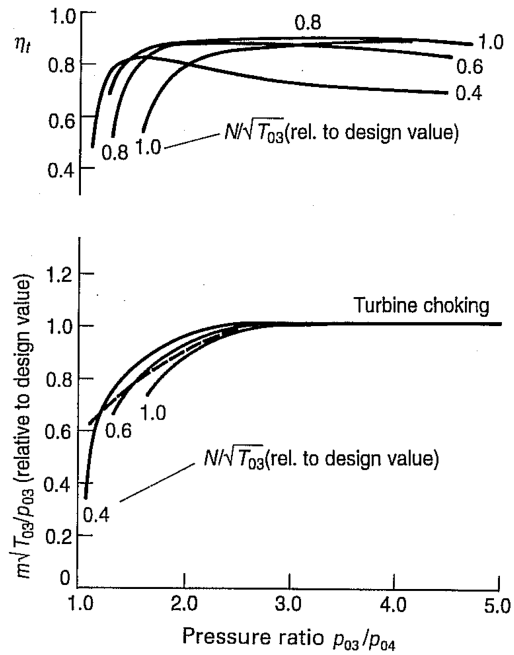


**Figure 4.** Turbine characteristics (credit of Saravanamuttoo, Rogers, and Cohen [2]).

four equations are needed to close the system. The first one is mass conservation Eq. (427). 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

$$\frac{\frac{\dot{m}\sqrt{T_{01}}}{p_{01}}}{\frac{\dot{m}_{max}\sqrt{T_{01r}}}{p_{01r}}} = f(p_{01}/p_{02})\,. \tag{437}$$

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 [90], the curve for a HP (High Pressure) steam turbine is defined as

$$f(p_{01}/p_{02}) = \sqrt{1 - \left(\frac{p_{02}}{p_{01}}\right)^2}\,. \tag{438}$$

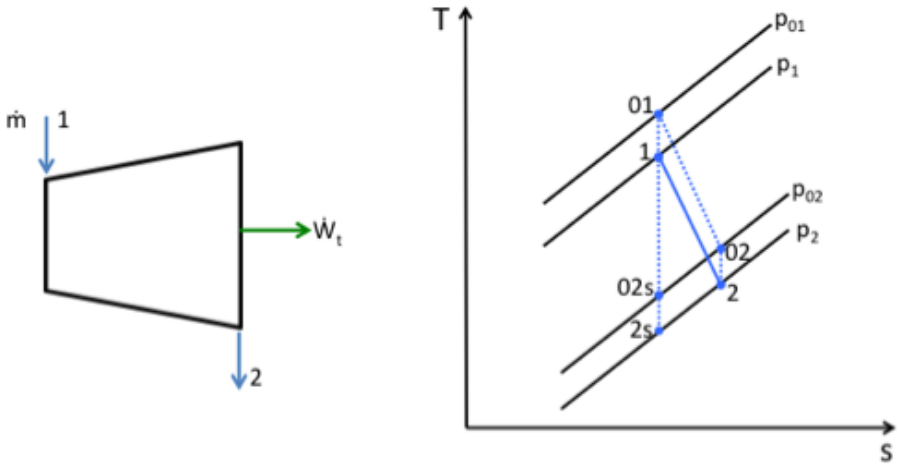


**Figure 5.** $T$-$s$ diagram for a turbine.

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

$$f(p_{01}/p_{02}) = \tanh\left(\beta\left(\frac{p_{02}}{p_{01}} - 1\right)\right) \tag{439}$$

where $\beta$ is a constant and is calculated by the following formula

$$\tanh\left(\beta\left(\frac{p_{01r}}{p_{012}} - 1\right)\right) = \frac{\dot{m}_r}{\dot{m}_{max}} \ . \tag{440}$$

Therefore, $\beta$ 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

$$\eta = \frac{h_{01} - h_{02}}{h_{01} - h_{02s}} \tag{441}$$

where $\eta$ is the turbine thermal efficiency, and $h$ the enthalpy. Fig. 5 shows the location of the thermodynamic states on a $T$-$s$ diagram.

Turbine shaft work is calculated by

$$\dot{W}_t = \dot{m}(h_{01} - h_{02}) \ . \tag{442}$$

Eqs. (427), (437) and (441) are used to solve for $p_1$, $p_2$, and $\rho_2$, and Eq. (442) is used to compute turbine power. To derive stagnation states, recognize that

$$h_0 = h + \frac{1}{2}u^2 \tag{443}$$

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 [2])

$$p_0 = p\left(1 + \frac{\gamma - 1}{\gamma}\frac{\rho u^2}{2p}\right)^{\gamma/(\gamma-1)} \tag{444}$$

$$T_0 = T\left(\frac{p_0}{p}\right)^{(\gamma-1)/\gamma} \tag{445}$$

where $\gamma$ 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. (443). $p_{01}$ is calculated according to Eq. (444). $T_{01}$ is calculated according to Eq. (445).

