

AN ABSTRACT OF THE THESIS OF

Jaden Miller for the degree of Master of Science in Nuclear Engineering presented on September 22, 2022.

Title: CFD Verification, Validation, and Sensitivity Study of Internally Heated Concentric Tube Thermosyphon O-SERTTA

Abstract approved: _____

Wade R. Marcum

Irradiation experiments are a critical aspect of the nuclear fuels and materials qualification process. Determining the flow conditions of these experiments is necessary for understanding response of nuclear components during transient and normal operation. Quality flow data is lacking for cartridge-type irradiation experiments with annular flow natural circulation designs. Data were obtained for informing this type of design using computational fluid dynamics (CFD) and experimentation. The CFD simulation was verified and validated and the CFD code was used to perform a sensitivity study. The verification and validation of the CFD modeling proves the code's ability to quantify the thermal hydraulics of internally heated concentric tube thermosyphons, which has not been shown in literature. The CFD code was then used to determine modeling and geometrical sensitivities on the solution of transient and pseudo-steady state operation of the internally heated concentric tube thermosyphon.

©Copyright by Jaden Miller

September 22, 2022

All Rights Reserved

CFD Verification, Validation, and Sensitivity Study of Internally Heated Concentric
Tube Thermosyphon O-SERTTA

by
Jaden Miller

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented September 22, 2022
Commencement June 2023

Master of Science thesis of Jaden Miller presented on September 22, 2022

APPROVED:

Major Professor, representing Nuclear Engineering

Head of the School of Nuclear Science and Engineering

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Jaden Miller, Author

ACKNOWLEDGEMENTS

The author expresses sincere appreciation to the Marcum Research Group, particularly Dr. Wade Marcum, Dr. Trevor Howard, Dr. Guillaume Mignot, and Aaron Weiss.

Appreciation is also extended to the Graduate Council Representative, Dr. Joshua Gess.

TABLE OF CONTENTS

	Page
1 Introduction.....	1
1.1 Research Objective.....	3
1.2 Overview of Following Chapters	5
2 Survey of Literature	7
2.1 Cartridge Experiments.....	8
2.1.1 TREAT Cartridge Experiments	8
2.1.2 Other Cartridge Experiments	9
2.2 Natural Circulation with STAR-CCM+	10
2.3 Natural Circulation Experiments.....	11
2.3.1 Annular Natural Circulation Loops	13
2.4 Concentric Tube Thermosyphons	16
2.4.1 Externally Heated Concentric Tube Thermosyphons	17
2.4.2 Internally Heated Concentric Tube Thermosyphons	18
3 Experimental Facility.....	24
3.1 O-SERTTA.....	24
3.1.1 Effective Heat Transfer Coefficient.....	31
4 Analytical Modeling	33
5 Numerical Modeling	41
5.1 Fluid Flow	43
5.1.1 Equations of State	43
5.1.2 Newtonian Fluid Treatment	44
5.1.3 Boundary Conditions	44
5.1.4 Gravity	45

TABLE OF CONTENTS (Continued)

	Page
5.2 Turbulence.....	46
5.2.1 Reynolds Averaged Navier-Stokes (RANS) Equations.....	47
5.2.2 Eddy Viscosity Turbulence Models.....	48
5.2.3 The Realizable k - ε Two-Equation Model	49
5.2.4 Turbulent Heat Transfer and Conditions	53
5.3 Heat Transfer.....	54
5.3.1 Conduction.....	55
5.3.2 Convection	55
5.3.3 Conjugate Heat Transfer (CHT)	57
5.3.4 Radiation	58
5.4 Verification and Validation.....	58
5.4.1 Experimental Measurement Uncertainty	63
6 Analysis Methods.....	64
6.1 Initial Conditions and Startup	66
6.1.1 O-SERTTA	66
6.1.2 CFD Models.....	67
6.2 Steady State Operation	68
6.2.1 O-SERTTA Steady State Parameters.....	69
6.2.2 Heat Transfer Coefficient	70
6.3 Pulse Operation	71
6.4 Parametric Study	73
6.4.1 Parametric Modifications.....	74
6.5 Dimensionless Quantities	76

TABLE OF CONTENTS (Continued)

	Page
7 Results.....	77
7.1 Verification.....	77
7.1.1 Determining Center of Heat Sink	78
7.1.2 Mesh Independence	79
7.2 Validation.....	84
7.2.1 LUNA Fiber Optic Sensor Calibration	84
7.2.2 Pulse Operation.....	85
7.2.3 Pseudo Steady State	88
7.3 Sensitivity Study	95
7.3.1 Modeling Parameters	96
7.3.2 Geometrical.....	99
7.3.3 Flow Pipe Height Dependence.....	101
8 Conclusion	104
Bibliography	108
9 Nomenclature	119
A. Appendix.....	125
A.1 Numerical Flow Solution	125
9.1.1 Gradients	128
9.1.2 Segregated Flow Solver	131
9.1.3 Algebraic Multigrid Methodology	133
A.2 Wall Treatment	135
9.1.4 Wall Functions	137
9.1.5 Wall Treatment for Flow and Energy	138

TABLE OF CONTENTS (Continued)

	Page
9.1.6 Wall Treatment for Turbulence	140
9.1.7 Wall Distance.....	141
B. Appendix.....	143

LIST OF FIGURES

	Page
Figure 1.1. Common internally heated concentric tube thermosyphon irradiation device.	2
Figure 2.1. Simple natural circulation loops. (A) Rectangular loop (B) Vertical U-loop (C) Toroidal loop (D) Simple passive thermosyphon	12
Figure 2.2. Simple thermosyphon, concentric tube thermosyphon, and internally heated concentric tube thermosyphon (from left to right).....	17
Figure 3.1. O-SERTTA facility actual primary and secondary tank images.	26
Figure 3.2. Secondary cooling system (A) flow meter and (B) helical heat exchanger. ..	27
Figure 3.3. O-SERTTA primary images of internal components.	28
Figure 3.4. O-SERTTA internals computer aided design (CAD) model breakdown.	30
Figure 3.5. O-SERTTA simplified depiction and radial dimensions.....	30
Figure 3.6. O-SERTTA instrumentation identification schematic.	31
Figure 4.1. Single-phase simple natural circulation loop.	33
Figure 4.2. Simplified internally heated concentric tube thermosyphon depiction.	34
Figure 4.3. Analytical flow rate estimate as a function of inner flow area to outer flow area ratio.	39
Figure 4.4. Analytical flow rate estimate dependent on flow pipe length.	40
Figure 5.1. CHT interface analysis.	57
Figure 6.1. CFD geometrical representation of O-SERTTA along with cross sectional mesh of 2 mm base size mesh.	64
Figure 6.2. Axial dimensions of the CFD model as measured on O-SERTTA.	65
Figure 6.3. Local O-SERTTA temperature measurements during warm up.....	66
Figure 6.4. Local CFD temperature estimates over about 400 seconds of simulation.	67
Figure 6.5. Center of heat detection example of the (a) center of pulse vs (b) center of heat detection [95].	72
Figure 6.6. Power level of O-SERTTA heater rod during pulsing.	73
Figure 6.7. Comparison between O-SERTTA CFD model and simplified CFD model. .	74
Figure 6.8. Geometrical modifications for CFD sensitivity study.....	75
Figure 7.1. Center of heat removal determination plots.	79
Figure 7.2. Flow pipe CFD near surface temperatures with numerical uncertainty.	80

LIST OF FIGURES (Continued)

	Page
Figure 7.3. Rod CFD surface temperatures with numerical uncertainty.	80
Figure 7.4. CFD temperatures near the atmospheric boundary with numerical uncertainty.	81
Figure 7.5. CFD riser centerline temperatures with numerical uncertainty.....	82
Figure 7.6. CFD downcomer centerline temperatures with numerical uncertainty.....	83
Figure 7.7. CFD mass flow rate with numerical uncertainty and analytical comparison.	84
Figure 7.8. O-SERTTA LUNA fiber and TC-103 calibration plots.	85
Figure 7.9. O-SERTTA close look at thermocouple response during pulse.....	86
Figure 7.10. O-SERTTA fiber optic sensor heat map plots during power pulsing.	87
Figure 7.11. O-SERTTA center of heat pulse velocity estimate plot.	88
Figure 7.12. CFD and O-SERTTA downcomer temperature distributions with CFD heat map.....	89
Figure 7.13. Outer wall thermocouple validation error and uncertainty.....	90
Figure 7.14. LUNA fiber validation error and uncertainty for downcomer flow pipe surface.	90
Figure 7.15. CFD and O-SERTTA riser temperature distributions with CFD heat map..	91
Figure 7.16. LUNA fiber optic sensor validation error and uncertainty in the riser.....	92
Figure 7.17. CFD and O-SERTTA rod surface temperature distributions with CFD heat map.....	92
Figure 7.18. Rod surface thermocouple validation error and uncertainty.	93
Figure 7.19. Heat transfer coefficient depending on flow regime.	93
Figure 7.20. CFD and O-SERTTA velocity estimates with CFD velocity contour map..	94
Figure 7.21. Reynolds number of various O-SERTTA models and the simplified geometry.	96
Figure 7.22. Centerline temperatures of various O-SERTTA models and the simplified geometry.	98
Figure 7.23. Reynolds dependence on radius for simplified CFD model.....	99
Figure 7.24. Reynolds number based on flow pipe diameter for analytical model, experimental facilities, and CFD models.....	100

LIST OF FIGURES (Continued)

	Page
Figure 7.25. Flow pipe height impact on CFD flow rate.	101
Figure 7.26. Various simplified CFD geometry temperature profiles.	102
Figure 8.1. Internally heated concentric tube thermosyphon design improvement considerations.	105
Figure A.1. Cell discretization illustration of two neighboring cells.....	126
Figure A.2. SIMPLE algorithm overview.....	133
Figure A.3. Fixed cycling strategies.	134
Figure A.4. Bi-Conjugate Gradient Stabilized method overview.....	135
Figure A.5. Turbulent boundary layers.....	136
Figure A.6. Simple 2D schematic for wall treatment.	139

LIST OF TABLES

	Page
Table 2.1. Natural circulation loop reference table summary.....	21
Table 2.2. Concentric tube thermosyphon with water as working fluid reference summary.	23
Table 3.1. O-SERTTA tank specifications.	26
Table 3.2. Internal component specifications.	29
Table 5.1. Governing numerical equations variables and descriptions.	42
Table 5.2. Velocity gradient tensors.	46
Table 5.3. Important non-dimensional quantities relevant to CFD simulation.....	47
Table 5.4. Turbulence model coefficients and variables.	51
Table 5.5. Roache's Grid Convergence Index (GCI) five-step procedure outline.	60
Table 6.1. Steady state determination criteria.....	69
Table 6.2. Steady state experimental values for CFD model input.	70
Table 6.3. Heat transfer calculations to estimate the effective heat transfer coefficient. .	71
Table 6.4. Various model nomenclature and the variation details.....	75
Table 7.1. Mass flow rate Roache's Grid Convergence Index (GCI) value examples.	78
Table 7.2. Aspect ratios and respective mass flow rates for simplified CFD models. ...	102
Table A.1. Transport equation variables and terms.	125
Table A.2. Diffusive flux term details.	127
Table A.3. Hybrid Gauss-Least Squares Method term definitions.	129
Table A.4. Wall functions definitions and equations.....	137
Table A.5. Wall specifications for momentum and energy equations.....	140
Table A.6. Final conditions for turbulence models.....	141

1 INTRODUCTION

Irradiation experiments capable of testing advanced reactor fuels and materials are critical for nuclear technology development. Components in reactor cores must demonstrate adequate performance under design-based operating conditions before they can be deployed in a licensed reactor. Fuels are often qualified through integral fuel tests, including irradiation testing, to evaluate steady-state and transient performance [1]. Results from fuel qualification programs are valuable for informing fuel performance models and other system models which support reactor licensing.

Cartridge irradiation experiments are given consideration for testing in recent years because of advantages with infrastructure requirements, design constraints, and cost [2]. A cartridge experiment is a self-containing apparatus readily movable to and from the irradiation site. These types of experiments are designed based on mechanical, radiation, and thermal hydraulic analyses to evaluate heat transport and flow conditions. The thermal hydraulic quantities must be known so response of tested fuels and materials can be accurately evaluated during operation. Data for informing the design, analysis, and qualification of cartridge experiments is limited, especially for experiments with annular natural circulation. The lack of data makes it challenging to predict the temperature and flow characteristics within the experiments.

Cartridges have proved useful for transient testing and demonstrating advanced reactor concepts. This type of experiment was used to prepare for the Molten Salt Reactor Experiment at Oak Ridge National Laboratory (ORNL) [3]. Similar vehicles are regularly implemented at the Transient Reactor Test Facility (TREAT), including light water and sodium loops [4]. Self-contained annular sodium and lead cartridge experiments have also been used with the BOR-60 reactor [5]. Experiments of this kind provide new capabilities for nuclear fuels research. Continued irradiation testing with cartridge devices will further improve understanding of fuel performance under normal operation and design-basis accident conditions leading to improvements in nuclear technology efficiency and safety.

Natural circulation designs are of particular interest because of the simple, efficient operation. Current published literature does not demonstrate how some of the design parameters of loops with annular natural circulation affect flow rates. Cartridge experiments with annular natural circulation have a heated device internally, which can be electric or radiative depending on the application. An example of such a device is shown in Figure 1.1.

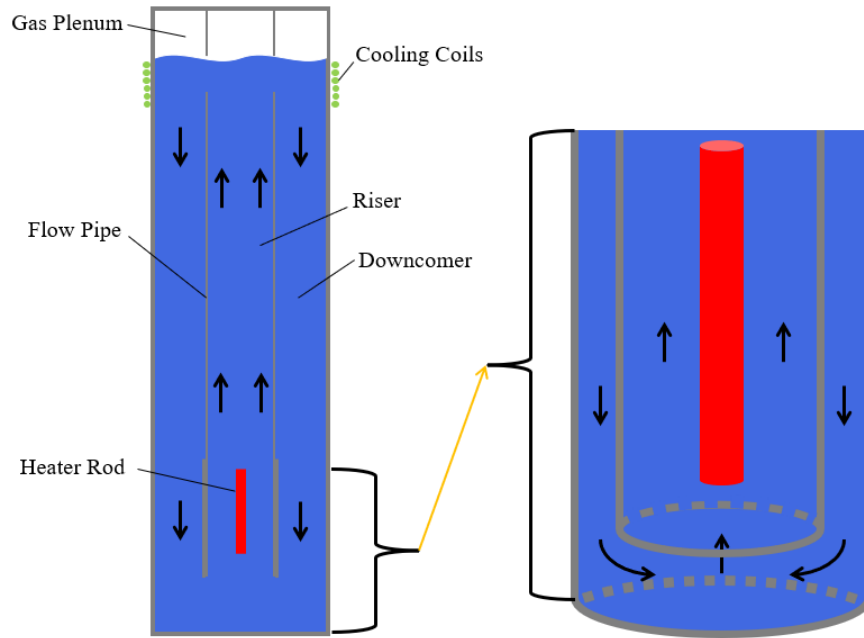


Figure 1.1. Common internally heated concentric tube thermosyphon irradiation device.

A flow pipe surrounds the heated specimen which is located within a cylindrical casing. Heat is removed from the surrounding casing to induce buoyancy driven flow. This design represents an internally heated concentric tube thermosyphon. The axial and radial dimensions of the flow pipe are parameters that impact natural circulation flow rates. Limited data for internally heated concentric tube thermosyphon designs results in large uncertainty in quantifying the temperature and flow distributions. The research presented here provides information relevant to future planned annular cartridge experiments with natural circulation and the off-normal evaluation of pumped experiments. For example, a loss of offsite power could result in loss of pumping accompanied by a gradual decrease in the coolant flow. In such a case where forced flow is lost, natural circulation flow rates

would provide input to determining the heating element surface temperatures during the accident scenario.

1.1 Research Objective

The objective of this research is to provide a validation and sensitivity study on internally heated concentric tube thermosyphon using STAR-CCM+. Validation studies of computational simulations of systems like Figure 1.1 are not available in the literature. The validation study is conducted to demonstrate the ability of the STAR-CCM+ code to evaluate thermal hydraulics of internally heated concentric tube thermosyphons. Natural circulation is buoyancy driven, where the density differences are caused by heating the fluid in the rising channel and then cooling the fluid in the downcomer channel. Natural circulation loops (NCLs) are well represented in the literature. However, the presence of a centered internal heat source within two concentric tubes introduces new parameters affecting flow rates that have not been well studied. The radial and axial dimensions of the flow pipe are presumed to affect flow rates in these devices, and it should be understood what aspect ratios optimize the flow rates. The specific goals of the thermal hydraulic analyses are to:

- Develop computational fluid dynamic (CFD) models to estimate the flow rates, temperature distributions, and pressure distributions of an internally heated concentric tube thermosyphon experimental facility called the Oregon State University Static Environment Rodlet Transient Test Apparatus (O-SERTTA).
- Provide experimental validation data from the O-SERTTA facility with thermocouples, pressure transducers, fiber optic sensors, and other instrumentation during normal and transient operation.
- Perform a study with the CFD models to address the sensitivity of turbulence modeling, initial conditions, flow pipe height, and flow pipe radius with respect to the resultant temperature and flow distributions.

In NCLs, density differences in the fluid should be facilitated to optimize flow. If the gravitational potential energy of the buoyant fluid impedes the natural circulation flow direction, flow may be slowed, stopped, or even reversed. This is of particular concern for systems driven by natural circulation alone, i.e., for systems with no pumps. Irradiation experiments should provide flow conditions representative of reactor technologies. To emulate reactor technologies and provide sufficient heat removal, flow rates should be as high as possible. It is also imperative to ensure the heat sink and heat source are positioned in a manner that results in high flow stability, so the flow is consistent and uni-directional.

The literature review on NCLs suggests flow rates increase as the height of the fluid being cooled relative to the height of the fluid being heated increases. This is because the buoyant fluid will have higher gravitational potential energy. The cooled fluid pushes the less dense, heated fluid up the riser channel. The higher the bulk fluid temperature in the riser section, the less force is required to move the fluid. Smaller riser flow areas have a positive impact on performance for some NCLs. However, reducing flow area can increase hydraulic losses. The dependencies of flow losses on cross flow areas are important to consider for designs.

In the internally heated concentric tube thermosyphon design, the radial dimensions are important to consider because of the heat transfer across the flow pipe. In NCLs, generally the fluid in the heated section does not lose heat to the fluid in the cooled section due to separation in the loop geometry. The literature review shows crossflow areas of NCLs have different impacts on flow rates for different loop geometries. Experiments with externally heated concentric tube thermosyphons have shown some of the optimal radial dimensions for the two concentric tube design [6], [7]. The NCL flow rates are optimized when the flow pipe radius is approximately half the enclosure radius. However, it has not been shown that this is true when the heat source is located internally. Internally heated concentric tube thermosyphons have different flow directions and crossflow geometries than externally heated designs. For the internally heated designs, it is hypothesized that the optimal radial dimensions are such that the buoyant resistance in the riser channel is minimized without

substantial restriction of the flow. For a fixed outer concentric tube radius, the riser channel flow area should be smaller than the downcomer area, but not small enough that the hydraulic losses outweigh the benefit of reduced body forces.