The pressure $p_2$ and density $\rho_2$ are scalar variable unknowns. According to an EOS (equation of state) relationship, $h_2$ is evaluated

$$h_2 = h\left(p_2, \rho_2\right) \tag{446}$$

$u_2$ is obtained from turbine outlet pipe as a coupled variable. $h_{02}$ is calculated according to Eq. (443). $p_{02}$ is calculated according to Eq. (444). 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. 5)

$$\frac{\rho_{2s}}{\rho_1} = \left(\frac{p_2}{p_1}\right)^{1/\gamma} . \tag{447}$$

According to the EOS relationship, $h_{2s}$ is evaluated with $p_2$ and $\rho_{2s}$. Now $h_{02s}$ can be calculated according to Eq. (443). 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.

## 6.8 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:

$$V\frac{d\rho_{sd}}{dt} + \sum_{i=1}^{3}(\rho\boldsymbol{u})_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{448}$$

$$V\frac{d(\rho e)_{sd}}{dt} + \sum_{i=1}^{3}((\rho e)_i + P_i)\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{449}$$

where $\rho_{sd}$ 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:

$$P_i = P_{sd} + \Delta P_{acc} + s\Delta P_{form} + \Delta P_g \tag{450}$$

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 $\Delta H$ is the height difference between the elevation of the connecting pipe and the center elevation of the `SeparatorDryer` component.

## 6.9 DownComer

The BWR pressure vessel down comer is a 0-D model with a large volume that connects the feedwater pipe, the separator dryer discharge, the steam dome, and the down comer outlet. The volume is filled with vapor at the top and liquid at the bottom. During transients, the liquid level will increase or decrease (depending on the nature of the transient), which affects the mass flow rate through the reactor core; therefore, it is important to track the liquid level for transient analysis.

In the current model, it is assumed that there is no mass and energy exchange between the liquid and vapor phase in the down comer. Additionally, the vapor phase pressure is the same as that of the steam dome. Therefore, all the balance equations are solved for the liquid phase only.

The mass and energy conservation for the liquid in the down comer model are

$$\frac{d\rho_\ell V_\ell}{dt} + \sum_{i=1}^{3}(\rho\boldsymbol{u})_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{451}$$

$$\frac{d(\rho e)_\ell V_\ell}{dt} + \sum_{i=1}^{3}((\rho e)_i + P_i)\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_i A_i = 0 \tag{452}$$

where $\rho_\ell$ is the liquid density in the down comer component. $V_\ell$ is the liquid volume of down comer. $(\rho\boldsymbol{u})_i$ is the mass flux at the connecting nodes. $A_i$ is the flow area of the connecting pipe. $(\rho e)_\ell$ is the internal energy of the liquid in the down comer and $(\rho e)_i$ is the internal energy at the connecting nodes. $P_i$ is the pressure at the connecting nodes between the down comer of other components.

The following pressure balance equation is used to calculate the liquid level:

$$P_{dc} = P_g + \frac{1}{2}\rho_\ell g z \tag{453}$$

where $P_{dc}$ is the down comer reference pressure with its value taken at the center of the liquid volume. $P_g$ is the pressure in the vapor space of the down comer and $z$ is the liquid level relative to the bottom elevation of down comer.

For the case of liquid level above the pipe connection elevation, the incomplete form of the momentum equation is used to account for the various pressure losses in the down comer component

$$P_i = P_{dc} + \Delta P_{acc} + s\Delta P_{form} + \Delta P_g \ . \tag{454}$$

The pressure loss due to acceleration is $\Delta P_{acc} = \frac{1}{2}(\rho u^2)_{dc,\ell} - \frac{1}{2}(\rho u^2)_i$. The variable $s = 1$ if fluid is flowing into the down comer and $s = -1$ if fluid is flowing out of the down comer. 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 $\Delta P_g = \rho_\ell g \Delta H$, where $\Delta H$ is the height difference between the elevation of the center of the liquid volume and the elevation of the connecting pipes. In contrast, if the liquid level is below the pipe connection elevation, then

$$P_i = P_g \ . \tag{455}$$

## 6.10 Valves

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.

## 6.11 Compressible Valve Models

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 [91]. 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:

$$(\rho u)_1 A_1 \hat{n}_1 - (\rho u)_o A_2 \hat{n}_2 = 0 \tag{456}$$

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

$$(\rho u)_o A_2 \hat{n}_2 - \dot{m}_c = 0 \tag{457}$$

where $\dot{m}_c$ is the critical mass flow rate calculated by the equation for isentropic ideal gas flow [92]

$$\dot{m}_c = A_t (\rho u)_c = A_t \left(\gamma p_c \rho_c\right)^{1/2} \tag{458}$$

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 $\rho_c$ are determined by

$$\frac{p_c}{p_{i0}} = \left(\frac{2}{\gamma + 1}\right)^{\frac{\gamma}{\gamma-1}} \tag{459}$$

$$\frac{\rho_c}{\rho_{i0}} = \left(\frac{2}{\gamma + 1}\right)^{\frac{1}{\gamma-1}} \tag{460}$$

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 $C_v$ [92] 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

$$\dot{m}_{sub} = A_2 \left\{ 2 \left( \frac{\gamma}{\gamma - 1} \right) p_{i0} \rho_{i0} \left( \frac{p_2}{p_{i0}} \right)^{\frac{2}{\gamma}} \left[ 1 - \left( \frac{p_2}{p_{0i}} \right)^{\frac{\gamma-1}{\gamma}} \right] \right\}^{\frac{1}{2}} . \tag{461}$$

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. (457) is used for momentum; Otherwise subsonic flow momentum equation is used:

$$(\rho u)_o A_2 \hat{n}_2 - \dot{m}_{sub} = 0. \tag{462}$$

The $H_o$ unknown will correspond to the energy conservation:

$$(\rho u)_1 A_1 \hat{n}_1 H_1 - (\rho u)_o A_2 \hat{n}_2 H_o = 0 \ . \tag{463}$$

When the valve is fully closed, the following equations are used for $p_i$, $(\rho u)_o$, and $H_o$, respectively

$$p_i - p_1 = 0 \tag{464}$$

$$(\rho u)_o - (\rho u)_2 = 0 \tag{465}$$

$$H_o - H_2 = 0 \ . \tag{466}$$

$p_1$, $(\rho u)_2$, 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.

## 6.12 Wet Well Model

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. 6 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

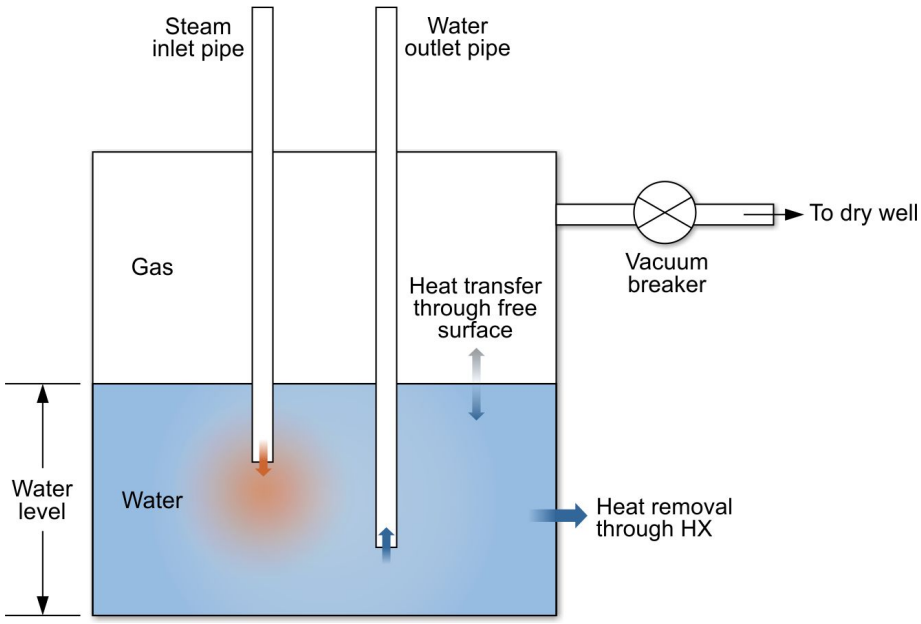


**Figure 6.** A simplified wet well model.

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

$$\frac{\mathrm{d}m_g}{\mathrm{d}t} = -\dot{m}_v \tag{467}$$

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

$$\frac{\mathrm{d}(me)_g}{\mathrm{d}t} = A_c\alpha\left(T_w - T_g\right) - \dot{m}_v H_v \tag{468}$$

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, $\alpha$ the effective 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

$$\frac{\mathrm{d}m_w}{\mathrm{d}t} = \dot{m}_{in} - \dot{m}_{out} \tag{469}$$

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

$$\dot{m}_{out} = (\rho u)_{out} A_{out} \tag{470}$$

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

$$\begin{aligned}\frac{\mathrm{d}(me)_w}{\mathrm{d}t} &= \dot{m}_{in}\left(H_{in} + (z_i - 0.5L_w)g\right) \\ &- \dot{m}_{out}\left(H_{out} + (z_o - 0.5L_w)g\right) \\ &- A_c\alpha(T_w - T_g) - \dot{q}\end{aligned} \tag{471}$$

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

$$H_{out} = \frac{\rho e_w + p_w(\rho_w, 0, \rho e_w)}{\rho_w} + \frac{1}{2}u_{out}^2 \ . \tag{472}$$

$u_{out}$ is the exit speed and is obtained from coupled water pipe end. The methods to calculate the average water density $\rho_w$ and specific volume energy $\rho e_w$ will be introduced shortly. For inflow condition, $H_{out}$ will be coupled from the pipe end. In Eq. (471), 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

$$p_w = p_g + \frac{1}{2} L_w \rho_w g \tag{473}$$

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

$$\rho E_t = \rho e = \rho \frac{me}{m} \ . \tag{474}$$

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 (467), (468), (469), (471), and (473). Another set of auxiliary variables is defined to close the system, which include gas density $\rho_g$ and water density $\rho_w$. Gas density is calculated according to

$$\rho_g = \frac{m_g}{A_c(L_t - L_w)} \tag{475}$$

where $L_t$ is the total effective height of the wet well. Similarly, the average water density is calculated according to

$$\rho_w = \frac{m_w}{A_c L_w} \ . \tag{476}$$

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 $p_{in}$

$$p_{in} + \frac{1}{2}(\rho u^2)_{in} = p_i + K_{in} \frac{1}{2}(\rho u^2)_{in} \tag{477}$$

where $K_{in}$ is the form loss coefficient and $p_i$ is the water pressure at the elevation of inlet steam pipe end

$$p_i = p_w + (0.5 L_w - z_i)\rho_w g \ . \tag{478}$$

The other two pipe's BCs are set in a similar manner.