1.2 Overview of Following Chapters

Chapter two presents the results of a literature review on relevant research of irradiation experimentation and natural circulation relating to the proposed problem. The importance of cartridge experiments and the lack of current thermal hydraulic data on these experiments are discussed. The use of STAR-CCM+ with natural circulation studies is reviewed to show the efficacy of CFD model predictions of flow characteristics. The concluding discussion of this section are of experiments and models most representative of internally heated concentric tube thermosyphon designs.

Chapter four covers the analytical equations that govern the flow of natural circulation loops. The analytical model used to estimate the flow rates and temperature changes and the associated derivation are discussed. The assumptions used to obtain one-dimensional equations are highlighted.

Chapter five covers the numerical methods used to develop the model in STAR-CCM+. The mathematical models describing the physics derived from fundamental laws that express conservation principles are explained in relevance to the model of the cartridge experiment. The model boundary conditions, and other model input values are also discussed. The theory most critical to the modeling are discussed in chapter four and more specific details of the governing methods are included in Appendix A.

Chapter six presents the specifics on the initiation phases and the steady-state analysis of the cartridge experiment. Experimental measurements are detailed as they relate to the verification and validation discussed in chapter five. This includes how the experimental and computational data was collected and compared.

The results for the evaluation of the natural circulation in the cartridge experiment are shown and discussed in chapter seven. The full verification and validation results of the model are shown followed by the sensitivity study. Finally, chapter eight includes conclusions of the research and its applicability to current fuel qualification efforts is presented. Future work is suggested to improve upon the shown research.

Appendix A contains more detailed information about the numerical methodology. Appendix B has the python programming language functions that were used to quantify the parameters through the equations discussed throughout the text.

2 SURVEY OF LITERATURE

Irradiation experiments usually require the fuel to be directly exposed to the coolant and thermal boundary conditions representative of the reactor technology being researched. Thermal hydraulic boundary conditions have generally been controlled with external loops or cartridge experiments [2]. External loops extend from the irradiation site to a facility located outside the reactor vessel. The facility manages flow rates, coolant chemistry, and heat removal [8]. A cartridge experiment contains all the necessary equipment within an apparatus located at the irradiation site.

There are advantages and disadvantages to using cartridge experiments over external loop experiments [9]. External loops present operational hazards for workers who must allocate large amounts of floor space, handle potential leaks, and enforce other procedures that would otherwise not be of concern with cartridge experiments. There is a lack of infrastructure in current and planned high-flux reactors for external loops, which eliminates the option. Cartridges are also desired more than external loops for non-water coolants because of the solidification issues while the coolant travels to and from the irradiating site. Cartridge experiments can also operate under natural circulation, providing inherently safe heat removal and potentially eliminating requirements of in-core pumps.

The caveats to pursuing cartridge experiments are the design constraints. Containing all the necessary equipment within the irradiation vehicle is challenging and key components may have to be miniaturized to fit. Some components require customization, increasing development time and cost. Perhaps the most limiting factor of implementing an in-core irradiation vehicle is the inability to use organic compounds in high flux environments, namely o-ring seals, required by most pumps. Thermal hydraulic data on self-contained irradiation experiments with sufficient detail are limited which leads to large uncertainties in computational modeling. Although the flow characteristics of cartridge experiments are difficult to analyze, the advantages in safety, facility impact, and cost are significant. Experiments like these are regularly implemented and planned at the Transient Reactor

Test Facility (TREAT) [10]. Use of cartridge experiments for fuels testing is planned for the Versatile Test Reactor and High Flux Isotope Reactor [11], [12].

2.1 Cartridge Experiments

2.1.1 TREAT Cartridge Experiments

The Transient Reactor Test (TREAT) facility is an air-cooled high-power transient reactor designed in the late 1950s to test nuclear fuel performance [13]. TREAT successfully operated for 35 years before being placed in standby in 1994. The facility reclaimed its crucial role in nuclear fuels research in 2017 when it restarted operations. TREAT's most notable research advances were in integral-scale testing of large fuel specimens under transient accident conditions. New research planned for TREAT will examine smaller fuel specimens in modular hardware that can be inserted and removed readily from the core to examine post irradiation effects in the span of a couple weeks. This cartridge experiment is termed the Minimal Activation Retrievable Capsule Holder (MARCH) irradiation vehicle system [10]. The MARCH system introduces capabilities for cost-effective irradiation research for small-scale fuel samples and advanced instrumentation. These capabilities will lay the foundation for larger integral-scale test devices more representative of the environment in reactor power plants.

The MARCH-Static Environment Rodlet Transient Test Apparatus (MARCH-SERTTA) module enables testing of light water reactor (LWR) rodlets in the presence of pressurized water. MARCH-SERTTA has capability to adjust pre-transient subcooling and water phase conditions to study transient thermal hydraulic conditions while containing all instrumentation necessary for monitoring pressure, temperature, fuel expansion, and water-phase change. Unfortunately, geometric constraints limit specimen size, energy capacity, instrumentation density, and features that actively manipulate thermal hydraulic conditions [14]. A larger experiment vehicle, Super-SERTTA, is being developed to overcome these constraints and enhance other data collection.

LOCA testing at TREAT was proposed with Super-SERTTA, a closed natural circulation loop with an annular heater to provide a thermal boundary condition to a central rodlet [15]. The system begins at atmospheric conditions with a predetermined water level that rises as the temperature and pressure increase. Natural circulation begins when the water level rises above the flow tube which separates the riser and downcomer channels. A heat sink in the form of cooling coils outside of the vessel removes heat from the liquid in the downcomer. This vessel is connected to a blowdown tank by means of fast open valves to mimic blowdown conditions in a LOCA event. The primary tank of Super-SERTTA is an internally heated concentric tube thermosyphon. This design utilizes natural convection which reduces pump requirements while improving spatial constraints.

Thermal hydraulic studies using RELAP5-3D were done for the Super-SERTTA primary tank design [16]. The modeling showed that the system can achieve natural circulation in the annular flow path at 15 MPa with an inlet temperature of 287 °C and an outlet temperature of 293 °C. Natural circulation velocities were estimated as 1 m/s, which is about 20% of a commercial PWR's coolant velocity. This study found that the lower coolant velocities limited the rodlet power to maintain a sufficient margin to departure from nucleate boiling. The model predicted significant natural circulation capability for the geometry, but there were no geometrical considerations presented so there is uncertainty in whether the design can be improved. There is also no experimental data to validate the modeling which further increases the uncertainty.

2.1.2 Other Cartridge Experiments

Nuclear materials data has been collected with cartridge irradiation experiments using several liquids. Molten salt irradiation loops have played a crucial role in furthering the understanding of molten salt reactor technologies. Forced circulation loops in the Materials Testing Reactor and Low Intensity Test Reactor have provided corrosion data for molten salt mixtures operating with nuclear structural materials, such as Inconel and INOR-8 [17], [18]. Natural circulation molten salt loops were operated at the Oak Ridge Research Reactor to study material compatibilities, fuel salt stability, and fission product chemistry [19].

Cartridge experiments play a critical role in the nuclear fuels and materials infrastructure. Many different designs are considered but nearly all of them require annular coolant flow around the test specimen to emulate flow through rod bundles. Designs utilize natural circulation for the primary method of flow if pumps are not practical or necessary. Natural circulation is also important to consider in designs where pump failure can occur. The designs of these devices should be chosen in a way that optimizes natural circulation which ultimately increases safety and the effectiveness of fuels and materials experiments.

2.2 Natural Circulation with STAR-CCM+

Research on natural circulation using the commercial CFD codes, like STAR-CCM+, has been done in the past. A validated model has been developed for calculating natural convection gas velocities, pressure changes, and temperature distributions in a rectangular cavity with vertically heated rod bundles [20]. Natural convection cooling was also modeled with STAR-CCM+ to provide evidence of effective long-term cooling of core decay heat in a pool-type sodium fast reactor during a protected loss-of-flow transient [21]. Protected loss-of-flow accidents occur when the core is no longer critical but forced flow is lost. A similar study modeled the natural convection cooling in a Westinghouse-SMR during a shutdown scenario [22]. Flow characteristics were accurately predicted for natural circulation in a 3x3 vertically heated rod bundle section under rolling conditions [23]. Rolling conditions are the most common condition in the ocean and have a large influence on the thermal fluid behavior in reactors on board a ship. STAR-CCM+ also demonstrated accurate predictions of natural convection flow in a heated lead-bismuth eutectic coolant pool [24].

It is evident that STAR-CCM+ has the computational capability to accurately predict the flow rates, temperature distributions, and pressure changes in thermal fluid systems operating with natural convection. Transient and steady state operation can be illustrated with CFD codes for simple and complex geometries across many temperatures and flow rates. CFD high-fidelity thermal-hydraulic analyses are useful in informing real world

designs of these systems. Although it has been demonstrated natural convective flows can be accurately assessed with STAR-CCM+, there are no studies in the literature with concentric tube thermosyphons assessed with STAR.

In most cases of natural circulation loops, there is no heat transfer between the riser and downcomer channels. This interdependency exists in internally heated concentric tube thermosyphons and the impact on flow rates based on the riser to downcomer flow area ratios is an important design consideration. For externally heated designs, literature suggests the area ratio should be 1. Internally heated designs have reversed flow direction and an annular riser which changes operation and research is lacking to inform the designs when the heat source is internal.

2.3 Natural Circulation Experiments

Many advantages are inherent with the utilization of natural circulation. Circulation pump requirements are reduced or eliminated which reduces capital, operating, and maintenance costs. The safety of systems operating with natural circulation is increased because NCLs operate on natural physics like gravity, which is not expected to fail. This is valuable during nuclear accidents because natural circulation can provide sufficient heat removal in transients [25]. NCLs have more simple designs than forced flow loops which makes fabrication and operation better [26]. However, natural circulation is a weak phenomenon and, in some cases, provides inadequate flow rates for intended applications.

The basic constituents of NCLs are a heat source, heat sink, and the connecting pipes through which fluid flows. The piping is connected to the heat source and sink in such a way that buoyancy forces caused by fluid density differences drive flow. Usually, the heat sink will be in a higher vertical position than the heat source which enhances NCL flow rates. These NCLs, also called thermosyphons, are used throughout many different fields where passive heat transfer is desired or required over forced heat transfer. Some simple designs of NCLs are shown in Figure 2.1.

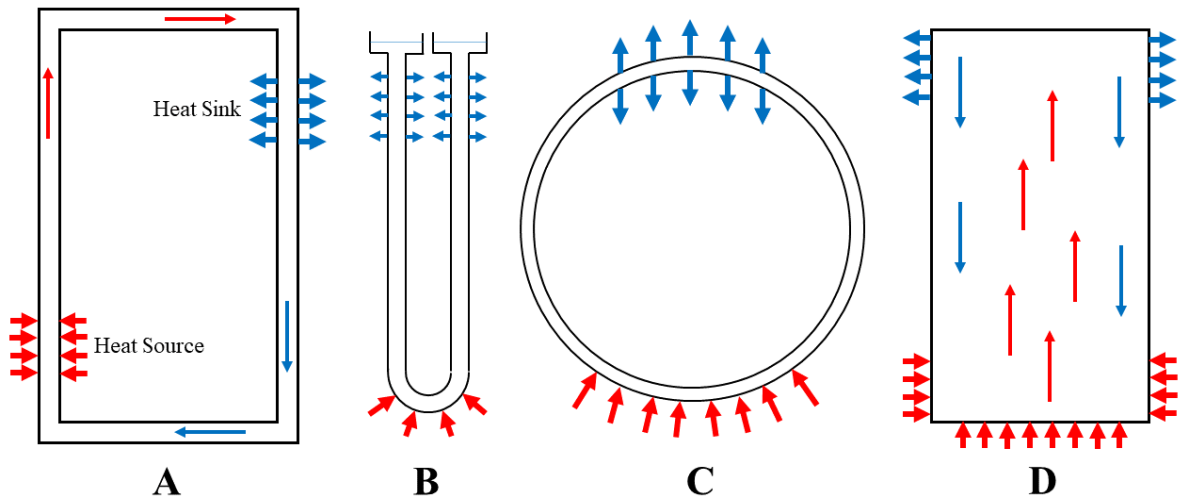


Figure 2.1. Simple natural circulation loops. (A) Rectangular loop (B) Vertical U-loop (C) Toroidal loop (D) Simple passive thermosyphon

Efforts have been undertaken to improve and understand the flow of many different NCLs. Analytical studies for steady-state loops show the performance is affected by the vertical distance between the heat source and heat sink, the fluid properties, and loop geometry [27]. The mass flow rates are generally expected to increase as the difference in bulk fluid temperatures between the riser and downcomer increases [28]. The temperature difference is the main influence of buoyancy driven flow. Several geometries can be considered for cartridge style irradiation experiments designed like NCLs.

NCL geometries can be classified based on the shape and many shapes have been studied which have led to a good understanding of the natural circulation process. Simple designs like the vertical U-loop [29], toroidal loop [30], rectangular loop [31], and many others have been investigated experimentally and theoretically for decades [32], [33], [34]. From this research it is known that for most NCL geometries, flow rates are enhanced by increasing the loop height and the density difference between the two vertical legs that are the downcomer and riser. Literature also suggests that increasing loop flow area and reducing hydraulic resistance also results in increased flow rates. The density differences

increase with higher power and heat removal [35]. Higher heat fluxes can potentially cause boiling which especially reduces fluid density in the riser.

Natural convection occurring around heated vertical cylinders has been studied for decades. Correlations for laminar and turbulent natural convective heat transfer around vertical cylinders have been tabulated and summarized [36]. Experiments were conducted to determine the local and average heat transfer coefficients for cylinders of various sizes and inclinations [37], [38]. The heat transfer and boundary layer regime transition are well understood for cylinders with no outer bounded wall, but less correlations have been made for natural convection occurring in a concentric annulus. The concentric annulus has a tube surrounding the cylinder forming an annular flow path with a heated interior surface.

Natural convective heat transfer in vertical internally heated annuli has been researched thoroughly because of the importance in a variety of fields. Natural convection in nuclear reactors is particularly important during loss of forced flow transients. The concentric annulus geometry is a convenient heat exchange configuration and represents a simple model for replicating heat and flow phenomena in rod bundles [39]. The geometry has been studied for many different boundary conditions and geometrical aspect ratios. The concentric annulus is the geometry of the riser for the internally heated concentric tube thermosyphon.

2.3.1 Annular Natural Circulation Loops

Relations of heat flux, physical properties of water, and heat removal have been developed for square shaped NCLs with concentric annuli as the riser. Research of test sections with natural circulation in two concentric tubes has characterized the flow behavior for several operating conditions and geometries [40], [41]. The heat transfer coefficient and Nusselt number is found to increase with increasing heat flux. The fluid flow rate was observed to increase exponentially with increasing heat flux. For high enough heat input, flow can be characterized as a boundary layer at the inlet of the riser, followed by subcooled boiling, and then boiling at the exit of the riser. Investigations of these loops has demonstrated the enhancement of flow rates for transition from single-phase to two-phase flow. Correlations

have been presented for determining where the point of boiling incipience (the instant two-phase flow begins in NCLs) is for a given heat flux and fluid flow rate [42]. This is important for evaluation of oxidation at the heated surface as this can pose problems, especially with nuclear materials. These correlations can be used for the internally heated concentric tube thermosyphon if riser inlet and outlet conditions are known.

The effect of pressure on critical heat flux in internally heated vertical annuli has been for many flow rates and system pressures [43]. It has been shown that the Doerffer correlation and the Bowring correlation predict CHF reasonably well for the concentric annuli [44]. These correlations have shown to overestimate CHF for low pressures and low flow rates, in which other correlations have been made [45]. CHF in low flow conditions was found to increase rapidly until pressures around a few MPa, after which CHF decreases slowly with increasing pressure. This research confirmed other findings that increasing flow rates are shown to increase and have more influence on CHF than pressure. The CHF is important when dealing with nuclear materials and these annular correlations can be used to inform the CHF of concentric tube NCLs.

Experimentation has been conducted in a vertical, electrically heated annulus with air as the working fluid [46]. This research investigated the effects on natural circulation with varying radius ratios. The facility was designed such that the cylindrical heater could be readily changed with heaters of different radii. The fluid goes up the riser and exits into the atmospheric air in a large surrounding tank. This research found that the Nusselt number increases as the annular gap increases. Experiments with polar fluids show the same results of improved heat transfer and velocities with increasing annular gap thickness [47]. These findings agree with previous research demonstrating performance of most NCLs increases with increasing flow areas for different geometries [26]. Nusselt number correlations have been made for large ranges of Rayleigh numbers for air in this geometry [48].

In annular geometries manufacturing tolerances, imperfections during construction, thermal stresses, etc. can cause the inner and outer surface to become eccentric. The effect of eccentricity on heat transfer in vertically heated annuli has been studied [49]. High

eccentricity results in large reductions in heat transfer. For the narrow gap region in eccentric geometries, the surface temperatures become higher than the wider gap regions. The highest heat transfer rates are achieved when the inner heated surface and outer surface are concentric [50], [51]. High eccentricities should be avoided because of the temperature spikes in the narrowed regions. However, small eccentricities have negligible effect on heat transfer characteristics and thus perfect concentricity is not necessary.

Validated numerical results have been presented for natural convective flows with annular geometry. CFD models have confirmed operating characteristics of natural convection flow in a partially heated vertical annulus with an aspect ratio of 352 [52]. The numerical results suggested increasing heat flux increases Nusselt number, which has been demonstrated experimentally. Two-dimensional numerical modeling was done to predict the location of incipient flow instability for low Reynolds number flows [53]. The instabilities were investigated throughout a parametric study on various operating conditions and geometrical parameters. Numerical modeling has investigated the effects of inclination angle on the flow and temperature distributions as well as the Nusselt number [54].

Transient natural convection of this geometry was modeled to investigate the dependence of heat transfer on governing dimensionless parameters [55]. The accumulated heat transfer was correlated to time, Rayleigh number, Prandtl number, and the aspect ratio. CFD was used to study the effects of using different inner cylinder material and outer cylinder geometric configurations when only the base of the inner cylinder is heated [56]. Numerical studies were also done for annuli closed at the top and opened at the bottom [57]. Validated numerical analyses of natural convection of high temperature gas between two concentric cylinders were conducted [58]. Results were obtained relating the heat transfer coefficient coupled with thermal radiation as functions of Rayleigh number, radius ratio, and temperatures of the heated inner surface and cooled outer surface.

The vertical concentric annular geometry has been studied for many fluid flows because of relevance to nuclear fuel bundles. Several correlations predicting fluid conditions have

been developed from experimentation and computation. The correlations can be utilized if boundary and initial conditions are known in the system. This data is valuable for informing the design and operational conditions of concentric tube thermosyphons with annular flow. However, concentric annular designs generally circulate the coolant through piping to the downcomer in a separate section of the loop. With concentric tube thermosyphons, the downcomer and riser are separated only by a flow pipe. Under these conditions, NCL operation is less understood and many correlations that have been developed for vertical annuli are not applicable due to invalid assumptions.

2.4 Concentric Tube Thermosyphons

The concentric tube thermosyphon has been proposed for use in nuclear fuels and materials research because of advantageous in space constraints, simplicity, and effectiveness. A typical nuclear reactor core consists of several hundred fuel assemblies that consist of fuel bundle elements. The cylindrical fuel pellets are located within an annular flow channel outside the fuel cladding. The riser channel of the concentric tube thermosyphon emulates the fuel bundle geometries found in reactor cores. Therefore, transient and steady operation of the thermosyphon design allows various phenomena to be observed in or out of the radioactive environment.