## 6.13 Time Dependent Volume

Time Dependent Volume is a 0-D component that provides a time dependent pressure and temperature boundary condition to its connected 1-D component. It is purely a boundary condition type of component, and it does not add any entries to the global unknowns vector. When acquired by its connected component, it provides a pressure, a temperature, and vapor or gas volume fraction boundary condition as follows:

$$P(t) = P_0 + \frac{dP}{dt}(t - t_0) \tag{479}$$

$$T(t) = T_0 + \frac{dT}{dt}(t - t_0) \tag{480}$$

$$\alpha(t) = \alpha_0(t) + \frac{d\alpha}{dt}(\alpha - \alpha_0) \ . \tag{481}$$

This component can trivially provide constant pressure and temperature (and vapor or gas volume fraction) boundary conditions. Obviously, more complicated functions can also be implemented.

## 6.14 Time Dependent Junction

The Time Dependent Junction (TDJ) component provides either a velocity or mass flow rate (not both) boundary condition. The fluids temperature and vapor or gas volume fraction are specified in TDJ in providing this boundary condition. It is purely a boundary condition type of component, adding no entries to the nonlinear equations set.

$$T(t) = T_0 + \frac{dT}{dt}(t - t_0) \tag{482}$$

$$\alpha(t) = \alpha_0(t) + \frac{d\alpha}{dt}(\alpha - \alpha_0) \tag{483}$$

The velocity or mass flow rate boundary condition can be set, respectively, as

$$v = v_0 + \frac{dv}{dt}(v - v_0) \tag{484}$$

$$\dot{m} = \dot{m}_0 + \frac{d\dot{m}}{dt}(\dot{m} - \dot{m}_0) \ . \tag{485}$$

## 6.15 SubChannel

A fully coupled subchannel channel 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

$$A_i \frac{\partial \rho_i}{\partial t} + \frac{\partial(\rho_i u_i A_i)}{\partial x} + \sum_{j \in K(i)} w_{i,j} = 0 \tag{486}$$

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

$$A_i \frac{\partial \rho_i u_i}{\partial t} + \frac{\partial(\rho_i u_i u_i A_i)}{\partial x} + A_i \frac{\partial P_i}{\partial x} + A_i g \rho_i + \frac{1}{2}\left(\frac{f}{D_h} + K_i'\right) \rho_i u_i |u_i| A_i + \sum_{j \in K(i)} w_{i,j} u^* + \sum_{j \in K(i)} w_{i,j}^t (u_i - u_j) = 0 \tag{487}$$

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 $\epsilon_t$, by $w_{i,j}^t = \epsilon_t \rho_i \frac{s_k}{l_k}$. In the current RELAP-7 implementation, $w_{i,j}^t$ is calculated as $w_{i,j}^t = \beta s_k \bar{G}$, where $\beta$ 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

$$\frac{\partial w_{i,j}}{\partial t} + \frac{\partial w_{i,j} \bar{u}}{\partial x} - \frac{s_k}{l_k}(P_i - P_j) + \frac{1}{2}\frac{s_k}{l_k} K_G \frac{|w_{i,j}|}{\bar{\rho} s_k^2} w_{i,j} = 0 \tag{488}$$

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

$$A_i \frac{\partial \rho_i E_i}{\partial t} + \nabla \cdot (\rho_i u_i H_i A_i) + \rho_i g A_i u_i + \sum_{j \in K(i)} w_{i,j} H^* + \sum_{j \in K(i)} w_{i,j}^t (H_i - H_j) + \sum_{j \in K(i)} \frac{k}{l_i}(T_i - T_j) + \sum_{r \in M(i)} \phi_{i,r} h_w a_w A_i (T_i - T_{w,r}) = 0 \tag{489}$$

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.

## 6.16 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.

# 7 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.

## 7.1 Point Kinetics Equations

The point kinetics equations are the following:

$$\frac{dn(t)}{dt} = \frac{\rho(t) - \beta}{\Gamma} n(t) + \sum_{i=1}^{N_d} \lambda_i C_i(t) + S \tag{490}$$

$$\frac{dC_i(t)}{dt} = \frac{\beta f_i}{\Gamma} n(t) - \lambda_i C_i(t),\ i = 1, 2, \ldots, N_d \tag{491}$$

where $t$ is time (s), $n$ is the neutron density (neutrons/m$^3$), $\rho$ is the reactivity (only the time-dependence has been indicated, however, the reactivity is dependent on other variables). $\beta_i$ 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. $\Gamma$ is the prompt neutron generation time (s). $\lambda_i$ 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

$$\phi(t) = n(t)v \tag{492}$$

where $v$ is neutron velocity ($m/s$). The fission rate (fissions/s) $\psi(t)$ is calculated as

$$\psi(t) = V\Sigma_f\phi(t) \tag{493}$$

where $V$ is the volume (m$^3$) and $\Sigma_f$ is the macroscopic fission cross section (1/m). The reactor power is calculated from

$$P_f(t) = Q_f\psi(t) \tag{494}$$

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).