Vertically heated thermosyphons have been used for decades to provide passive thermal transport. These devices operate like other NCLs but have a cylindrical boundary in which they operate. The cylindrical tank or casing is heated at the bottom and cooled at the top. The simple vertical passive thermosyphon and its evolution is shown in Figure 2.2.

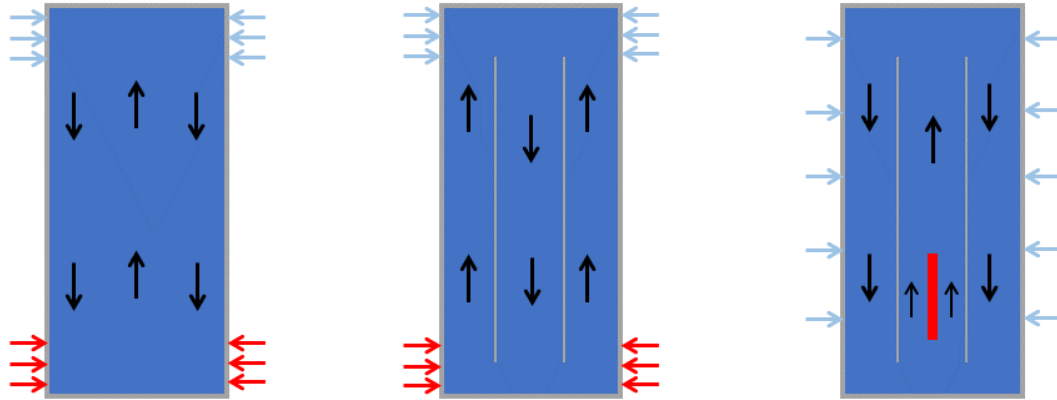


Figure 2.2. Simple thermosyphon, concentric tube thermosyphon, and internally heated concentric tube thermosyphon (from left to right).

Research advances led to the implementation of concentric tubes within the cylindrical fluid domain to facilitate flow and improve heat transfer. The externally heated concentric tube thermosyphons have annular riser channels on the outer surface of the concentric tube and circular downcomer channels inside the tube. The more novel designs use internally heated specimen which creates annular flow in the riser section which is not on the interior of the concentric tube. These devices are valuable in the nuclear industry for emulating flow in tube bundles passively.

2.4.1 Externally Heated Concentric Tube Thermosyphons

The vertical passive thermosyphon shown on the left in Figure 2.2 are well understood and have been used for many applications, like heat recovery and solar heating systems [59]. Investigations of this design have introduced an inner concentric tube within the cooled volume as shown in the middle of Figure 2.2. Flow is designed to go upward in the heated channel and downward in the central channel for concentric tube thermosyphons that are heated on the exterior of the fluid volume. This research found that the presence of the inner tube may counteract undesirable mixing of the hot and cool fluids [60]. Of notable importance in the findings of this research is the improvement in heat transfer characteristics by including an inner tube. Prior to this research, the single-tube open thermosyphon design was most heavily used in related research. The inclusion of an inner

tube in this experimental facility found that the overall heat transfer coefficient in the thermosyphon increased by a factor of 2 to 10.

The concentric tube thermosyphon is generally heated on the exterior bottom surface of the fluid test section which causes the heated fluid to rise in the outer channel. This fluid rises because the heat sink removes heat from the fluid at the top of the test section causing buoyancy buildup over the middle channel. Geometrical parameters have been investigated such as the diameters of the inner and outer tubes, the heated length, and the inner tube height [61]. The CHF maximum was found to occur when the inner tube diameter is about half the diameter of the outer tube [7]. Several other experimental and computational efforts have been undertaken to characterize the concentric tube thermosyphon CHF when the outer tube is heated [6]. However, these designs are all externally heated, which differ from cartridge experiments. The flow direction of internally heated devices is reversed. The interaction between the heat source and heat sink is also different. Externally heated designs remove heat at the upper exterior surface which results in fluid density increases above the riser, which introduces counteracting body forces. Instabilities can occur which impacts flow rates during buoyancy opposed flow [62].

2.4.2 Internally Heated Concentric Tube Thermosyphons

For current irradiation experiment designs, the loop is heated internally. Internally heated concentric tube thermosyphons studies are limited. Experimental facilities have been characterized for a heated inner tube with a cylindrical rod in the center channel [63], [64]. This data showed the CHF increases as the ratio of the inner tube height to the inner tube diameter increases. This is not contradictory to the conventional externally heated concentric tube thermosyphon. The literature does not investigate the annular dimension that exists when the cylindrical rod is in the center of the channel. The new irradiation experimental designs will also have a heated center portion rather than the heated inner tube. It is important to understand how the new geometrical parameters introduced in the concentric tube thermosyphon design affects flow rates and CHF for the designs.

Performance data was collected for a single-phase natural circulation loop with annular geometry [65]. This experimental research evaluated steady-state operation and transient behaviors of the thermosyphon test loop (TSTL) while varying the heater power and coolant flow rate of the external heat removal system. TSTL includes a central heated section surrounded by an annulus. Flow moves upward in the center of the device and flows down outside the annulus in a sealed vessel. The vessel is cooled with water driven by a centrifugal pump along the entire external test section length. The purpose for having the external heat removal along the entire length of the test section was to achieve representative conditions for cartridge experiments that can use the reactor coolant as the external heat removal. However, the reactor coolant temperatures can be too high or low for the intended operating temperature of the cartridge loop [2]. This is the case for the Super-SERTTA experiment which will be equipped with cooling coils near the top of the primary vessel.

Computational efforts have been undertaken to analyze the flow conditions of an electrically heated, pressurized water cartridge experiment with two-phase natural convection annular flow [66]. The facility is planned to be utilized in the Oak Ridge National Laboratory (ORNL) High Flux Isotope Reactor (HFIR). The loop will use the reactor's primary coolant as the heat sink along the entire length of the test section. Heating specimens will be in the bottom central part of the apparatus where water will boil and rise to the condenser section where condensate will form and continue to be cooled as it recirculates through the downcomer back to the heated section. The computational results from two different thermal hydraulic codes were compared and both predicted significant natural circulation (0.37-0.64 kg/s).

Annular natural circulation experiments and modeling are relevant to the discussed research. A summary of the references relevant to this research are shown in Table 2.1. The references are according to the bibliography at the end of this thesis. The letter 'x' indicates whether the labeled column applies to the reference where every column marked 'x' applies to the research in this document. The experiment column indicates if there was experimental data generated, the model indicates if a numerical or analytical model was

used, the 'V&V' column indicates if there was any level of verification or validation conducted during the research. The remaining columns are geometrical considerations of the natural circulation design. Based on the table, vacancies in research can clearly be seen. The reference that satisfies every column only does so with analytical modeling which is limited in detail when compared to numerical methods. Analytical models do not provide some details of the governing phenomena based on more assumptions made compared to numerical methods.

Table 2.1. Natural circulation loop reference table summary.

Reference	Experiment	Model	V&V	Annular Riser	Annular Downcomer	Concentric Tube
[15], [16]		X		X	X	X
[20]		X	X	X		
[21]		X	X			
[22]		X	X			
[23]		X	X	X		
[24]		X	X			
[25]	X					
[27]		X		X	X	
[30]	X	X	X			
[31]		X				
[32]						
[33]	X	X	X			
[34]		X				
[35]	X					
[36]	X	X	X			
[39]	X	X	X	X		
[40]	X	X	X	X		
[41]	X	X	X	X		
[43]	X	X	X	X		
[44]	X	X	X	X		
[45]	X			X		
[46]	X			X		
[47]		X	X	X		
[48]	X	X	X	X		
[49]	X			X		
[50], [51]		X		X		
[53]	X	X	X	X		
[54]		X		X		
[55]		X		X		
[56]		X		X		
[57]		X	X	X		
[58]	X	X	X			X
[59]	X	X	X			
[60]	X			X		X
[61]	X			X		X
[63]	X			X		X
[64]	X	X	X	X	X	X
[65]	X	X	X		X	X
[66]		X			X	X

The designs most applicable to this research are concentric tube thermosyphons using water as the working fluid. The published research papers on these devices are summarized in Table 2.2. The design is based on the two different ways a concentric tube can be heated: internally or externally. The flow rates are given for the smallest power due to the low power of the design discussed in this research. The flow is quantified in terms of mass flow rate (mfr) unless not able to be calculated based on the information in the referenced document. The primary purpose of the research for each design is discussed followed by more specific notes on the finding's other discussion. The area ratio is the crossflow area of the inner flow channel (A_i) divided by the crossflow area of the outer flow channel (A_o). The ratio is given for the actual dimensions of the facility or the ratio which yields the best performance in the case of variable flow pipe radii.

Table 2.2. Concentric tube thermosyphon with water as working fluid reference summary.

Design	Flow Rates	Primary Investigation	A_i/A_o	Notes
Externally heated [60]	Unknown	Nusselt number for different radial ratios, experimental and analytical	0.35	CHF max at this ratio.
Externally heated [61]	Unknown	CHF with tube diameters and heated length, experimental and analytical	0.33	Increased heated length increases CHF.
Externally heated [6]	Unknown	CHF with tube diameters and section length, experimental and analytical	0.33	Larger surrounding diameter increases CHF. Increased outer tube length decreases CHF
Internally heated [64]	0.016 kg/s at 4 kW	General characteristics at different powers, experimental and 1D analytical	0.692	Flow pipe is heated, inner rod is unheated. Flow velocities estimated by tracing salt with electrical conductivity probes.
Internally heated [65]	0.54 kg/s at 10 kW	Thorough characterization, experimental and numerical	3.6	Riser is 3 heater rods resulting in non-annular channel. Flow velocities calculated by change in pressure across orifice.
Internally heated [16]	1 m/s (mfr unknown)	Flow rates using RELAP, numerical	Unknown	RELAP results have not been validated and there are no discussions on geometrical parameters.

3 EXPERIMENTAL FACILITY

Experimental data is critical to using CFD modeling. As discussed in the literature review, computational simulations on internally heated concentric tube thermosyphons have not been experimentally validated. Without validation it is not possible to quantify the uncertainty of the predicted temperature and flow distributions in relation to reality. Experimental data was collected during this research to validate the CFD results. Validated CFD results become valuable to for determining experimental unknowns. This can be seen in Table 2.2 with externally heated designs and the unknown flow rates within the experiments.

3.1 O-SERTTA

The Halden boiling water reactor (HBWR) has been shut down and many in-situ diagnostic capabilities have been lost. The HBWR has been a significant source of experimental data to support the understanding of fuel-cladding response during LWR LOCA scenarios. In-situ diagnostics become increasingly important as the nuclear industry pursues LWR advanced fuels which extend burnup. Efforts are being made at Oregon State University (OSU) to develop in-situ measurement capabilities of distributed temperatures and local strain for test materials in PWR environments. Particularly, OSU has developed an out-of-pile LOCA simulating test loop which is accommodating of various instrumentation techniques.

Experiments in high temperature and pressure conditions play an integral role in developing and validating numerical tools. Many key parameters for material performance must be controllable and measurable during operation. Real-time in-situ measurements of importance include creep, crack propagation, swelling, two-phase flow, and many others. HBWR leveraged multiple diagnostic tools to demonstrate and improve in-situ measurements. Linear voltage differential transformers (LVDTs) are mentionable instruments. HBWR development efforts adapted LVDTs to monitor in-pile strain,

temperature, and elongation of test materials. A large focus was placed on diagnostic methods for fuel specimen temperature and strain.

Understanding the importance of in-situ temperature and strain measurements, the OSU LOCA simulating loop research is aimed at supporting the development of innovative instrumentation techniques focused on these measurements. The OSU loop also provides a benchmark design for LOCA emulating irradiation experiments with the ability to accommodate larger instruments than previous designs [16]. The LOCA-capability should impose boundary conditions upon the test specimen to reflect LWR conditions prior to blowdown.

The LOCA simulating loop, shown in Figure 3.1, is called the out-of-pile static environment rodlet transient test apparatus (O-SERTTA). The facility is a 1-to-1 scale of an experiment that could be used as an irradiation vehicle in TREAT or other test reactors. The facility consists of a primary tank and blowdown tank linked by fast acting valves to control emulated LOCA size. PWR conditions are attained in the primary tank prior to blowdown via natural circulation in an internally heated concentric tube thermosyphon design.

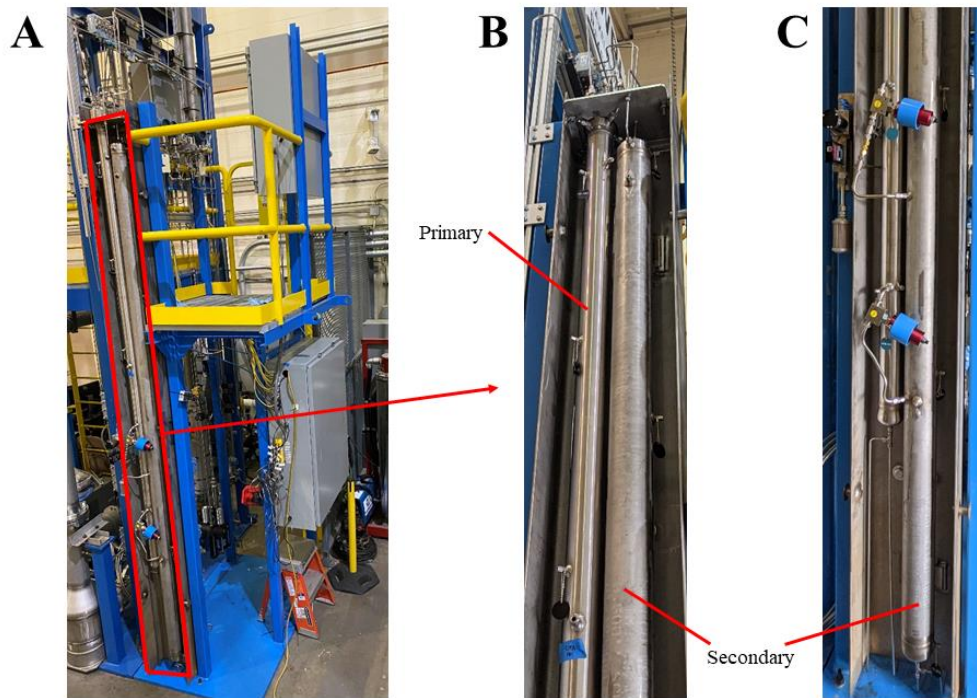


Figure 3.1. O-SERTTA facility actual primary and secondary tank images.

The O-SERTTA tanks were built by welding pipe caps to each end of a length of conventional piping. The primary tank has a removable head at the top that is sealed with a metal o-ring during operation. The specifications of the primary and secondary tanks are shown in Table 3.1.

Table 3.1. O-SERTTA tank specifications.

Parameter	Primary Tank	Secondary Tank
Length [m] (in)	2.9 (114.4)	3.429 (135)
Material	INCONEL 625	INCONEL 625
Schedule Designation	160	80S
ANSI Nominal Pipe Size [m] (in)	0.0762 (3)	0.0889 (3.5)
Outer Diameter [m] (in)	0.0794 (3.125)	0.1016 (4)
Inner Diameter [m] (in)	0.0666 (2.624)	0.0854 (3.364)

O-SERTTA is comprised of main feedwater, pressurizing, primary coolant, cooling, and air systems to control operation. The main feedwater system is equipped with valves and an air-driven pump to fill the test loop and provide coolant in the case of lost inventory. The pressurizing system consists of a spray tank that is connected to the primary and blowdown tank spray lines controllable through regulating valves. High pressure nitrogen is connected to the top of the primary and spray tanks to control pressure throughout operation. Heat removal of the primary coolant is done by convection and radiation to the surrounding atmosphere and a secondary system. The secondary cooling system, shown in Figure 3.2 (B), is made of tubing with compressed air flow wrapped around the upper portion of the primary tank.

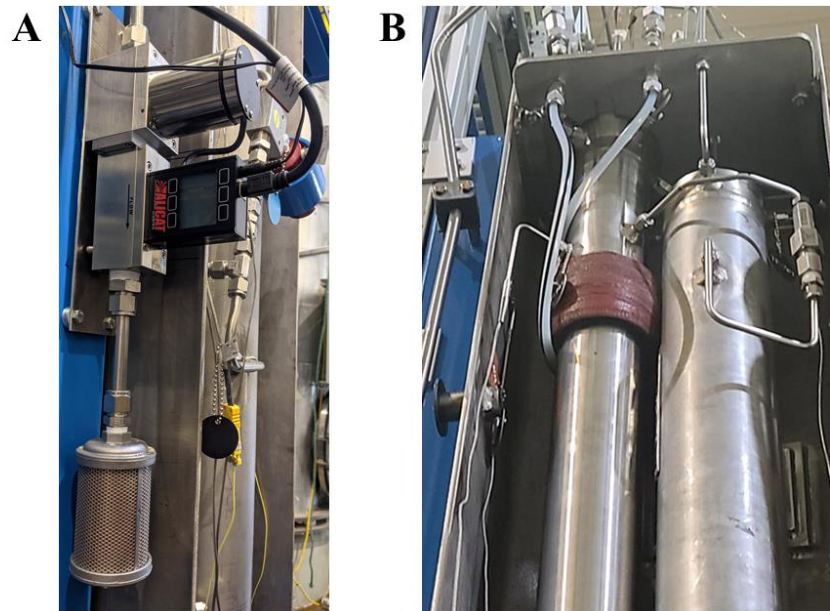


Figure 3.2. Secondary cooling system (A) flow meter and (B) helical heat exchanger.

Temperatures of the air at the inlet and exit of the secondary cooling system are monitored to estimate the heat removed during operation. The primary coolant system is composed of several annular components to create a natural circulation loop. A long concentric tube is attached to the primary tank head. Just below the attachment are three azimuthally symmetric slots that allows flow from the riser channel to go into the downcomer channel. Attached at the bottom of the long tube are a core heating component and a larger diameter

concentric tube that is attached to the longer tube. The core heating component is held within the larger diameter tube by a three-pronged component attached to the bottom of the assembly. The primary coolant internal assembly can be seen in Figure 3.3.

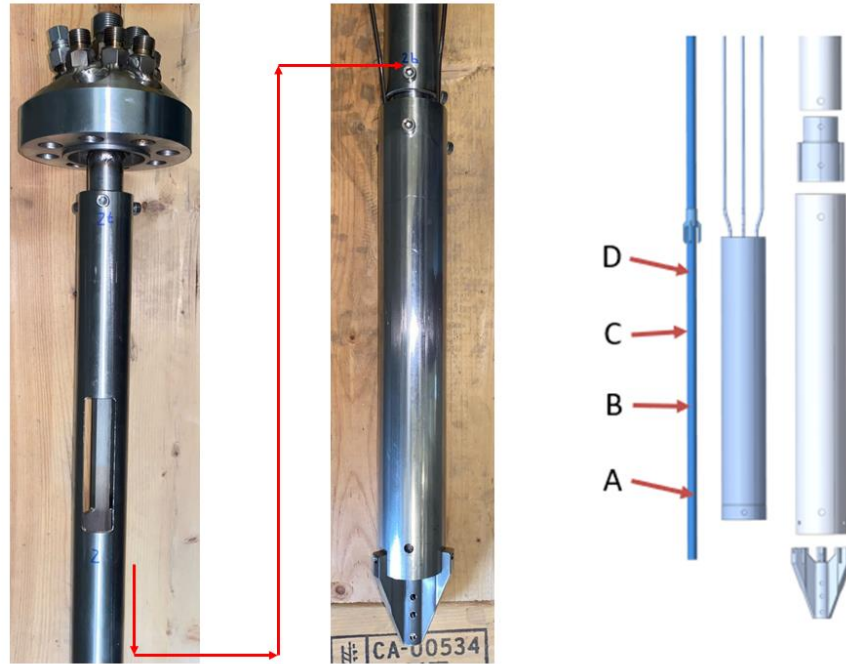


Figure 3.3. O-SERTTA primary images of internal components.

The three-pronged attachment at the bottom, called the spider, is hollow in the center to accommodate instrumentation such as LVDTs. The center annulus is formed when the central heater rod is placed within the internal assembly. The 2.86 m long heater rod extends from the top of the primary tank head and the base sits in the center of the top portion of the spider. Parameters of the internal components are shown in Table 3.2.

Table 3.2. Internal component specifications.

Parameter	Upper Tube	Lower Tube	Core Heater	Heater Rod
Length [m] (in)	2.108 (83)	0.335 (13.2)	0.2794 (11)	2.86 (112.6)
Material	Stainless Steel	Stainless Steel	Steel, Epoxy Ceramacast 645N	Monel K500, INCONEL 718, BN
Outer Diameter [m] (in)	0.0422 (1.66)	0.051 (2)	0.044 (1.75)	0.0095 (0.374)
Inner Diameter [m] (in)	0.035 (1.38)	0.047 (1.84)	0.02 (0.79)	N/A

The heater rod is produces heat within a 0.0254 m long section near the bottom portion of the rod. The heated section is signified by the thermocouple locations (A, B, C, D) shown in Figure 3.3. The core heater, shown to the right of the heater rod, is an annular heated component with an active heated length of 0.279 m. The core heater and heater rod can each reach powers of 5 kW. The core heater, however, is inactive during the experimental operations for this research.

The internal assembly of O-SERTTA is composed of several coupling components and weldments. The primary tank is shown in blue in Figure 3.4 with a detailed breakdown of the internal assembly. The heater rod is fed through the top of the primary tank cap as shown and is held airtight with a fitting. The assembly is held to the primary tank cap via the welded coupler shown in a teal color in the lower right portion of Figure 3.4. The flow pipe is slotted with rectangular sections for the turnaround region. The green coupler fastens the flow pipe to the core heater tube which helps hold the core heater in place. The spider region at the bottom of the core heater tube is connected via set screws and holds the heater rod centered in the riser channel.

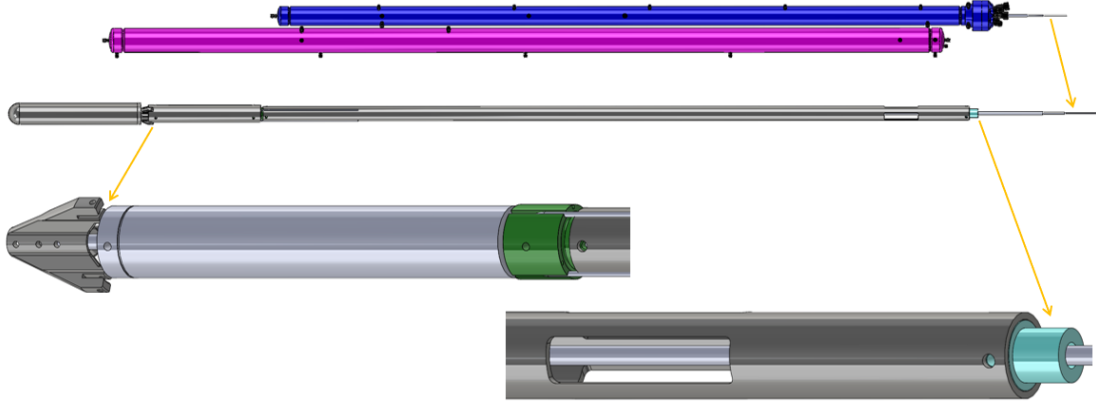


Figure 3.4. O-SERTTA internals computer aided design (CAD) model breakdown.

The riser and downcomer channels are annular flow paths with three different cross flow areas in the riser and two different cross flow areas in the downcomer. The radial dimensions of the annular flow paths are shown in Figure 3.5, along with a simplified breakdown of the main coolant system.

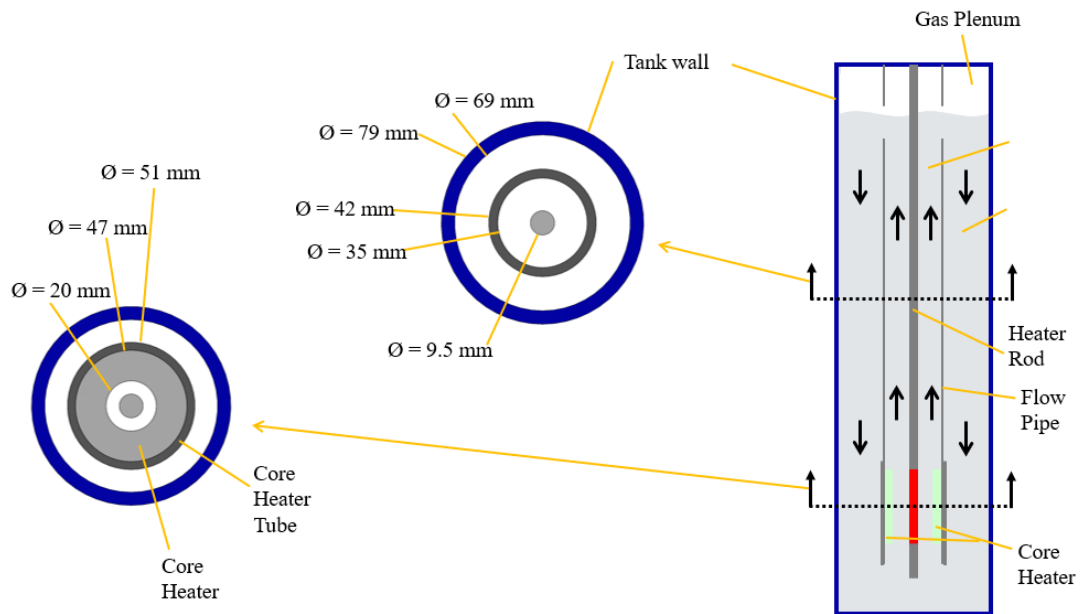


Figure 3.5. O-SERTTA simplified depiction and radial dimensions.

O-SERTTA is equipped with pressure and temperature instrumentation to evaluate the conditions during operation. Several thermocouples are located on the primary tank outer wall to measure fluid temperature near the surface. Four thermocouples are encased in the

heater rod to provide rod surface temperatures. A pressure transmitter gives pressure readings of the gas plenum, and a differential pressure transmitter measures the change in pressure from the gas plenum to the bottom of the fluid which verifies the liquid height in the tank. For the experiments conducted in this research, O-SERTTA was also equipped with fiber optic sensors developed by LUNA Innovations in the riser and downcomer channels. The temperature instrumentation locations are shown in Figure 3.6. The axial parameters are shown in the CFD section since the models are based on actual dimensions.

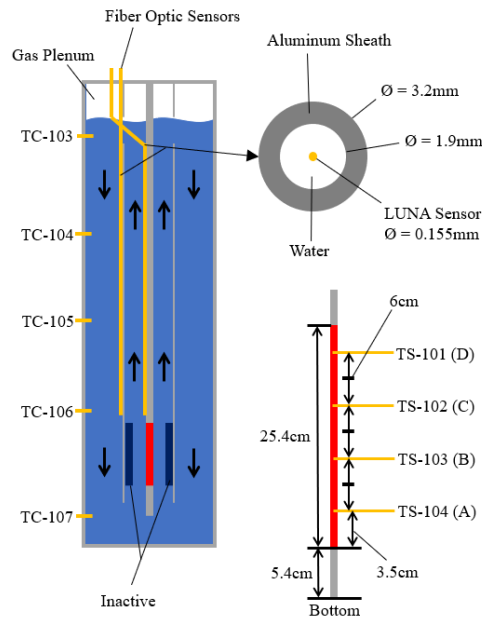


Figure 3.6. O-SERTTA instrumentation identification schematic.

3.1.1 Effective Heat Transfer Coefficient

Radiative heat transfer is considered in the effective heat transfer coefficient applied to the outer surface boundary condition of the simulation. The derivation of the effective heat transfer coefficient of the external tank surface is shown later. The convection coefficient is $10 \text{ W/m}^2\text{-K}$ for stainless steel in still air with similar properties to the O-SERTTA tank (INCONEL) [67]. However, this has been experimentally demonstrated to strongly depend on geometry, temperature, and the surroundings [68]. Since data directly applicable to O-SERTTA is not available, the experimentally determined value of $10 \text{ W/m}^2\text{-K}$ is used. The system is subject to radiative heat loss as well. The combined heat transfer coefficient is

comprised of both the radiation and convective heat transfer coefficients. The radiative heat transfer coefficient is calculated by the following formulation.

$$h_{rad} = \varepsilon\sigma(T_{s2}^2 + T_{\infty}^2)(T_{s2} + T_{\infty}) \quad 3.1$$

The emissivity of the O-SERTTA vessel is known and the interior surface temperatures are known to estimate the exterior surface temperatures required in Equation 3.1. The heat transfer from the fluid to the atmosphere is given by 3.2.

$$\dot{Q} = \frac{kA}{L}(T_{s1} - T_{s2}) = h_{combined}A(T_{s2} - T_{\infty}) \quad 3.2$$

This relation is used in conjunction with the radiative heat transfer coefficient relation to solve for T_{s2} which is the only unknown. This heat transfer coefficient was used for experimentation even though cooling coils covered approximately 6% of the area. The cooling coils were estimated to have the same heat transfer coefficient as above based on an estimate of heat transferred via the following equation.

$$\dot{Q} = \dot{m}c_p\Delta T \quad 3.3$$

Equation 3.3 can be set equivalent to 3.2 with a factor accounting for the percentage of area assuming the heat transfer coefficient is constant along the surface. The results based on these calculations are shown later in Table 6.2. Although this is a rough estimation the convective heat transfer coefficients usually have at least a 10% uncertainty [69].

4 ANALYTICAL MODELING

The analytical model is based on classic theory for buoyancy-driven flows in a one-dimensional approach [26]. Natural circulation loops must obey the laws of conservation of mass, momentum, and energy. Some additional assumptions can be applied to the conservation equations to obtain exact analytical solutions. Consider a rectangular loop with a uniform area flow path as shown in Figure 4.1. The heat source supplies a uniform heat flux over length S_h . The cooler removes heat with a constant heat transfer coefficient over the length $S_{hl} - S_c$.

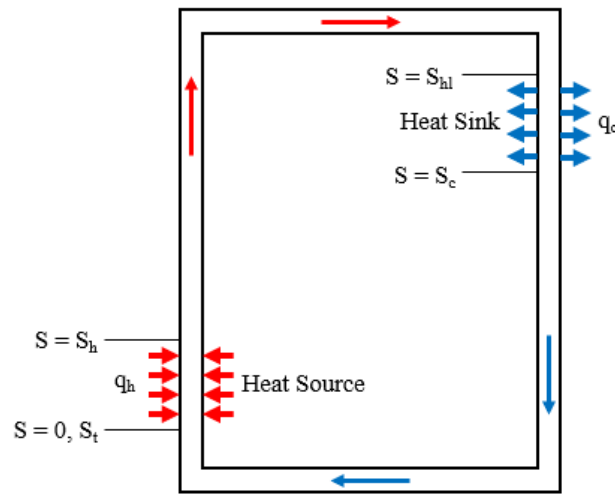


Figure 4.1. Single-phase simple natural circulation loop.

The average cross section temperature must be assumed to be equal to the mixed mean temperature throughout the loop. Consider the simplified O-SERTTA schematic shown in Figure 4.2. Assuming the flow pipe behaves as an insulative boundary (i.e., no heat transfer across flow pipe), we can represent the single-phase natural circulation loop with annular flow paths.

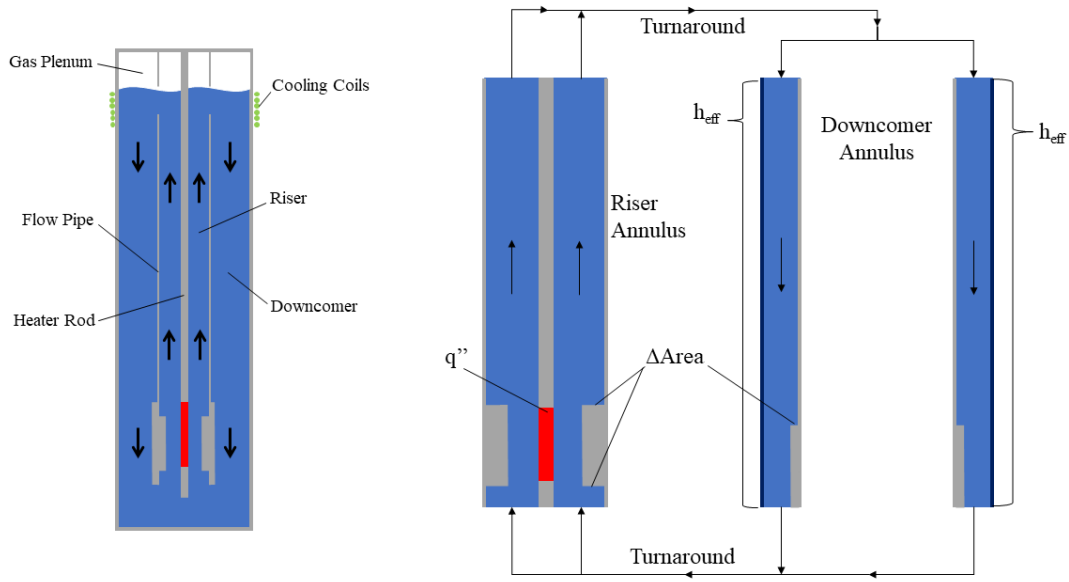


Figure 4.2. Simplified internally heated concentric tube thermosyphon depiction.

The cross-flow area change caused by the core heater is corrected for in the model resistance parameter. The fluid is light blue, the grey boundaries are assumed to be adiabatic, the red is the heat source, and the dark blue is the heat sink. The cooling coils have an effective heat transfer coefficient around the same value as heat transfer to the atmosphere. The calculations for the heat transfer coefficient are calculated by Equations 3.1 and 3.2. The coils are included as the same heat sink as the remainder based on the estimations of the heat removal according to Equation 3.3.

We can now apply traditional one-dimensional analytical equations to quantify the flow [26], [70]. Considering the fluid volume of the simplified system as the control volume, the conservation of mass is shown in Equation 4.1.

$$\frac{\partial}{\partial t} \int_{CV} \rho dV + \int_{CS} \rho \vec{V} \cdot d\vec{A} = 0 \quad 4.1$$

A basic assumption in this one-dimensional model is the average cross section temperature T , is equal to the mixed mean temperature. Systems of this type are generally held at some constant operating temperature and the difference in temperature through the system is

assumed to be small. The analytical model further assumes for single-phase incompressible fluids considered, the Boussinesq approximation is valid. The density is regarded as a constant density at the bulk fluid temperature in the governing equations except for the buoyancy force term. Under these conditions, Equation 4.1 reduces to Equation 4.2.

$$\frac{\partial(A\rho u)}{\partial s} = \frac{\partial(Au)}{\partial s} = 0, u = f(t) \quad 4.2$$

The variable u is the one-directional velocity in the s direction that flows through the loop. The mass and volumetric flow rates are thus uniform around the loop and are dependent only on time. Now consider conservation of momentum shown in Equation 4.3.

$$\frac{\partial}{\partial t} \int_{CV} \vec{V} \rho dV + \int_{CS} \vec{V} \rho \vec{V} \cdot \hat{n} dA = \sum F_{\text{contents of the CV}} \quad 4.3$$

As shown by Equation 4.2, integrating over the control surface for the second term on the left-hand side of Equation 4.3 yields a volumetric flow rate term that is constant everywhere in the loop, thus cancelling the term. The system considers pressure, gravitational, and frictional forces. Considering a finite length of the system (ds) and negligible cross sectional area changes through the flow path yields Equation 4.4. The area changes at the turnaround points and the core heater section are accounted for in the loss terms since area is constant elsewhere through the flow path. The momentum equation can be written as follows.

$$A\Delta s\rho \frac{\partial u}{\partial t} = \Delta(AP) - \tau_w \xi \Delta s - A\Delta s\rho g \hat{e}_z \cdot \hat{e}_s \quad 4.4$$

P is the pressure in the system, g is the acceleration of gravity, τ_w is the shear stress at the wall, and ξ is the wetted perimeter of the flow section. The unit vectors are vectors in the vertical (\hat{e}_z) and flow direction (\hat{e}_s). The shear stress term and the associated friction coefficient are given in Equations 4.5.

$$\tau_w = \frac{1}{2} f \rho_0 u^2, \quad f = \frac{a}{Re^b}, \quad Re = \frac{\rho_0 u d_H}{\mu} \quad 4.5$$

Re is the Reynolds number, μ is the dynamic viscosity of the fluid, and the variables a and b depend on the flow regime. Dividing by the area and Δs allows for further simplification. Knowing that the hydraulic diameter, d_H , is four times the area divided by the wetted perimeter yields Equation 4.6.

$$\rho_0 \frac{\partial u}{\partial t} = \frac{\partial p}{\partial s} - \rho g \hat{e}_z \cdot \hat{e}_s - 4 \frac{\tau_w}{d_H} \quad 4.6$$

Most theoretical models of thermosyphons have taken friction factor correlations (values of a and b) for fully developed forced flow. It has been shown that the forced flow correlations have an order of 30% uncertainty for many natural convective flow geometries [71], [72]. The driving flow mechanism in natural convection loops is different and the velocity distributions are different as a result. Lack of experimental correlations requires use of forced flow correlations since the variables strongly depend on specific loop geometry. The values for the friction factor in section i of the loop are given by the Fanning friction factor, which have been shown to provide good estimates of the frictional forces in annular loops [2].

$$f_i = \frac{16}{Re_i} \text{ for } Re \leq 1500, f_i = \frac{0.046}{Re_i^{0.2}} \text{ for } Re > 1500 \quad 4.7$$

The conservation of energy equation can be seen in 4.8.

$$\rho_0 c \left(\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial s} \right) - k \frac{\partial^2 T}{\partial s^2} = 4 \frac{q}{d_H} - 4 \frac{h}{d_H} (T - T_0) + \frac{1}{A} \int \Phi dA \quad 4.8$$

The variables c and k are the specific heat and thermal conductivity of the fluid, respectively. The heat flux input is q , T_0 is the reference (or wall) temperature, and Φ is the dissipation function. Previous work has shown that the dissipation effects can be neglected in most cases [73]. The term with the variable k accounts for axial conduction. The effects of axial conduction are very small and negligible for liquids and gases. The effects become important for liquid metals, or other fluids with small Prandtl numbers [70]. The heat transfer coefficient, h , is taken from forced flow correlations for the same reason as the friction factor.