## 7.2 Fission Product Decay Model

The 1979, 1994, and 2005 Standards for decay power can be implemented by advancing the differential equations, which become

$$\frac{d\gamma_{\alpha j}(t)}{dt} = \frac{F_\gamma a_{\alpha j}}{\lambda_{\alpha j}} F_\alpha \psi(t) - \lambda_{\alpha j}\gamma_{\alpha j}(t) \quad j = 1, 2, \ldots, N_\alpha \tag{495}$$

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 $\lambda$ 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, $U^{235}$, $U^{238}$, $Pu^{239}$, $Pu^{241}$. $F_\gamma$ is an input factor to allow easy specification of a conservative calculation. It is usually 1.0 for best-estimate calculations. $F_\alpha$ is the fraction of fissions from isotope $\alpha$. Summation of $F_\alpha$ over $\alpha$ is 1.0. The uncorrected decay power is calculated as

$$P'_\gamma(t) = \sum_{\alpha=1}^{N_I} \sum_{j=1}^{N_\alpha} \lambda_{\alpha j}\gamma_{\alpha j}(t) \tag{496}$$

where $N_I = 3$ for the 1979 Standard and $N_I = 4$ for the 1994 and 2005 Standards. $\psi$ 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:

$$G(t) = 1.0 + (3.24E - 6 + 5.23E - 10t)T^{0.4}\psi_g \tag{497}$$

where $\psi_g$ 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

$$P_\gamma = G(t)P'_\gamma \ . \tag{498}$$

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.

## 7.3 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

$$\frac{d\gamma_U(t)}{dt} = F_U\psi(t) - \lambda_U\gamma_U(t) \tag{499}$$

$$\frac{d\gamma_N(t)}{dt} = \lambda_U\gamma_U(t) - \lambda_N\gamma_N(t) \ . \tag{500}$$

The actinide decay power is calculated as

$$P_\alpha(t) = \eta_U\lambda_U\gamma_U(t) + \eta_N\lambda_N\gamma_N(t) \ . \tag{501}$$

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 $\lambda$ and $\eta$ 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 $Np^{239}$.

## 7.4 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

$$\frac{d}{dt}\left[\frac{X\psi(t)}{v}\right] = \frac{[\rho(t)-\beta]X\psi(t)}{\Gamma v} + \sum_{i=1}^{N_d}\lambda_i XV\Sigma_f C_i(t) + XV\Sigma_f S \tag{502}$$

$$\frac{d}{dt}[X\Sigma_f C_i(t)] = \frac{\beta f_i X\psi(t)}{\Gamma v} - \lambda_i XV\Sigma_f C_i(t) \quad i = 1, 2, \ldots, N_d \tag{503}$$

$$\frac{d}{dt}[X\gamma_{\alpha j}(t)] = \frac{F_\gamma a_{\alpha j} F_\alpha X\psi(t)}{\lambda_{\alpha j}} - \lambda_{\alpha j} X\gamma_{\alpha j}(t) \quad j = 1, 2, \ldots, N_\alpha \tag{504}$$

$$\frac{d}{dt}[X\gamma_U(t)] = F_U X\psi(t) - \lambda_U X\gamma_U(t) \tag{505}$$

$$\frac{d}{dt}[X\gamma_N(t)] = \lambda_U X\gamma_U(t) - \lambda_N X\gamma_N(t) \tag{506}$$

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

$$P_T(t) = Q_f X\psi(t) + G(t)\sum_{\alpha=1}^{N_I}\sum_{j=1}^{N_\alpha}\lambda_{\alpha j} X\gamma_{\alpha j}(t) + \eta_U\lambda_U X\gamma_U(t) + \eta_N\lambda_N X\gamma_N(t) \tag{507}$$

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:

$$\rho(t) = \beta r(t) \tag{508}$$

$$X\psi(t) = \psi'(t) \tag{509}$$

$$\frac{XV\Sigma_f\Gamma v S}{\beta} = S' \tag{510}$$

$$XV\Sigma_f v C_i(t) = \frac{\beta f_i}{\Gamma\lambda_i} W_i(t) \quad i = 1, 2, \ldots, N_d \tag{511}$$

$$X\gamma_{\alpha j}(t) = \frac{F_\gamma a_{\alpha j} F_\alpha}{\lambda_{\alpha j}^2} Z_{\alpha j}(t) \quad j = 1, 2, \ldots, N_\alpha \tag{512}$$

$$X\gamma_U(t) = \frac{F_U}{\lambda_U} Z_U(t) \tag{513}$$

$$X\gamma_N(t) = Z_N(t) \tag{514}$$

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

$$\frac{d}{dt}\psi'(t) = \frac{\beta}{\Gamma}\left[(r(t)-1)\psi'(t) + \sum_{i=1}^{N_d} f_i W_i(t) + s'\right] \tag{515}$$