An equation of state is added to the conservation of mass, momentum, and energy equations to solve for the four unknown variables u , T , ρ , and p . The coupled equations can only be solved numerically. However, in incompressible flows as shown earlier, the flow rate is uniform around the loop and the pressure terms can be eliminated by integrating around the loop. Integrating 4.6 with respect to s yields 4.9.

$$\rho_0 \gamma \frac{d\dot{V}}{dt} = -g \int \rho dz - 2\rho_0 \int f u^2 \frac{ds}{d_H} \quad 4.9$$

The geometrical parameter, γ , is introduced here to account for area changes through the flow path. Although the flow rate is constant, velocity can vary in various sections of the loop.

$$\gamma = \int \frac{ds}{A(s)} \quad 4.10$$

Another term is introduced to simplify the last term on the right side of 4.9. The overall flow resistance parameter, R , is given by 4.11.

$$R = 4 \int \frac{f ds}{d_H A(s)^2} \quad 4.11$$

The integral includes frictional losses in flow channels and any form losses from bends, valves, and other flow inhibitors. Equation 4.9 can now be rewritten as 4.12.

$$\rho_0 \gamma \frac{d\dot{V}}{dt} = -g \int \rho dz - \frac{1}{2} \rho_0 R \dot{V}^2 \quad 4.12$$

Equations 4.8 and 4.12 form a set of coupled equations for volumetric flow rate, \dot{V} , and temperature, T . A relation for ρ assumes linearly dependence on temperature through a constant thermal expansion coefficient, β .

$$\rho = \rho_0 [1 - \beta(T - T_0)] \quad 4.13$$

Numerical solutions can implement a more accurate function of ρ and in some cases use a temperature dependent thermal expansion coefficient. For steady state motion in a natural circulation loop, 4.8 and 4.12 govern the flow without the inertia terms. Assuming negligible axial conduction and viscous dissipation. Using the relation in 4.13 yields the following equations.

$$\frac{1}{2} \rho_0 R \dot{V}^2 = \beta g \rho_0 \int T dz \quad 4.14$$

$$\rho_0 c u \frac{\partial T}{\partial s} = 4 \frac{q}{d_H} - 4 \frac{h}{d_H} (T - T_0) \quad 4.15$$

If one solves 4.15 for various components of the loop, $h = 0$ in the heat source section, $q = 0$ in the heat sink section, and $h = q = 0$ in the connecting sections. This equation can be approximated to yield linear temperature distributions which is the exact solution for

uniform heat flux input. Integrating 4.14, the following derivation for flow rate in the loop near the heat source is found.

$$\dot{V} = \left(\frac{2\beta g \Delta z \mathbf{P}}{\rho_0 c R} \right)^{1/3} \quad 4.16$$

Here \mathbf{P} is the total input power and Δz is the equivalent driving head, which is the vertical distance between the centers of the heat source and the heat sink. The center of the heat sink is not necessarily the axial center of the atmospheric boundary. The quantification of the heat source center is discussed in the results section (Figure 7.1). Since 4.16 is a first approximation for the heat source section of the loop, the equation is independent of the details of the heat transfer in the heat sink. The actual fluid temperature in the loop depends on the heat transfer coefficient of the heat sink. The model considers five contributions to the overall flow resistance: form losses due to the bend and flow restricted path at the bottom of the test section, form losses from the bend in the flow path at the top of the test section, and frictional losses in the heated riser section ($i = q$), unheated riser section ($i = u$), and downcomer section ($i = d$). With this considered, the flow resistance is represented by 4.17.

$$R = \frac{k_b}{A_b^2} + \frac{k_t}{A_d^2} + \frac{k_c}{A_b^2} + \frac{k_e}{A_d^2} + \sum_{i \in \{d, q, u\}} \frac{4f_i L_i}{d_{h,i} A_i^2} \quad 4.17$$

The k variables are loss coefficients, where the subscript b refers to the bottom turnaround section, t refers to the top turnaround section, c refers to the contraction in the flow path below the core heater, and e refers to the expansion after the heated region. A_i , L_i , f_i , and $d_{h,i}$ are the cross-sectional area, flow length, friction factor, and hydraulic diameter of region i , respectively. 4.16 and 4.17 must be solved iteratively due to the dependence of the friction factor on flow rate. The bend loss coefficient at the top of the test section was taken as the typical value when the centerline bend radius is much larger than the internal pipe diameter, $k_t = 0.18$. Quantifying the loss coefficient at the bottom of the test section is more difficult because of the unique geometry of the spider component, however, it is expected to have a similar effect as the top turnaround region and is thus taken as the same loss coefficient as a bend. Using the volumetric flow rate relation in the energy equation gives the bulk fluid temperature difference across the heat source.

$$\Delta T_H = \frac{P}{\rho_0 c \dot{V}} \quad 4.18$$

The expressions for \dot{V} and ΔT_H represent first approximations for rough estimates of the steady state behavior of NCLs [74]. The steady state analytical model is compared to the pseudo steady state results of the simulation and experimental results. Pseudo steady state is the terminology used here because of the unsteady nature of natural circulation, where the steady state behavior is highly dependent on initial conditions [70]. The analytical model tells information about the flow rate dependence on the flow pipe height and radial parameters. The analytical model predictions based on changing the ratio of the inner flow area to the outer flow area by changing the flow pipe radius with a constant thickness is given in Figure 4.3. Concentric tube thermosyphons detailed in the literature are included in the plot to view the variances seen based on geometries.

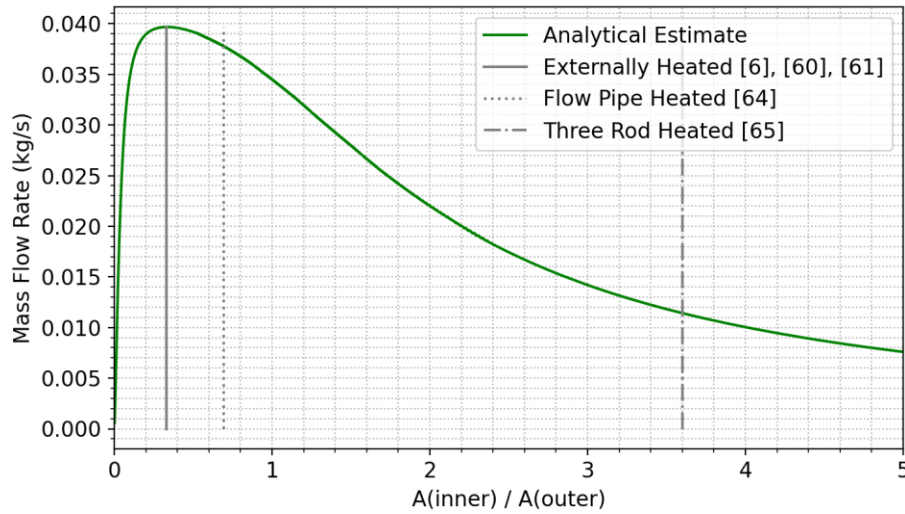


Figure 4.3. Analytical flow rate estimate as a function of inner flow area to outer flow area ratio.

The outer flow pipe diameter cannot be larger than the fluid diameter which can be seen by the gray dashed line. Additionally, the inner flow pipe diameter cannot be smaller than the rod diameter which is equal to 1 on the x-axis. The same impact on mass flow rate for flow pipe height can also be plotted.

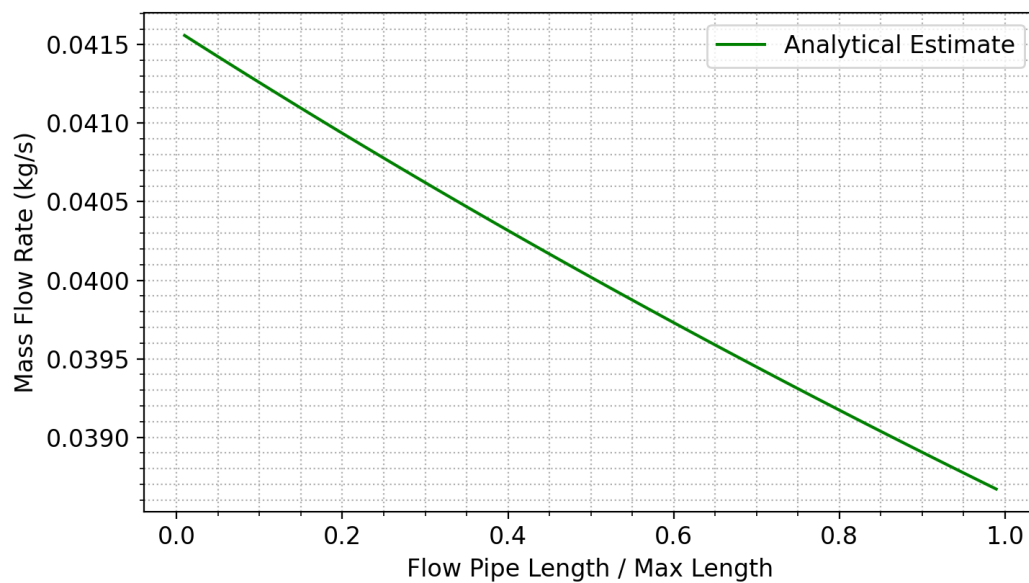


Figure 4.4. Analytical flow rate estimate dependent on flow pipe length.

5 NUMERICAL MODELING

CFD is a numerical methodology used for analyzing physical phenomena involving the heat transfer and flow of fluids. Useful applications are in the fields of aerospace, mechanical, nuclear, and other engineering applications. When observing materials macroscopically, the discrete structure of matter can be neglected, and material can be modeled as continua. CFD models consist of solid and/or fluid continua that are given physical attributes representative of their real-world properties. Using the finite volume approach to discretize the domain, the main governing conservation equations are solved to quantify the heat transfer and fluid flow behavior of the problem in question. Conservation of mass, momentum, and energy must be satisfied to establish a CFD solution. The governing equations are discretized over the domain to form a system of equations that are solved by computers.

The governing equations, the Navier-Stokes equations, assume the fluid behaves as a continuum instead of discrete particles. The original form of these equations is accurate for a wide range of fluid problems, however direct solution of these equations is far too computationally expensive outside of the simplest of cases. The Reynolds decomposition of the Navier-Stokes equations assumes the time-dependent turbulent velocity fluctuations can be separated from the mean flow velocity. This decomposition introduces Reynolds stresses which must be solved for with a turbulence model to solve the system of equations. This assumption is important to consider because there are many limitations and inaccuracies to the turbulence models used in present CFD software.

The balance of mass through a control volume is expressed in Equation 5.1. Equation 5.2 shows the time rate of change of linear momentum is equal to the resultant force acting on the continuum. Conservation of angular momentum, Equation 5.3, requires the stress tensor to be symmetric. The stress tensor is given by the sum of normal stresses and shear stresses. Equation 5.4 depicts the first law of thermodynamics applied to a control volume where energy is conserved. The integral forms of the continuity, momentum, and energy

equations are given by the second equations of 5.1, 5.2, and 5.4, respectively. The several variables in the governing equations are listed and described in Table 5.1.

$$\left[\begin{array}{l} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \\ \frac{\partial}{\partial t} \int_V \rho dV + \oint_A \rho \mathbf{v} \cdot d\mathbf{a} = \int_V S_u dV \end{array} \right] \quad 5.1$$

$$\left[\begin{array}{l} \frac{\partial(\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = \nabla \cdot \boldsymbol{\sigma} + \mathbf{f}_b = \nabla \cdot (p\mathbf{I}) + \nabla \cdot \mathbf{T} + \mathbf{f}_b \\ \frac{\partial}{\partial t} \int_V \rho \mathbf{v} dV + \oint_A \rho \mathbf{v} \otimes \mathbf{v} \cdot d\mathbf{a} = - \oint_A p\mathbf{I} \cdot d\mathbf{a} + \oint_A \mathbf{T} \cdot d\mathbf{a} + \int_V \mathbf{f}_b dV + \int_V \mathbf{s}_u dV \end{array} \right] \quad 5.2$$

$$[\boldsymbol{\sigma} = \boldsymbol{\sigma}^T = (-p\mathbf{I} + \mathbf{T})^T] \quad 5.3$$

$$\left[\begin{array}{l} \frac{\partial(\rho E)}{\partial t} + \nabla \cdot (\rho E \mathbf{v}) = \mathbf{f}_b \cdot \mathbf{v} + \nabla \cdot (\mathbf{v} \cdot \boldsymbol{\sigma}) - \nabla \cdot \mathbf{q} + S_E \\ \frac{\partial}{\partial t} \int_V \rho E dV + \oint_A \rho H \mathbf{v} \cdot d\mathbf{a} = - \oint_A \mathbf{q} \cdot d\mathbf{a} + \oint_A \mathbf{T} \cdot \mathbf{v} d\mathbf{a} + \int_V \mathbf{f}_b \cdot \mathbf{v} dV + \int_V S_u dV \end{array} \right] \quad 5.4$$

Table 5.1. Governing numerical equations variables and descriptions.

Variable	Quantity	Description
ρ	Density	Mass per unit volume
$\nabla \cdot$	Gradient	The divergence of dotted variable
\mathbf{v}	Velocity	The velocity of the continuum
\otimes	Outer Product	Procedure in linear algebra combining two vectors
\mathbf{f}_b	Body Forces	Resultant of the body forces (like gravity) per unit volume acting on continuum
$\boldsymbol{\sigma}$	Stress Tensor	Sum of normal stresses and shear stresses in fluid
$-p\mathbf{I}$	Normal Stress	p is pressure and \mathbf{I} is identity tensor
\mathbf{T}	Shear Stress	Viscous stress tensor to model stress attributed to strain rate
E	Energy	Total energy per unit mass
\mathbf{q}	Heat Flux	Heat per unit area
S_E	Source	Energy source per unit volume
S_u, \mathbf{s}_u	Source	Specified source terms
\mathbf{a}	Area	Area vector for a surface on cell surface area (A)
H	Enthalpy	Total enthalpy
t	Time	The temporal parameter

CFD uses discretization methods to convert the continuous system of equations into a set of discrete algebraic equations. Discretization methods divide the continuum into a finite number of subdomains known as cells or elements. The equations are applied across the

simulation domain and unknowns are stored at specific locations of the developed mesh, e.g., vertices, cell centroids, face centroids, or edges. The differential equations are turned into an integral form for the spatial discretization. For transient simulations, temporal discretization must also occur. The time derivative is discretized forming a coupled system of algebraic equations that are solved at each time-step. Important aspects of the CFD modeling are discussed in this section but the more detailed theory is included in Appendix A.

5.1 Fluid Flow

The partial differential equations (PDEs) of the mathematical models are often not closed sets, i.e., the number of unknown quantities exceeds the number of equations. Additional equations, called constitutive laws, are provided for closure depending on material being considered. To solve the momentum equations for the velocity field, closure equations exist that incorporate material properties like fluid viscosity and the normal stress coefficients. Equations of continua state are also required for closure of the governing equations.

5.1.1 Equations of State

The equations of state relate the material density and internal energy to pressure and temperature. The density of the solid materials in the presented models was modeled as a constant, thus not accounting for thermal deformation. Modeling of the fluid was single phase water using the models of IAPWS-IF97 (International Association for the Properties of Water and Steam, Industrial Formulation 1997) [75]. The IAPWS models calculate the density and other thermodynamic properties of water or steam by providing fundamental polynomial equations for the specific Gibbs free energy. Specific volume, internal energy, entropy, enthalpy, heat capacity, and speed of sound are derived by solving the equations. The cells in the domain that fall within the subcooled liquid or superheated vapor phases are valid with the IAPWS models incorporated into STAR-CCM+. In the case of superheated liquid and subcooled vapor, the enthalpies are calculated by Equations 5.5 and 5.6.

$$H_{sh}^L(T, P) = H_{r1}(T_{sat}, P) + c_{p,r1}(T_{sat}, P)(T - T_{sat}) \quad 5.5$$

$$H_{sc}^V(T, P) = H_{r2}(T_{sat}, P) + c_{p,r2}(T_{sat}, P)(T - T_{sat}) \quad 5.6$$

Where H_{sh}^L and H_{sc}^V are the enthalpies of superheated liquid and subcooled vapor, respectively. The subscripts $r1$ and $r2$ refer to the reference phases, where $r1$ is subcooled liquid and $r2$ is superheated steam. P is cell pressure and T is temperature of the cell being considered.

5.1.2 Newtonian Fluid Treatment

Constitutive equations are integrated for the viscous stress tensor (\mathbf{T}) if the fluid is considered Newtonian. The fluid modeled was water, which is treated as a Newtonian fluid. The explicit equation linearly relates the tensor to the velocity field through viscosity. The velocity field is expressed with a rate of deformation tensor, \mathbf{D} .

$$\mathbf{D} = \frac{1}{2}(\nabla \mathbf{v} + (\nabla \mathbf{v})^T) \quad 5.7$$

$$\mathbf{T} = 2\mu\mathbf{D} - \frac{2}{3}\mu(\nabla \cdot \mathbf{v})\mathbf{I} \quad 5.8$$

The dynamic viscosity (μ) is modeled as a constant for some fluid. The second term in 5.8 is zero based on the continuity equation for incompressible flows.

5.1.3 Boundary Conditions

The conditions at the solution domain boundaries need to be specified for closure. CFD can model a variety of conditions that cover practical situations of fluid flow. The boundaries may be confining, or flow may be allowed to enter or leave the domain. The boundary conditions used in the models shown here are wall or pressure outlet boundaries.

Wall boundaries represent impermeable surfaces confining a continuum. Viscosity effects are taken into consideration based on the fluid velocity and flow regime. Thermal specifications are available to determine the heat flux versus temperature relationship at the wall. The no-slip condition assumes the fluid sticks to the wall and has the same velocity as the wall at the interface. Thus, for a stationary wall, the fluid cell face at the wall is given a zero velocity. Convective heat transfer was considered for the outer surface walls of the model. The convection wall boundary condition considers a convective heat flux. The static fluid temperature at the wall is computed and is related to heat flux via an

effective heat transfer coefficient and an ambient temperature. The heat transfer elsewhere in the simulations was quantified by fundamental equations discussed in the following sections.

The pressure outlet boundary imposes working pressure on the system. A static pressure is set for pressure outlet boundaries into which the fluid can leave and re-enter the system. With normal outflow conditions, variable values are extrapolated from the interior of the solution domain to the boundary face values at the defined outlet surface. When backflow occurs through the pressure boundary, the boundary-normal specification is used to assume the incoming flow enters the simulation orthogonally to the pressure outlet surface. The pressure outlet boundary was chosen to represent the gas plenum of O-SERTTA. Little heat transfer from the fluid to the gas plenum is assumed and no flow occurs at the gas plenum interface which makes the pressure outlet a valid representation of the gas plenum. The boundary temperature and pressure were controlled to limit flow to near zero and limit heat transfer across the boundary.

5.1.4 Gravity

Gravitational forces are critical to the momentum balance equations when modeling natural convection. The body forces from gravity (\mathbf{f}_g) are added as a source term to the momentum equations.

$$\mathbf{f}_g = (\rho - \rho_{ref})\mathbf{g} \quad 5.9$$

The term \mathbf{g} is the classical gravity vector and the fluid reference density (ρ_{ref}) is dependent upon the continua at some reference temperature and pressure. Buoyancy forces caused by non-uniform density fields are considered significant when the Grashof (Gr) to Reynolds (Re) number ratio goes to or exceeds a value of one. This ratio, also known as the Richardson number (Ri), is shown in Equation 5.10.

$$Ri = \frac{Gr}{Re^2} = \frac{g\beta(T_{hot} - T_{ref})L}{\mathbf{v}^2} \quad 5.10$$

The variables β , L , and \mathbf{v} are the thermal expansion coefficient, the characteristic length, and the characteristic velocity, respectively. T_{hot} is the hot wall temperature and T_{ref} is the fluid reference temperature. When the temperature differences are small, the Boussinesq

model can be used to simulate buoyancy effects. However, temperature-dependent density models of the fluid continua can be more accurate with the sacrifice of additional computational resources, so the Boussinesq model is not necessary. When the gravity model is applied, the pressure of a cell is calculated as the piezometric pressure with respect to a reference static altitude pressure shown in Equation 5.11.

$$p_{piezo} = p_{static} - \rho_{ref} \mathbf{g} \cdot (\mathbf{x} - \mathbf{x}_0) \quad 5.11$$

The reference altitude (\mathbf{x}_0) is the point where piezometric and static pressure are equal.

5.2 Turbulence

Fluid flows of most engineering applications are characterized by random fluctuating flow quantities. The fluctuations are usually at small time scales and high frequencies making them computationally expensive to resolve in simulation by solving the governing equations of turbulent flows with Direct Numerical Simulation (DNS). Turbulence modeling is used to solve for averaged or filtered values to approximate the impact of flow fluctuations. The Reynolds-Averaged Navier-Stokes (RANS) turbulence models are of the most used and experimentally validated in CFD applications. The specific model should be chosen based on the application and engineering experience. Turbulence models are approximations of the physical phenomena of turbulence and the degree of accuracy depends on the nature of the flow being simulated. There are some important quantities in the context of turbulence modeling. Table 5.2 shows the strain rate tensor and vorticity tensor definitions.

Table 5.2. Velocity gradient tensors.

Variable	Description	Notes
S	$ \mathbf{S} = \sqrt{2\mathbf{S}:\mathbf{S}^T}$ $= \sqrt{2\mathbf{S}:\mathbf{S}}$	Modulus of the mean strain rate tensor.
\mathbf{S}	$\frac{1}{2}(\nabla \bar{\mathbf{v}} + \nabla \bar{\mathbf{v}}^T)$	The mean strain rate tensor, where $\bar{\mathbf{v}}$ is the mean velocity.
W	$ \mathbf{W} = \sqrt{2\mathbf{W}:\mathbf{W}}$	Modulus of the mean vorticity tensor.
\mathbf{W}	$\frac{1}{2}(\nabla \bar{\mathbf{v}} - \nabla \bar{\mathbf{v}}^T)$	The mean vorticity tensor, where $\bar{\mathbf{v}}$ is the mean velocity.

Several variables and non-dimensional quantities are also important to consider when selecting turbulence models. Popular quantities relevant to turbulence modeling are shown in Table 5.3.

Table 5.3. Important non-dimensional quantities relevant to CFD simulation.

Variable	Description	Notes
Re_d	$\frac{\sqrt{k}d}{\nu}$	Wall distance Reynolds number: k is turbulent kinetic energy, d is distance to the wall, ν is kinematic viscosity.
Re_t	$\frac{k^2}{\nu\varepsilon}$ or $\frac{k}{\nu\omega}$	Turbulent Reynolds number: ε is turbulent dissipation rate, ω is specific dissipation rate.
Re_ε	$\frac{(\nu\varepsilon)^{1/4}d}{\nu}$	Kolmogorov Reynolds number.
Re_v	$\frac{d^2S}{\nu}$	Strain-rate Reynolds number: S is modulus of mean strain rate tensor.
Ra	$\frac{\ g_i\ \beta\Delta TL_{ref}^3}{\nu\alpha}$	Rayleigh number: $\ g_i\ $ is norm of gravity vector, β is thermal expansion coefficient, ΔT is temperature difference, L_{ref} is characteristic length.
α	$\frac{\nu}{Pr}$	Thermal diffusivity
Pr	$\frac{c_p\mu}{k}$	Prandtl number: c_p is specific heat, μ is dynamic viscosity, k is thermal conductivity.

5.2.1 Reynolds Averaged Navier-Stokes (RANS) Equations

The Navier-Stokes equations (Equations 5.1, 5.2, and 5.4) describe the physics of fluid motion in CFD simulations. Analytical solutions are available for simple laminar flows. Most real-world flows are turbulent and three-dimensional, which makes analytical solutions unachievable in most scenarios. The Navier-Stokes discretized equations can be solved using Direct Numerical Simulation (DNS); however, this method would require extensive computational resources unfeasible for most CFD applications. The Reynolds-Averaged Navier-Stokes (RANS) equations are time-averaged equations used to simplify the fluid flow.

The simplification in the RANS equations is to decompose each solution variable (ϕ) in the instantaneous equations into two components, the mean ($\bar{\phi}$) and fluctuating (ϕ') components. This is known as Reynolds decomposition.

$$\phi = \bar{\phi} + \phi' \quad 5.12$$

ϕ represents velocity components, pressure, energy, or species concentration. Averaging of the variables is considered time-averaged for steady-state situations and ensemble averaged for transient simulations. The decomposed variables are inserted into the Navier-Stokes equations shown below.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{\mathbf{v}}) = 0 \quad 5.13$$

$$\frac{\partial}{\partial t}(\rho \bar{\mathbf{v}}) + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\nabla \cdot (\bar{p} \mathbf{I}) + \nabla \cdot (\bar{\mathbf{T}} + \mathbf{T}_{RANS}) + f_b \quad 5.14$$

$$\frac{\partial}{\partial t}(\rho \bar{E}) + \nabla \cdot (\rho \bar{E} \bar{\mathbf{v}}) = -\nabla \cdot \bar{p} \bar{\mathbf{v}} + \nabla \cdot (\bar{\mathbf{T}} + \mathbf{T}_{RANS}) \bar{\mathbf{v}} - \nabla \cdot \bar{\mathbf{q}} + f_b \cdot \bar{\mathbf{v}} \quad 5.15$$

The additional term in the momentum and energy transport equations is the stress tensor (\mathbf{T}_{RANS}).

$$\mathbf{T}_{RANS} = -\rho \begin{pmatrix} \overline{u'u'} & \overline{u'v'} & \overline{u'w'} \\ \overline{u'v'} & \overline{v'v'} & \overline{v'w'} \\ \overline{u'w'} & \overline{v'w'} & \overline{w'w'} \end{pmatrix} + \frac{2}{3} \rho k \mathbf{I} \quad 5.16$$

The components of the stress tensor must be solved to close the system, known as turbulence closure. Different turbulence models have been used for a wide range of problems. The turbulence model should be reasonably accurate and computationally inexpensive. Simplifications can be made to the Reynold stresses to incorporate two-equation models which have been widely used in CFD. Eddy viscosity models and Reynolds stress transport models are common approaches to turbulence modeling in the RANS equations.

5.2.2 Eddy Viscosity Turbulence Models

Models based on the analogy between molecular gradient-diffusion processes and turbulent motion are known as eddy viscosity models. Turbulent eddy viscosity (μ_t) allows for modeling the stress tensor as a function of mean flow quantities. The most common model is the Boussinesq hypothesis which simplifies turbulence models into two-equation terms.

$$\mathbf{T}_{RANS} = 2\mu_t \mathbf{S} - \frac{2}{3} (\mu_t \nabla \cdot \bar{\mathbf{v}}) \mathbf{I} \quad 5.17$$

Boussinesq assumes the momentum transfer from turbulent eddies can be modeled using a derived viscosity value. The assumption of the stress tensor being linearly proportional to

the mean strain rate does not consider turbulence anisotropy. Turbulence models can include a non-linear constitutive relationship to account for anisotropy with additional equations.

The realizable k - ε two-equation model was selected for turbulence closure in the shown CFD models. The selection was based on the wide, validated use in industrial applications and academia. The realizable k - ε model is a modified version of the standard k - ε model which applies the Boussinesq hypothesis to the Reynolds stress terms [76]. Two additional transport equations determine the turbulent kinetic energy (k) and turbulent dissipation (ε). The realizable modified version captures physics of flow separation, flow recirculation, rotational flows, and other turbulent phenomena more accurately than the standard model [77]. The realizable model was chosen over the standard model because of better simulation of buoyancy induced flow [78].

The Spalart-Allmaras turbulence model was not chosen because of the inaccurate predictions for flows involving buoyancy [79]. The standard k - ω model was not used because of the sensitivity to free stream and initial conditions [80]. The heating up of O-SERTTA to operating temperature is too computationally expensive to simulate so assumptions of the boundary conditions are made in the CFD model. The inability to simulate startup results in imperfect initial conditions which would likely be an issue with the standard k - ω model. The shear stress transport (SST) k - ω model addresses shortcomings of the standard model. Many studies have shown the SST k - ω and realizable k - ε models are both accurate for flows with convective heat transfer, but grid resolution requirements are greater for the SST k - ω model [81], [82]. To save computational time and considering there is no clear advantage in computational accuracy with the SST k - ω model for the application, the realizable k - ε model was chosen.

5.2.3 The Realizable k - ε Two-Equation Model

The realizable k - ε is a modified version of the standard k - ε model to better capture flow separation, flow recirculation, rotational flows, and other physics. The realizable version contains a new transport equation for the turbulent dissipation rate (ε). A variable damping

function (f_μ) is applied to a critical coefficient in the model (C_μ) to satisfy mathematical constraints on the normal stresses consistent with turbulent physics. The damped coefficient is consistent with experimental observations observed in boundary layers.

The two-layer approach allows the k - ε model to be applied in the viscous sub-layer and buffer layer. The computation is divided into two layers: one next to the wall and one away from the wall. The turbulent dissipation rate (ε) and the turbulent viscosity are specified as a function of wall distance. The turbulent dissipation in the near-wall layer is blended smoothly with the values computed in the other layer. Three common two-layer formulations have been used based on shear-driven and buoyancy driven flows. Wolfstein and Norris-Reynolds formulations are shear driven and the Xu formulation is for buoyancy driven flows. The turbulent eddy viscosity (μ_t) is calculated by the following.

$$\mu_t = \rho C_\mu f_\mu k T_e \quad 5.18$$

The large-eddy time scale (T_e) is the ratio of the turbulent kinetic energy (k) to ε . The transport equations for k and ε are given by the following.

$$\frac{\partial}{\partial t}(\rho k) + \nabla \cdot (\rho k \bar{\mathbf{v}}) = \nabla \cdot \left[\left(\mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right] + P_k - \rho(\varepsilon - \varepsilon_0) + S_k \quad 5.19$$

$$\frac{\partial}{\partial t}(\rho \varepsilon) + \nabla \cdot (\rho \varepsilon \bar{\mathbf{v}}) = \nabla \cdot \left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \nabla \varepsilon \right] + \frac{1}{T_e} C_{\varepsilon 1} P_\varepsilon - C_{\varepsilon 2} f_2 \rho \left(\frac{\varepsilon}{T_e} - \frac{\varepsilon_0}{T_0} \right) + S_\varepsilon \quad 5.20$$

The variables seen here are consistent with previous mentions. The dynamic viscosity (μ) is particular to the fluid being modeled. The optional turbulent kinetic energy and dissipation sources are modeled by S_k and S_ε , respectively. The ambient turbulence value (ε_0) in the source term counteracts turbulence balance decay [83]. When using this term, the specific time-scale (T_0) must follow Equation 5.21.

$$T_0 = \max \left(\frac{k_0}{\varepsilon_0}, C_t \sqrt{\frac{\nu}{\varepsilon_0}} \right) \quad 5.21$$

The formulation of the production terms is given in 5.22.

$$P_k = f_c G_k + G_b - Y_M, \quad P_\varepsilon = f_c S k + C_{\varepsilon 3} G_b \quad 5.22$$

Turbulent production (G_k) and buoyancy production (G_b) are given by the following.

$$G_k = \mu_t S^2 - \frac{2}{3} \rho k \nabla \cdot \bar{\mathbf{v}} - \frac{2}{3} \mu_t (\nabla \cdot \bar{\mathbf{v}})^2 \quad 5.23$$

$$G_b = \beta \frac{\mu_t}{Pr_t} (\nabla \bar{T} \cdot \mathbf{g}) \quad 5.24$$

The compressibility modification is taken from Sarkar [84].

$$Y_M = \frac{\rho C_M k \varepsilon}{c^2} \quad 5.25$$

To resolve the viscous sub-layer and buffer-layer, functions to mimic the decrease of turbulent mixing near the walls are required. The damping functions are as follows.

$$f_2 = \frac{k}{k + \sqrt{\nu \varepsilon}}, f_\mu = \frac{1}{C_\mu \left\{ 4 + \sqrt{6} \cos \left[\frac{1}{2} \cos^{-1} \left(\sqrt{6} \frac{s^{*3}}{\sqrt{s^*:s^*}^3} \right) \right] \right\} \frac{k}{\varepsilon} \sqrt{\mathbf{S}:\mathbf{S} + \mathbf{W}:\mathbf{W}}} \quad 5.26$$

The values for \mathbf{W} and \mathbf{S} are calculated according to Table 5.2. The additional constants and variables not yet discussed are given in Table 5.4.

Table 5.4. Turbulence model coefficients and variables.

Variable	Description	Notes
C_M	2	Sarkar coefficient
C_t	1	Model coefficient
$C_{\varepsilon 1}$	$\max \left(0.43, \frac{\eta}{5 + \eta} \right)$	Model coefficient, where $\eta = \frac{Sk}{\varepsilon}$
$C_{\varepsilon 2}$	1.9	Model Coefficient
C_μ	0.09	Model Coefficient
σ_ε	1.2	Model Coefficient
σ_k	1	Model Coefficient
$C_{\varepsilon 3}$	$\tanh \frac{ \mathbf{v}_b }{ \mathbf{u}_b }$	Model Coefficient
s^*	$\mathbf{S} - \frac{1}{3} \text{tr}(\mathbf{S}) \mathbf{I}$	Modified strain stress tensor
f_c	1	The curvature correction factor
c	343 m/s	Speed of sound
β	$-\frac{1}{\rho} \frac{\partial \rho}{\partial \bar{T}}$	Coefficient of thermal expansion

The two-layer models solve for k but give ε as a function of distance from the wall in the viscosity dominated regions (near-wall flow). The two-layer models prescribe the dissipation rate near the wall as follows.

$$\varepsilon = \frac{k^{3/2}}{l_\varepsilon} \quad 5.27$$

The length scale function (l_ε) is calculated by the Xu correction which is specific to natural convective flows [85] or the Wolfstein correction for shear driven flows [86]. The Xu correction is given by Equation 5.28, and the Wolfstein correction is given by Equation 5.29.

$$l_\varepsilon = \frac{8.8d}{1 + \frac{10}{y_v^*} + 5.15 * 10^{-2} y_v^*} \frac{1}{\sqrt{\frac{v^2}{k}}} \quad 5.28$$

$$l_\varepsilon = c_l d \left[1 - e^{-\frac{Re_d}{2c_l}} \right] \quad 5.29$$

The variable d is the distance to the wall and c_l is calculated by $0.42C_\mu^{1/4}$. The Xu correction was shown to improve the prediction of buoyancy-affected flows while reducing computational costs for natural convection along vertical plates and in cavities. However, results from this research demonstrated the Xu correction underpredicted velocity compared to experimental results and CFD results using the Wolfstein correction. Further discussion of the CFD results using the two different corrections is given in the results section. A wall-proximity indicator combines the two-layer formulation with the full two-equation model [87].

$$\lambda = \frac{1}{2} \left[1 + \tanh \left(\frac{Re_d - Re_y^*}{A} \right) \right] \quad 5.30$$

The value of A determines the width of the wall-proximity indicator. It is defined such that λ is calculated within 1% of its far-field value for a given variation of ΔRe_y .

$$A = \frac{|\Delta Re_y|}{\tanh 0.98} \quad 5.31$$

The turbulent viscosity from the $k-\varepsilon$ (μ_t) is blended with the two-layer value as follows.

$$\mu_t = \lambda \mu_t|_{k-\varepsilon} + (1 - \lambda) \mu \left(\frac{\mu_t}{\mu} \right)_{2layer} \quad 5.32$$

The two-layer ratio value is quantified by the following for natural convective flows (Xu correction).

$$\left(\frac{\mu_t}{\mu} \right)_{2layer} = \frac{0.544 y_v^*}{1 + 5.025 * 10^{-4} y_v^{*1.65}} \quad 5.33$$

The two-layer ratio for shear driven flows, Wolfstein correction, is given by the following.

$$\left(\frac{\mu_t}{\mu}\right)_{2layer} = 0.42 Re_d C_\mu^{1/4} \left[1 - e^{-\frac{Re_d}{70}}\right] \quad 5.34$$

The stress tensor and the mean strain rate are related through equations used with the Boussinesq approximation.

$$y_v^* = Re_d \sqrt{7.19 * 10^{-3} Re_d - 4.33 * 10^{-5} Re_d^2 + 8.8 * 10^{-8} Re_d^3} \quad 5.35$$

5.2.4 Turbulent Heat Transfer and Conditions

Using the RANS turbulence models defines the mean heat flux in the energy equation based on the Boussinesq approximation [88].

$$\bar{q} = -\left(k + \frac{\mu_t C_p}{Pr_t}\right) \nabla \bar{T} \quad 5.36$$

The turbulent eddy viscosity (μ_t) is given by the respective turbulence model, k is the fluid thermal conductivity, C_p is the specific heat, Pr_t is the turbulent Prandtl number, and \bar{T} is the mean temperature. Where non-flow boundaries exist in the domain, like walls, k and ε must satisfy the following.

$$\frac{\partial k}{\partial n} = \frac{\partial \varepsilon}{\partial n} = 0 \quad 5.37$$

The vector n is normal to the boundary face. Specified turbulence intensity (I) and turbulent viscosity ratio ($\frac{\mu_t}{\mu}$) derives the initial values of k and ε .

$$k = \frac{3}{2} (Iv)^2, \varepsilon = \frac{\rho C_\mu k^2}{\frac{\mu_t}{\mu} \mu} \quad 5.38$$

The local velocity magnitude (v) is taken for a local frame of reference. The boundary layer initialization of k and ε is taken with consideration of the local relative velocity (u_r), the wall-parallel non-dimensional velocity (u^+), and the wall y^+ value (given in Table A.4).

$$k = \sqrt{\left(\frac{1}{du^+/dy^+} - 1\right) \frac{\varepsilon \kappa v}{\exp\left(-\frac{1.0674 * 10^{-3}}{4} y^{+3}\right)}}, \varepsilon = \frac{1}{\max(17, y^+)} \frac{u_r^4}{\kappa v} \quad 5.39$$

The von Karman constant is the value for κ in this equation. The realizable k - ε model captures transition behavior under some flow conditions [89]. Transition refers the phenomenon of laminar to turbulence boundary layers.