$$\frac{d}{dt}W_i(t) = \lambda_i \psi'(t) - \lambda_i W_i(t) \quad i = 1, 2, \ldots, N_d \tag{516}$$

$$\frac{d}{dt}Z_{\alpha j}t = \lambda_{\alpha j}\psi'(t) - \lambda_{\alpha j} Z_{\alpha j}(t) \quad i = 1, 2, \ldots, N_d \tag{517}$$

$$\frac{d}{dt}Z_U(t) = \lambda_U \psi'(t) - \lambda_U Z_U(t) \tag{518}$$

$$\frac{d}{dt}Z_N(t) = F_U Z_U(t) - \lambda_N Z_N(t) \tag{519}$$

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

$$P_T(t) = Q_f \psi'(t) + G(t)\sum_{\alpha=1}^{N_I}\sum_{j=1}^{N_\alpha}\frac{F_\gamma a_{\alpha j} F_\alpha Z_{\alpha j}(t)}{\lambda_{\alpha j}} + F_U \eta_U Z_U(t) + \eta_N \lambda_N Z_N(t) \tag{520}$$

where $N_I = 3$ for the 1979 standard and $N_I = 4$ for the 1994 and 2005 Standards.

## 7.5 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:

$$r(t) = \sum_{i=1}^{n_s} r_{si}(t) + \sum_{i=1}^{n_\rho}[W_{\rho i} R_\rho(\rho_i(t)) + a_{Mi}\Delta T_{Mi}(t)] + \sum_{i=1}^{n_F}[W_{Fi} R_F(T_{Fi}(t)) + a_{Fi}\Delta T_{Fi}(t)] \tag{521}$$

The quantities $r_{si}$ are obtained from input tables defining $n_s$ reactivity (scram) curves as a function of time. $R_\rho$ 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 $n_\rho$ 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.

# 8 Multi-Dimensional Capability and Interface with RAVEN

The RELAP-7 code consists mainly of 1-D and 0-D components, however, for certain applications, such as the reactivity insertion accident, the multi-dimensional effects are essential. Other ongoing MOOSE based codes provide the multi-dimensional capabilities the RELAP-7 code can leverage. These codes include the RattleSnake code for 3-D neutron kinetics, BISON for 3-D fuel performance, and BigHorn for 2-D/3-D computational fluid dynamics. These codes can be coupled with the RELAP-7 code to provide the necessary multi-dimensional analysis capability. It is noted that the coupling work is not yet complete.

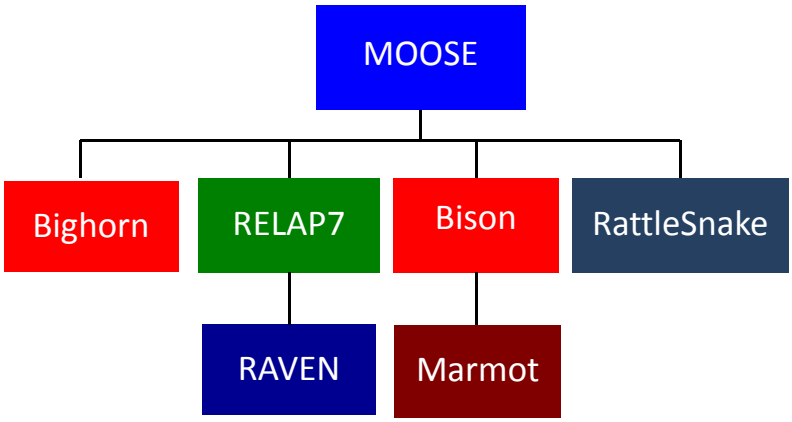


**Figure 7.** Multi-physics and multi-dimensional capability coupling for RELAP-7

## 8.1 RattleSnake

RattleSnake is a radiation $S_N$ transport code built using the MOOSE framework. It solves the multigroup $S_N$ transport equation in the formulation of Self-Adjoint Angular Flux (SAAF) with the continuous finite element method. RattleSnake can solve both the transient and eigenvalue problems with arbitrary order of scattering anisotropy. It can solve problems on 2D and 3D unstructured higher-order meshes in parallel. RattleSnake is designed to be easily inserted into a system of from pin-resolved fuel performance analysis to full-core safety analysis for tightly coupled multiphysics simulations. A multigroup cross section library format in XML (eXtendable Makeup Language) and its manipulator for interpolation and mixing operations are created within RattleSnake. Depletion capability is being developed for the code.

## 8.2 Bison/MARMOT

BISON [93] is a MOOSE based nuclear fuel performance code applicable to a variety of fuel forms including light water reactor fuel rods, TRISO particle fuel, and metallic rod and plate fuel. It solves the fully-coupled equations of thermomechanics and species diffusion, for either 1D spherical, 2D axisymmetric or 3D geometries. Fuel models are included to describe temperature and burnup dependent thermal properties, fission product swelling, densification, thermal and irradiation creep, fracture, and fission gas production and release. Plasticity, irradiation growth, and thermal and irradiation creep models are implemented for clad materials. Models are also available to simulate gap heat transfer, mechanical contact, and the evolution of the gap/plenum pressure with plenum volume, and fission gas addition. BISON has been coupled to the mesoscale fuel performance code MARMOT, demonstrating fully-coupled multiscale fuel performance capability. BISON is currently being validated against a wide variety of integral fuel rod experiments.

MARMOT [94] is a finite element-based phase field code for modeling irradiation-induced microstructure evolution. It predicts the effect of radiation damage on microstructure evolution, including void nucleation and growth, bubble growth, grain boundary migration, and gas diffusion and segregation. The phase field equations can be coupled with heat conduction and solid mechanics from ELK (Extended Library of Kernels which is a library for common kernels, boundary conditions and material base classes) to consider the effect of temperature and stress gradients on the evolution. In addition, MARMOT calculates the effect of the microstructure evolution with various bulk material properties, including thermal conductivity and porosity. Once the bulk properties have been calcu-

lated, they can be passed to BISON for a fuel performance simulation.

## 8.3 BigHorn

The overall research goal of Bighorn is to develop a simulation capability for reactor multiphysics around the next-generation implicit all-speed (nearly incompressible to highly compressible) flow method with an all-fluid (gas, liquid, liquid metal, two-phase mixture, etc.) capability to simulate the mass and energy transport of nuclear energy systems. Strong mathematical coupling of fluid flow and solid-state heat transfer, or conjugate heat transfer (CHT), results in a unique approach to energy conservation. Combined with JFNK, severe multi-scale temporal issues will be alleviated and phenomena that exhibit strong physical coupling yet defy strong mathematical coupling, such as radiation transport, will be fully-coupled in an implicit numerical scheme. The coupling techniques incorporated into Bighorn will result in a state-of-the-art multiphysics approach by allowing the strong mathematical coupling of turbulent coolant flow and solid-state heat conduction with the implicit numerical coupling of radiation transport (both neutron and photon). With an all-fluid capability, the Bighorn simulation tool will have application to Light Water Reactors (LWR), the Next Generation Nuclear Plant (NGNP) concepts, Sodium-cooled Fast Reactors (SFR), etc.

## 8.4 RAVEN

RAVEN(Reactor Analysis and Virtual Control Environment) is a multi-tasking application focused on RELAP-7 simulation control, reactor plant control logic, reactor system analysis, uncertainty quantification, and probability risk assessments (PRA) for postulated events. RAVEN is being developed to drive RELAP-7 for which the following functional capabilities are provided:

- Derive and actuate the control logic required to do the following:
  - Simulate the plant control system
  - Simulate the operator (procedure guided) actions
  - Perform Monte Carlo sampling of random distributed events
  - Perform event three based analysis.

- Provide a GUI to do the following:
  - Input a plant description to RELAP-7 (component, control variable, control parameters)
  - Concurrent monitoring of Control Parameters
  - Concurrent alteration of control parameters.
- Provide Post Processing data mining capability based on the following:
  - Dimensionality reduction
  - Cardinality reduction.
  - Uncertainty quantification and propagation

# References


[1] F. J. Moody, *Introduction to Unsteady Thermofluid Mechanics*. New Jersey: Wiley, 1990.

[2] H. I. H. Saravanamuttoo *et al.*, *Gas Turbine Theory, 5th edition*. Pearson Education Limited, 2001.

[3] 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 Sciencies (in French)*, vol. 43, pp. 265–276, Mar. 2004. `http://dx.doi.org/10.1016/j.ijthermalsci.2003.09.002`.

[4] 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`.

[5] 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`.

[6] 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`.

[7] 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`.

[8] 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`.

[9] 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.

[10] M. Heroux *et al.*, "An overview of Trilinos," Tech. Rep. SAND2003-2927, Sandia National Laboratories, 2003.

[11] A. Prosperetti and G. Tryggvason, *Computational methods for multiphase flow*. Cambridge University Press, 2007.

[12] 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.

[13] 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).

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] R. D. Falgout and U. M. Yang, "HYPRE: A Library of High Performance Preconditioners," in *International Conference on Computational Science*, pp. 632–641, 2002.

[19] R. C. Martineau, "The PCICE-FEM Scheme for Highly Compressible Axisymmetric Flows," *Computers & Fluids*, vol. 36, pp. 1259–1272, 2007.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] 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.

[25] INL, "RELAP5-3D Code Manual Volume IV: Models and Correlations," Tech. Rep. INEEL-EXT-98-00834, Idaho National Laboratory, June 2012.

[26] 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.

[27] 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`.

[28] R. Courant and K. Friedrichs, *Supersonic flow and shock waves*. New York: Interscience, 1948.

[29] F. Dobran, *Theory of Structured Multiphase Mixtures*. Berlin: Springer-Verlag, 1991.

[30] S. Whitaker, *The Method of Volume Averaging*. Dordrecht: Kluwer Academic, 1999.

[31] M. Ishii, *Thermo-Fluid Dynamic Theory of Two-Phase Flow*. France: Eyrolles, 1975.

[32] 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.

[33] D. A. Drew and S. L. Passman, *Theory of Multicomponent Fluids*. New York: Springer-Verlag, 1999.

[34] 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.

[35] D. Lhuillier, "The macroscopic modeling of multi-phase mixtures," in *Flow of Particles in Suspensions* (U. Schaflinger, ed.), New York: Springer-Verlag, 1996.