5.3 Heat Transfer

The continua in the models are subject to radiative, convective, and conductive heat transfer. STAR-CCM+ solves for the heat transfer within a fluid or solid, between different fluids (gas/liquid interface), and between a fluid and solid. Solving heat transfer between a liquid and solid interface is done with conjugate heat transfer analysis. In conjugate heat transfer analysis, the energy equation is solved throughout the fluid and solid mesh domain with an implicit thermal coupling at the interface. Other conservation equations are solved for in fluid continua only. Heat transfer is calculated within single or multi-component fluids, between fluid streams, between fluid and solids, and within solids. The energy equation for fluids is as follows.

$$\begin{aligned} \frac{\partial}{\partial t} \int_V \rho E dV + \oint_A \rho H \mathbf{v} \cdot d\mathbf{a} \\ = - \oint_A \dot{q}'' \cdot d\mathbf{a} + \oint_A T \cdot \mathbf{v} d\mathbf{a} + \int_V f_b \cdot \mathbf{v} dV + \oint_A \sum_i h_i J_i d\mathbf{a} \end{aligned} \quad 5.40$$

The term with total energy, E , is the rate of increase of energy of a fluid particle (cell). The transfer of enthalpy across the fluid surface is accounted for in the total enthalpy, H , term. On the right-hand side of the equation the terms account for conduction heat transfer, viscous work, body forces, and enthalpy diffusion, respectively moving left to right. The total energy and total enthalpy have the following definitions where h is the static enthalpy.

$$E = H - \frac{p}{\rho}, H = h + \frac{|\mathbf{v}|^2}{2} \quad 5.41$$

The governing equation for energy transport within a solid is given by the following.

$$\frac{d}{dt} \int_V \rho C_p T dV + \oint_A \rho C_p T \mathbf{v}_s \cdot d\mathbf{a} = - \oint_A \dot{q}'' \cdot d\mathbf{a} + \int_V S_u dV \quad 5.42$$

The two terms on the left side of the equation are the stored cell's volumetric and surface thermal energy, respectively. The term with the heat flux vector, \dot{q}'' , quantifies the cells surface heat flux energy transfer. The term with the user-specified volumetric heat source within the solid, S_u , quantifies the cell thermal energy source.

5.3.1 Conduction

Heat conduction is the transfer of internal energy through microscopic reactions between particles composing a conductive substance. Energy flowing from high temperature particles to lower temperature particles takes place in solids, liquids, gases, and plasmas. Conduction occurs in solids by the molecule vibrations in the structural lattice and by diffusion of free electrons. Collisions between molecules and molecular diffusion is how conduction transfers heat in fluids and gases. The local heat flux is calculated using the law of heat conduction, i.e., Fourier's law. The integral form of the equation considers the amount of energy flowing into or out of a body. The differential form considers the local fluxes of energy, shown below.

$$\dot{q}'' = -k\nabla T \quad 5.43$$

The thermal conductivity (k) of a material can be strongly dependent on temperature, pressure (gas), spatial location, and sometimes exhibits anisotropic behavior which means it varies with orientation and is represented by a second-order tensor. The local heat flux vector (\dot{q}'') feeds the governing energy equations based on the temperature gradient (∇T) between cells. Conductive heat transfer is important to the resulting temperature field in the simulation domain. The conservation of energy and Fourier's law can be used to derive the heat diffusion equation, which allows for the temperature field to be quantified in steady or unsteady problems.

5.3.2 Convection

Convective heat transfer occurs by random molecular motion, i.e., diffusion within fluids and the overall movement of the fluid. Near the surface of a fluid interface where the velocity is zero, heat transfer occurs by diffusion only. As distance from the surface increases, heat is transported downstream and into the bulk fluid flow because of fluid motion. Convection is naturally occurring when buoyancy forces are induced or externally driven by some force. Newton's law of cooling governs convective heat transfer.

$$\dot{q}_s'' = h(T_s - T_{ref}) \quad 5.44$$

The law of cooling shows a linear relationship between local surface heat flux (\dot{q}_s'') and the difference between the local surface temperatures (T_s) and the fluid temperatures (T_{ref}). The linear relationship, however, is only approximate because the relationship can be

nonlinear in most real situations. Since flow conditions vary by surface location, \dot{q}_s'' and the local convective heat transfer coefficient (h) can vary spatially and temporally.

Natural convection occurs when temperature differences in a fluid cause bulk fluid movement. The convection is more likely and rapid if the density variances are higher between the mixing fluids and the vertical distance between the hot and cool fluids is higher. Natural convection is dampened when diffusion is more rapid, e.g., high flow rates, or the fluid is more viscous. The dimensionless Rayleigh number is an important quantity expressing the ratio of buoyant and viscous forces multiplied by the ratio of momentum and thermal diffusivities. Natural convection is possible when the Rayleigh number exceeds 10^5 . Fans, pumps, and other propelling devices induce forced convection, which is typically done to increase heat transfer rates. During heat transfer analysis, it is important to determine how much each mode of convective heat transfer contributes total heat transfer. A combination of the Rayleigh, Prandtl, and Reynolds numbers indicates how the heat transfer should be addressed.

$$\begin{aligned}\frac{Ra}{PrRe^2} &\gg 1, \text{ natural convection dominates} \\ \frac{Ra}{PrRe^2} &\ll 1, \text{ forced convection dominates} \\ \frac{Ra}{PrRe^2} &\cong 1, \text{ mixed convection dominates}\end{aligned}\tag{5.45}$$

The terms in Equation 5.45 are fundamental in nature aside from h , which is a constant of proportionality relating the fundamental parameters. The heat flux and surface temperatures are unambiguous, but the reference temperature (T_{ref}) offers variable options for satisfying the convection equation. The heat transfer coefficient is chosen based on the reference temperature. The expression for local surface heat flux can be defined as follows.

$$\dot{q}_s'' = \frac{\rho_f(y_c)C_{p,f}(y_c)u_\tau}{T^+(y^+(y_c))}(T_s - T_c)\tag{5.46}$$

$$y^+ = \frac{u_\tau y_c}{v_f}\tag{5.47}$$

Equating the terms in 5.45 and 5.46 defines a value for h in terms of the local flow conditions. The heat transfer coefficient can then be thought of as a function of fluid density

(ρ_f), fluid-specific heat capacity ($C_{p,f}$), wall shear stress-based velocity scale (u_τ), dimensionless temperature (T^+), and a modified Reynolds number (y^+). The normal distance of the near-wall cell (y_c) and the temperature of the near wall cell (T_c) influence the local heat flux. Standard wall functions (SWFs) are used to satisfy the flow physics in the boundary layer of fluids. Relationships for T^+ and u_τ in terms of Prandtl numbers, dimensionless velocity, and turbulent kinetic energy are given by the SWFs. Local heat transfer is quantified by the term in 5.46 or user specified as follows.

$$h = h_{user} \left(\frac{T_s - T_c}{T_s - T_{ref}} \right) \quad 5.48$$

In this way, h_{user} and T_{ref} can be defined such that the resultant \dot{q}_s'' can suit the requirements of simulation.

5.3.3 Conjugate Heat Transfer (CHT)

CHT is a problem solved across a contact interface of different materials such as a fluid and solid or two solid materials. The conservation of energy requires the total heat flux across an interface be conserved. Refer to Figure 5.1 for two boundaries defined at a contact interface.

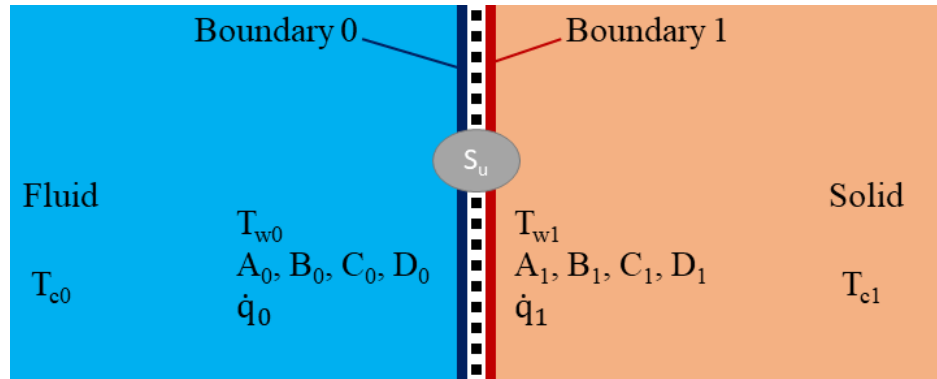


Figure 5.1. CHT interface analysis.

Energy conservation assures the following relationships.

$$\dot{q}_0 + \dot{q}_1 = -S_u \quad 5.49$$

$$\dot{q}_0 = A_0 + B_0 T_{c0} + C_0 T_{w0} + D_0 T_{w0}^4 \quad 5.50$$

$$\dot{q}_1 = A_1 + B_1 T_{c1} + C_1 T_{w1} + D_1 T_{w1}^4 \quad 5.51$$

The sum heat flux from boundary 0 (\dot{q}_0), boundary 1 (\dot{q}_1), and the specified heat source within the interface (S_u) must sum to zero. A, B, C, and D are linearized heat flux coefficients. The subscripts c and w in the temperature terms are the cell temperatures and cell surface temperatures for each cell, respectively. The heat flux across the interface can be expressed with respect to the interface temperature difference and the interface thermal resistance (R).

$$\dot{q}_0 = -\frac{S_u}{2} + \frac{T_{w1} - T_{w0}}{R} \quad 5.52$$

The four unknowns in 5.49 and 5.52 are solved analytically when there is no radiation term, i.e., the D heat flux coefficients are zero. If radiation is accounted for, the equations are solved numerically.

5.3.4 Radiation

Thermal radiation represents the conversion of thermal energy into electromagnetic energy from all matter that has a temperature greater than absolute zero. Charged particles in matter have thermal motion resulting in charge-acceleration and dipole oscillations. This behavior drives electrodynamic generation of coupled electric and magnetic fields, causing the emission of radiation. The maximum flux at which radiation can be emitted from a surface is given by the Stefan-Boltzmann law.

$$\dot{q}_{bb}'' = \sigma T_s^4 \quad 5.53$$

The local surface heat flux from a black body (\dot{q}_{bb}'') is a function of the surface temperature. It was based on this law to omit a radiation model from the current simulations. The heat flux at the O-SERTTA rod surface temperature and exterior tank wall are approximately four orders of magnitude higher. Natural circulation systems also have small temperature differences relative to other applications where radiative heat transfer must be considered.

5.4 Verification and Validation

Verification and validation (V&V) assess the accuracy of CFD results. Validation is the primary goal which is defined by the process of determining the level of accurate representation of the real world the model provides. Verification, which must proceed

validation, is the process of ensuring and estimating the numerical accuracy of the code mathematical model solutions and the calculations. The V&V procedures followed in this research conform to the American Society of Mechanical Engineers (AMSE) V&V 20 Standard [90]. The core validation metrics are the validation comparison error and uncertainty (E and u_{val} , respectively).

$$E = S - D = \delta_S - \delta_D, u_{val} = \sqrt{u_{num}^2 + u_{input}^2 + u_D^2} \quad 5.54$$

The values S and D represent the solution value and the real-world data value, respectively. The validation comparison uncertainty is determined by the values, from left to right, the uncertainty of the numerical solution of equations, the input parameters, and the data values. Verification is concerned with determining u_{num} , the uncertainty of the numerical solution caused by numerical error, δ_{num} . Code verification and solution verification are the comprising methods for determining u_{num} . Establishing the correctness of the code by discretization convergence tests and monitoring the code solution toward a benchmark comparison is done in code verification. The best benchmark solution is an analytical solution with sufficient complexity to capture all the relevant physics being modeled in the code. The analytical model presented here satisfies the conditions for code verification. Solution verification addresses estimates of error and uncertainty. The typical uncertainty estimate provides an interval that the value being computed falls in with some probability. Uncertainty targets are commonly the 95% confidence interval, which can be calculated by Roache's Grid Convergence Index (GCI). Uncertainty estimation by GCI can be done in five steps, outlined by Table 5.5. The Factor of Safety value is quantified through empirical studies. A value of 1.25 often results in a GCI with a 95% confidence interval [91].

Table 5.5. Roache's Grid Convergence Index (GCI) five-step procedure outline.

Step	Description	Relevant Equations
1	Define representative grid size, h , based on total cells (N) and total cell volume (V).	$h = \left[\frac{V}{N} \right]^{1/3}$
2	Select three significantly different grid sizes and monitor key variables during simulation. The grid refinement factor (r) should be between 1.1 and 1.5.	$r = \frac{h_{coarse}}{h_{fine}}$
3	Calculate the apparent order, p , for the different key variables, φ . Let $h_1 < h_2 < h_3$, $r_{21} = \frac{h_2}{h_1}$, and $r_{32} = \frac{h_3}{h_2}$. When r is not constant the key variables must be used to correct; $\varepsilon_{32} = \varphi_3 - \varphi_2$ and $\varepsilon_{21} = \varphi_2 - \varphi_1$.	$p = \left[\frac{1}{\ln(r_{21})} \right] \left[\ln \left \frac{\varepsilon_{32}}{\varepsilon_{21}} \right + q(p) \right]$ $q(p) = \ln \left(\frac{r_{21}^p - s}{r_{32}^p - s} \right)$ $s = 1 * \text{sign} \left(\frac{\varepsilon_{32}}{\varepsilon_{21}} \right)$
4	Calculate extrapolated values.	$\varphi_{ext}^{21} = \frac{r_{21}^p \varphi_1 - \varphi_2}{r_{21}^p - 1}$
5	Calculate and report error estimates and the apparent order. The Factor of Safety, FS , can be taken as 1.25 for three-grid studies.	$e_a^{21} = \varphi_1 - \varphi_2 $ $GCI_{fine}^{21} = \frac{FS * e_a^{21}}{r_{21}^p - 1}$

To determine u_{num} , an assumption must be made about error distribution of the problem. For solutions with appropriate behavior and high resolution, the error distribution is Gaussian about the extrapolated solution. Analyses of this situation leads to an expansion factor (k) to obtain a conservative value for u_{num} .

$$u_{num} = \frac{U_{num}}{k} = \frac{GCI}{1.15} \quad 5.55$$

The procedure is independent of time and so the calculations must be done for each relevant timestep during simulation. Upon completion of code verification and solution verification the CFD values can be assigned a numerical uncertainty. Validation is the next step to assess the overall model accuracy.

The next term to address in the simulation value uncertainty is u_{input} , the estimated uncertainty caused by simulation input parameters. For the CFD presented here, u_{input} is determined with the local view of uncertainty. The local view is also known by the

sensitivity coefficient method, perturbation method, mean value method, and others. The global view, i.e., Monte Carlo or sampling method, was not used because of high computational costs and u_{input} is not expected to be high relative to u_{num} . The input uncertainty propagation for a simulation result, S , with n uncorrelated input parameters is given by Equation 5.56.

$$u_{input}^2 = \sum_{i=1}^n \left(\frac{\partial S}{\partial X_i} u_{X_i} \right)^2 \quad 5.56$$

X_i is some input parameter and u_{X_i} is the corresponding standard uncertainty. The simulation result can be taken for any quantity of interest in the solution domain. The partial derivative in 5.56 is the sensitivity coefficient of the solution result with respect to some input parameter. Input parameter uncertainty (u_{X_i}) is often quantified during prior experiments. For values that do not have reported uncertainties, expert judgment must be used to quantify the input parameter uncertainty. The sensitivity coefficients are determined with the finite difference (FD) technique in parameter space when the analytical model was deemed inadequate. Calculating the coefficients with this method requires perturbing the input parameter in question while holding other parameters constant and running both simulations to compare solutions.

$$\frac{\partial S}{\partial X_i} = \frac{S(X_1, \dots, X_i + \Delta X_i, \dots, X_n) - S(X_1, \dots, X_i, \dots, X_n)}{\Delta X_i} + O(\Delta X_i) \quad 5.57$$

Now there is only one term left in quantifying validation uncertainty, u_D . Experimental uncertainty analysis is done with test uncertainty method provided in ASME PTC 19.1 [69]. This method uses the standard deviation for each error source to calculate uncertainty for some measured variable. Random and systematic classifications are used to address the uncertainty of measurement and uncertainty of test result. The systematic uncertainty of a measurement is given by 5.58.

$$b_i = \sqrt{b_{i_1}^2 + b_{i_2}^2 + \dots + b_{i_n}^2} \quad 5.58$$

For each systematic error source present, an estimation of the systematic standard uncertainty, b_r^2 , must be obtained from previous experience, calibration data, analytical models, or engineering judgement. The random standard uncertainty (s_r^2) for the measurement of a variable must be considered. The measurements to calculate random

standard uncertainty must be taken during test conditions and time frame. Multiple sets of data over time while holding all other parameters constant will identify random variation. The uncertainty of a result for the total amount of measured variables is given by 5.59.

$$u_D = \sqrt{b_r^2 + s_r^2} \quad 5.59$$

Where b_r^2 is the systematic uncertainty of the measurement result.

$$b_r^2 = \sum_{i=1}^J \left(\frac{\partial r}{\partial X_i} b_i \right)^2 + 2 \sum_{i=1}^{J-1} \sum_{k=i+1}^J \frac{\partial r}{\partial X_i} \frac{\partial r}{\partial X_k} b_{ik} \quad 5.60$$

The random standard uncertainty of the result is given by the following.

$$s_r^2 = \sum_{i=1}^J \left(\frac{\partial r}{\partial X_i} s_i \right)^2 + 2 \sum_{i=1}^{J-1} \sum_{k=i+1}^J \frac{\partial r}{\partial X_i} \frac{\partial r}{\partial X_k} s_{ik} \quad 5.61$$

The terms b_{ik} and s_{ik} are the covariance of the uncertainties. When systematic error for two separately measured variables are correlated, the covariance is nonzero. The systematic error covariance term b_{ik} , is the sum of the systematic standard uncertainties of variables i and k . The random standard uncertainties for different measured variables are considered independent, i.e., $s_{ik} = 0$. This analysis considers the experimental result to have a value, a fixed (unknown) error, and a systematic component of uncertainty that incorporates random uncertainty.

Although exact analytical equations are not possible for most CFD codes, there is a general procedure to generate accurate analytical solutions for verification known as the method of manufactured solutions (MMS). This was not required for the presented CFD simulation results because the input uncertainty is assumed to be negligible compared to the experimental and numerical uncertainty. Additional verification tools are integrated into STAR-CCM+ which provides more rigor to the verification of the CFD simulations. Unstructured refinement, which was implemented for the shown research, is more thorough than structured grid refinement in code verification when using the GCI procedure. With the implemented factor of safety in the GCI procedures, it was assumed the experimental and numerical uncertainties were sufficient to quantify the validation error uncertainty with a high confidence interval.

5.4.1 Experimental Measurement Uncertainty

The determination of u_D for the thermocouples is a straightforward process. Type K Omega thermocouples have standard errors of 2.2 °C (3.96 °F) [92]. This standard error can be used for the systematic uncertainty in Equations 5.59 and 5.60. The random standard uncertainty can be estimated as the standard deviation of many experimental measurements under the same conditions, i.e., steady state or pulse transients. Addressing the experimental uncertainties for power and the LUNA fiber measurements is done in a similar manner. The O-SERTTA power is quantified by the following relationship.

$$P = V * I \quad 5.62$$

Where the voltage measurements and current measurements have systematic and random uncertainties. The temperature measurements obtained from the fiber optic sensors were corrections of the direct strain measurements to temperature via an in-situ calibration with a local thermocouple. The temperature (T) calculation as a function of the measured strain is calculated by 5.63.

$$T = \mu\epsilon * m + b \quad 5.63$$

Where m and b are the calibration correction slope and y-intercept, respectively. The systematic uncertainty in the micro strain ($\mu\epsilon$) measurements are quantified by the manufacturer [93]. The random uncertainty is quantified in the same way as the thermocouples. Uncertainty for fluid properties assigned by the IAPWS models are given in literature [94].

6 ANALYSIS METHODS

The CFD model was designed to preserve the attributes of O-SERTTA relevant to the flow conditions achieved during steady state experimental operation. The geometry was maintained close to the experiment while neglecting small components that are assumed to negligibly affect the fluid conditions, e.g., screws and small gaps. Geometrical and boundary condition assumptions have been made for the CFD models. The CFD geometry is shown in Figure 6.1 where the simplifications can be seen. The cross-sectional mesh is also shown for two different flow areas. The spider component and the azimuthally symmetric slots were included in the geometry because they were expected to provide resistance to flow during operation.