[36] 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.

[37] 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.

[38] I. M. Gelfand and G. E. Shilov, *Generalized Functions*. New York: Academic, 1964.

[39] 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`.

[40] 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.

[41] R. I. Nigmatulin, *Dynamics of Multiphase Media*. New York: Hemisphere, 1990.

[42] D. C. Wilcox, *Turbulence Modeling for CFD*. La Canada, California: DCW Industries, 1998.

[43] C. Truesdell, *Rational Thermodynamics*. New York: Springer-Verlag, 1984.

[44] 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.

[45] 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.

[46] 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.

[47] H. Nguyen, "One-dimensional models for transient two-phase separated flow," in *Third CSNI Specialist Meeting on Transient Two-Phase Flow*, (Pasadena, California), 1981.

[48] D. L. Hicks, "Hyperbolic models for two-phase (or two-material) flow," Tech. Rep. SAND81-0253, Sandia National Laboratories, U. S. A., 1981.

[49] 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.

[50] 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.

[51] 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.

[52] 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.

[53] S. Banerjee, "Analysis of separated flow models," Tech. Rep. NP-1442, Electric Power Research Institute, 1979. Research project 888-2.

[54] 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.

[55] 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.

[56] 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.

[57] 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.

[58] R. J. Leveque, *Finite Volume Methods for Hyperbolic Problems*. Cambridge, UK: Cambridge University Press, 2002.

[59] 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.

[60] 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.

[61] 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.

[62] J. Hadamard, "Sur les problemes aux derivees partielles et leur signification physique," *Princeton University Bulletin*, pp. 49–52, 1902.

[63] 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.

[64] R. A. Berry, "The seven-, five-, four-, and three-equation models of compressible, two-phase flow: A fully consistent, hierarchical set." In preparation, 2010.

[65] 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.

[66] 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.

[67] R. Abgrall and R. Saurel, "Discrete equations for physical and numerical compressible multiphase mixtures," *J. of Computational Physics*, vol. 186, pp. 361–396, 2003.

[68] A. Chinnayya, E. Daniel, and R. Saurel, "Modelling detonation waves in heterogeneous energetic materials," *J. of Computational Physics*, vol. 196, pp. 490–538, 2004.

[69] 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.

[70] 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.

[71] K. K. Kuo, *Principles of Combustion*. New Jersey: Wiley, second ed., 2005.

[72] 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.

[73] 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`.

[74] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*. New York: John Wiley & Sons, 2003.

[75] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Philadelphia: SIAM, 1998.

[76] R. A. Berry, "Notes on the pcice method: Simplifications, generalization, and compressibility properties," *Journal of Computational Physics*, vol. 215, pp. 6–11, 2006.

[77] R. A. Berry and R. C. Martineau, "Examination of the pcice method in the nearly incompressible, as well as strictly incompressible, limits," *Journal of Power and Energy Systems*, vol. 2, pp. 598–610, 2008.

[78] J. T. Oden and J. N. Reddy, *An Introduction to the Mathematical Theory of Finite Elements*. New York: Wiley & Sons, 1976.

[79] 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.

[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] 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, in preparation.

[85] M. O. Delchini, J. C. Ragusa, and R. A. Berry, "Extension of the entropy viscosity method to flows with friction forces and source terms," in *Physor2014*, (LaGrange Park, IL), American Nuclear Society, 2014. Kyoto, Japan.

[86] J.-L. Guermond and B. Popov, "Viscous regularization of the euler equations and entropy principles," 2014. Under review.

[87] B. I. Wohlmuth, "A mortar finite element method using dual spaces for the Lagrange multiplier," *SIAM Journal on Numerical Analysis*, vol. 38, no. 3, pp. 989–1012, 2000.

[88] US NRC, "TRAC-M/FORTRAN 90 (Version 3.0) Theory Manual," Tech. Rep. NUREG/CR-6724, US NRC, 2001.

[89] US NRC, "TRACE V5. 0 Theory Manual, Field Equations, Solution Methods, and Physical Models," tech. rep., US NRC. draft.

[90] A. Chaibakhsh and A. Ghaffari, "Steam turbine model," *Simulation Modelling Practice and Theory*, vol. 16, pp. 1145–1162, 2008.

[91] R. T. Layhey, Jr. and F. J. Moody, *The thermal-hydraulics of a boiling water nuclear reactor*. American Nuclear Society, 1993.

[92] S. Levy, *Two-phase flow in complex systems*. John Wiley & Sons, 1999.

[93] R. L. Williamson, J. D. Hales, S. R. Novascone, M. R. Tonks, D. R. Gaston, C. J. Permann, D. Andrs, and R. C. Martineau, "Multidimensional Multiphysics Simulation of Nuclear Fuel Behavior," *Journal of Nuclear Materials*, vol. 423, pp. 149–163, 2012.

[94] M. R. Tonks, D. Gaston, P. C. Millett, D. Andrs, and P. Talbot, "An Object-Oriented Finite Element Framework for Multiphysics Phase Field Simulations," *Computational Materials Science*, vol. 51, pp. 20–29, 2012.

Idaho National Laboratory