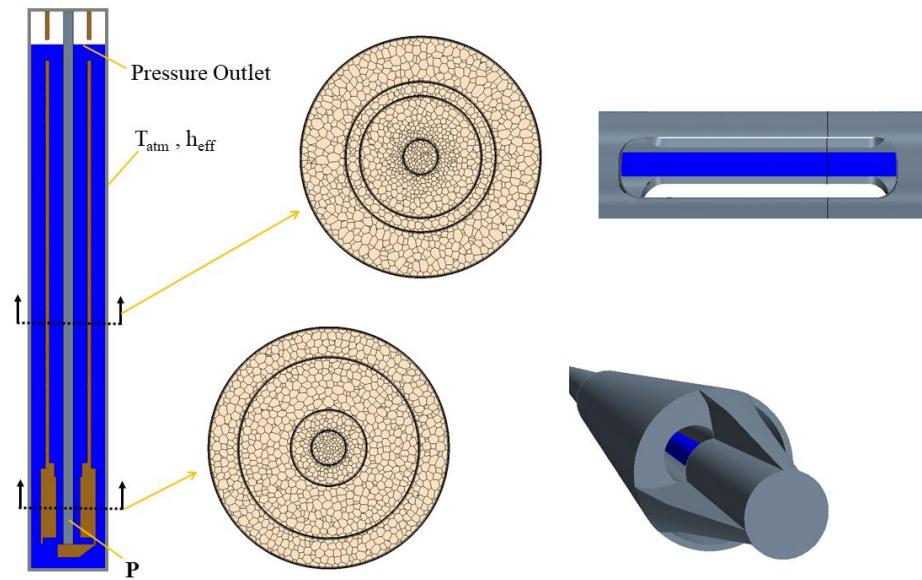


Figure 6.1. CFD geometrical representation of O-SERTTA along with cross sectional mesh of 2 mm base size mesh.

Although the spider component was maintained, the component was simplified to be smooth along the edges of the outer core heater tube. The flow pipe, core heater, core heater tube, and spider component are assumed to be a single structure within the fluid facilitating flow. The geometrical simplifications are not expected to impact flow significantly. The axial dimensions of the model and O-SERTTA are shown in Figure 6.2. Radial dimensions are the same as shown in Figure 3.5.

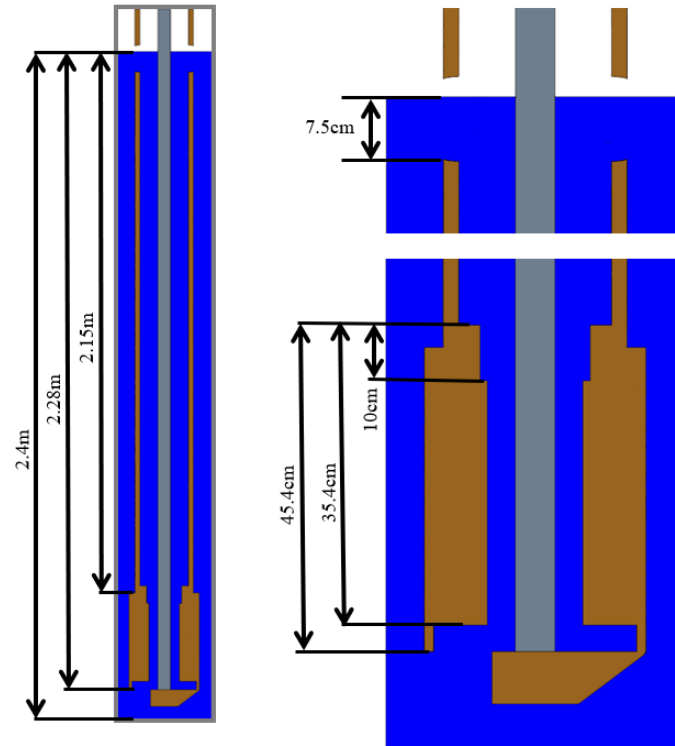


Figure 6.2. Axial dimensions of the CFD model as measured on O-SERTTA.

The heater rod is modeled as a separate solid structure in contact with the remaining internals at the bottom surface of the rod. The internal components are made of similar metal where the differences in material properties are not expected to change the flow conditions. The flow pipe material is used to model the solid continua because the heat transfer from the riser to the downcomer is of concern for analyzing the flow. The core heater has epoxy material in the internals, which acts as an insulative barrier. The epoxy was not expected to be of high influence because the change in temperature between the downcomer and riser surface temperatures of the core heater as shown by the CFD results and experimental estimates is not large enough for the difference in thermal resistances to substantially make a difference. This is based on the fact the core heater is horizontal to the heated section and is seeing the cool inlet temperatures of the fluid very close to the other side of the core heater in the downcomer. The epoxy material was designed to provide resistance when the core heater was in use which is not the case for this research. The epoxy is not considered to influence the overall operation of O-SERTTA because convective heat transfer dominates over conduction in the heated portion of the riser.

6.1 Initial Conditions and Startup

The initial conditions of the simulation differ from experimentation because of the long experimental startup process. Heating the fluid of the O-SERTTA facility requires thousands of seconds which is too computationally expensive to emulate in CFD. Therefore, the initial conditions and startup simulation for the CFD models differ from experiment. CFD simulation results depend strongly on the initial and boundary conditions and is expected to be an impactful limitation to the verification and validation of the models.

6.1.1 O-SERTTA

Initiation of the experimental facility involves filling the primary tank to a specific water height level and then heating the fluid to operational temperature allowing space for fluid thermal expansion. When the fluid reaches operating temperature, the power is dropped to approach steady state operation. The qualification of steady state status is dependent on the temperature, pressure, and power measurements taken with the O-SERTTA instrumentation. The startup procedure of O-SERTTA with respect to the temperature measurements at certain locations is shown in Figure 6.3.

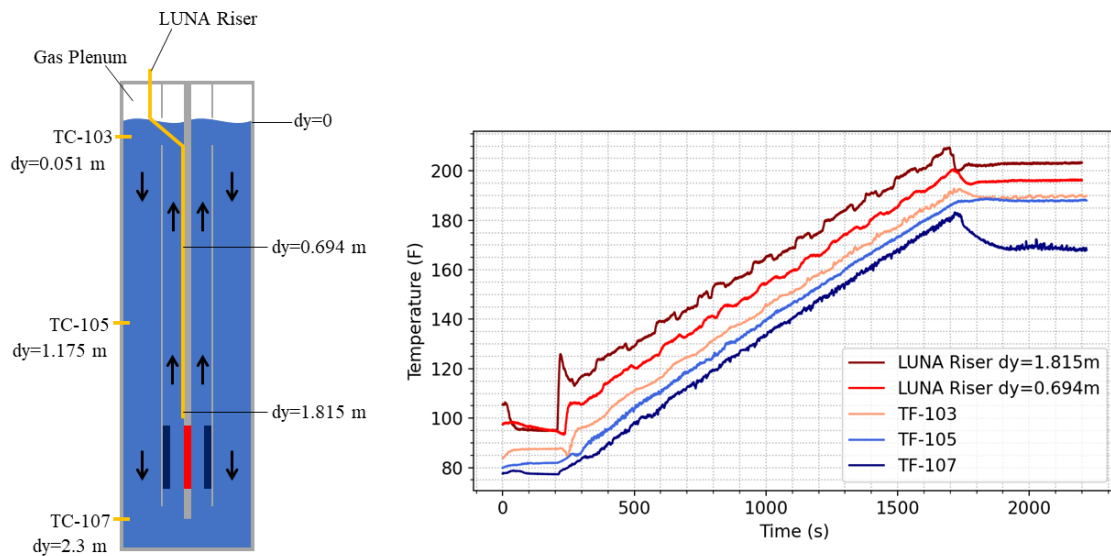


Figure 6.3. Local O-SERTTA temperature measurements during warm up.

The required time to reach operating conditions is approximately 1700 seconds as seen in the temperature plot. After the fluid reaches satisfactory temperatures, the steady state characteristics can be seen in the end of the plot when the power is lowered. Determining steady state operation is done quantitatively with similar criteria applied to the CFD simulations.

6.1.2 CFD Models

The initiation of the CFD models assumed that applying the experimental steady state operating boundary conditions to the simulation domain, where the initial temperature is the operational temperature, leads to the same steady state conditions that heating up the fluid leads to. Transient CFD simulation was implemented because of the divergence using STAR-CCM+ steady state solvers. The transient simulation approaches constant values after a few hundred seconds. When the CFD solutions are not changing based on convergence criteria, the CFD simulation is assumed to be at the steady state operation. The steady state must also satisfy the verification procedures discussed earlier. The first 420 seconds of simulation can be visualized in Figure 6.4.

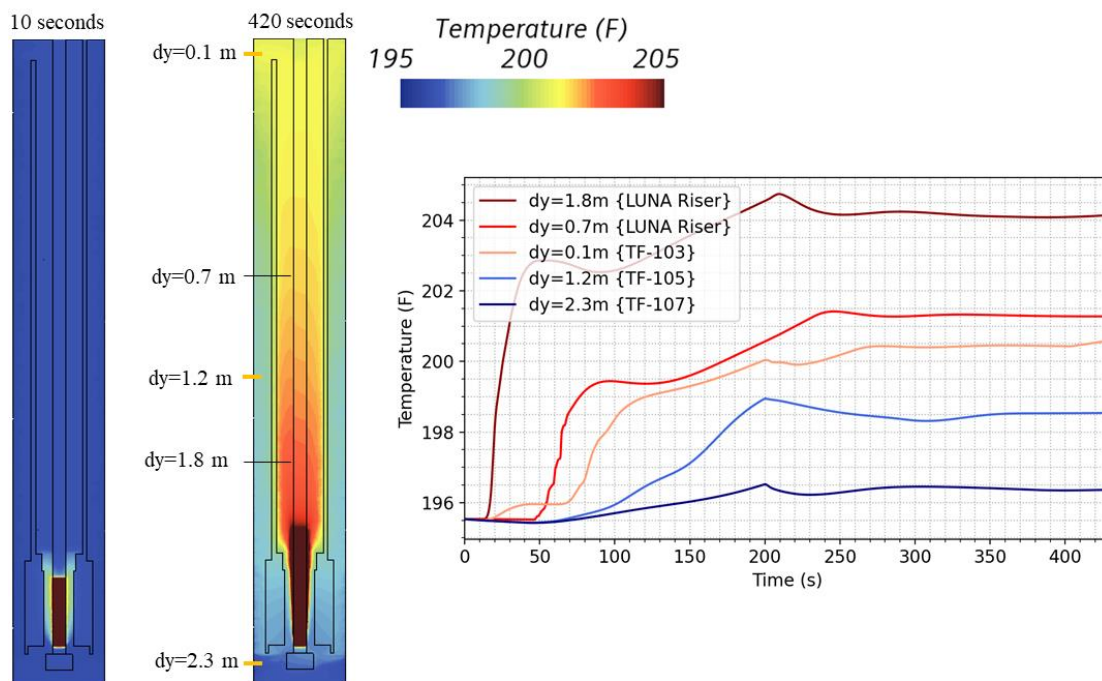


Figure 6.4. Local CFD temperature estimates over about 400 seconds of simulation.

Initially, the CFD model input for the effective heat removal coefficient to the atmosphere is lower than the determined value for the experimental steady state value. This was done for the first 200 seconds to allow for an artificial heat up procedure since the full experimental heat up is not feasible to simulate. The artificial heat up period was desired to capture any effects on the resultant system operation that may incur because of increasing fluid temperatures. At 200 seconds, the heat transfer value was changed to the steady state experimental value so the simulation could approach steady state. During simulation, the rod power was maintained at the experimental steady state value. The reasoning for lowering the heat removal instead of increasing the power was because of mesh constraints on convergence. Finer meshing is required for higher heat flux values, which increases computational time. For the other simulations discussed, the heat transfer coefficient was maintained throughout to compare to the simulation with an initial period of fluid temperature increase.

6.2 Steady State Operation

The steady state operation for experiment and simulation were qualified using similar criteria. The criteria for time rate changes of certain parameters are given in Table 6.1. The criteria in the CFD simulations were enforced with global and local monitors in the mesh domain. The criteria for the experiment were determined with the available instrumentation.

Table 6.1. Steady state determination criteria.

Quantity	Data	Criteria	Description
Average Fluid Temperature	CFD	$< \frac{\Delta 0.25 \text{ }^{\circ}\text{F}}{1 \text{ min}}$	Average fluid cell temperature.
	Experiment		Average temperature reading from LUNA and TFs (rod TCs excluded).
Power	CFD	$< \frac{\Delta 3 \text{ W}}{3 \text{ min}}$	Calculated rod heat transfer.
	Experiment		Power input.
Local Fluid Temperatures	CFD	$< \frac{\Delta 1 \text{ }^{\circ}\text{F}}{2 \text{ min}}$	Relative locations of experimental measurements, e.g., TF-103 location.
	Experiment	$< \frac{\Delta 1 \text{ }^{\circ}\text{F}}{5 \text{ min}}$	Local thermocouple and LUNA measurements.
Mass Flow Rate	CFD	$< \frac{\Delta 0.001 \frac{\text{kg}}{\text{s}}}{1 \text{ min}}$	Average local mass flow rates along riser and downcomer.
	Experiment	NA	No method to measure steady state mass flow rates.

The steady state criteria were chosen so negligible changes in the system occur as time continues. As discussed earlier, natural convection systems may have multiple steady states or no steady state operation. Therefore, it is expected that no perfectly steady instrumentation readings or CFD monitor readings will be seen.

6.2.1 O-SERTTA Steady State Parameters

Steady state of O-SERTTA is said to be achieved when the temperature, pressure, and power readings remain relatively unchanged with respect to time per the criteria in Table 6.1. For the operating temperature of O-SERTTA, the steady-state values used for the CFD model input were averaged over several hundred seconds of steady state operation at said temperature. The time averaged values are given in Table 6.2. The calculations of the heat transfer coefficient were done by Equations 3.1, 3.2, and 3.3 and the input values are also shown.

Table 6.2. Steady state experimental values for CFD model input.

Parameter	Value	Unit	Notes
T_f	195	°F	Average fluid temperature during steady state measured by thermocouples.
P	389	W	Steady state operating input power.
h_{eff}	10	W/m ² K	Effective overall heat transfer coefficient from convection and radiation to the atmosphere.
P_0	0	psig	Pressure in the gas plenum assumed to be equal to pressure at the gas/liquid interface.
T_{atm}	70	°F	Atmospheric temperature averaged over room temperature and nearby thermocouple temperatures.
L_l	107.9	inH ₂ O	Liquid level as measured by a linear differential pressure instrument.

Note the steady state experimental quantities do not have uncertainties associated with them. The uncertainties are quantified as discussed in the verification and validation section and are shown in the results section. The estimated heat removal from the coils is equivalent to approximately 6% of the total heat removal based on the rod power and conservation of energy assuming constant heat flux on the heat removal portion of O-SERTTA. Since the coils cover about 6% of the surface area it is expected the coils were removing heat with approximately the same heat transfer coefficient as the remainder of the system.

6.2.2 Heat Transfer Coefficient

As discussed earlier there is usually at least a 10% uncertainty associated with effective heat transfer coefficient estimates. The heat transfer coefficient for O-SERTTA was estimated based on experimental studies given in the literature and basic heat transfer equations (Equations 3.1, 3.2, and 3.3) assuming constant heat flux at the atmospheric boundary of the system. The resulting calculations are shown in Table 6.3.

Table 6.3. Heat transfer calculations to estimate the effective heat transfer coefficient.

Parameter	Value	Unit	Notes
\dot{Q}_{total}	389	W	Total heat removal assuming the rod power is thermal power into the system.
$\dot{Q}_{6\%}$	23.34	W	Assuming heat flux is constant on the surface area, this is 6% surface area heat removal.
\dot{Q}_{coils}	23.4	W	Estimated heat removal with coils based on Equation 3.3.
$h_{eff, calc}$	10.09	W/m ² K	Calculated heat transfer coefficient from Equations 3.1 and 3.2.
$h_{eff, lit}$	10	W/m ² K	Effective heat transfer coefficient for similar conditions from experimentation.

The literature suggests a value close to the estimated value based on the equations. 6% of the surface area is expected to remove approximately the same amount of heat as the cooling coils. Since the cooling coils cover 6% of the surface area the effective heat transfer properties are assumed to be the same elsewhere on the surface. The coils were thus modeled as part of the atmospheric boundary with the same heat transfer coefficient.

6.3 Pulse Operation

Pulse operation of O-SERTTA was used to characterize the steady state velocity. Since there is no flow meter in the experimental facility, alternative methods were required to determine experiment flow rates. Ramping of the heater rod was done after steady state was achieved. Increasing the power then reducing back to steady state power, called pulsing, of the heater rod power produces thermal slugs that are clearly observable with the LUNA fiber optic sensor in the riser section. If the pulses are powerful enough (high power for long enough time), the thermocouples in the downcomer also respond to the thermal slugs. A methodology, called center of heat (COH) detection, used for tracking the center of the thermal slugs can be used to estimate the fluid velocity [95].

When the power is pulsed to create a thermal slug that flows along the loop path, the response of three different temperature measurements along the flow path is given by Figure 6.5 (a). The circles for each location represent the location of the peak temperature seen with respect to time. Tracking the location each peak is observed versus time allows

for simple quantification of a first approximation of the velocity. However, since convective heat transport results in asymmetric profiles, the peak location of the heat pulse does not identify the center of the thermal slug. Tracking the center of heat, which is the time at which the integral reaches half the maximum value, results in more precise estimations of the flow velocities. Tracking the center of heat can be visualized in Figure 6.5 (b).

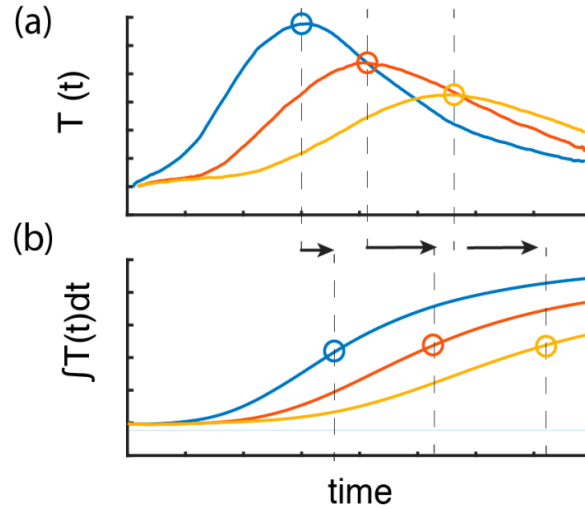


Figure 6.5. Center of heat detection example of the (a) center of pulse vs (b) center of heat detection [95].

This method provides verification of the directional movement of the fluid in the system. Since power is being increased during pulse transients, the flow is expected to accelerate. Analyzing a handful of velocities estimated by the COH method for different pulses allows for an estimated steady state velocity. A methodology for quantifying the uncertainty of the estimated velocities using this method has not been proposed. Therefore, the velocities calculated from this method are assumed to be only rough estimates of the flow rates in O-SERTTA. An example of the power profile during pulse transients is shown in Figure 6.6.

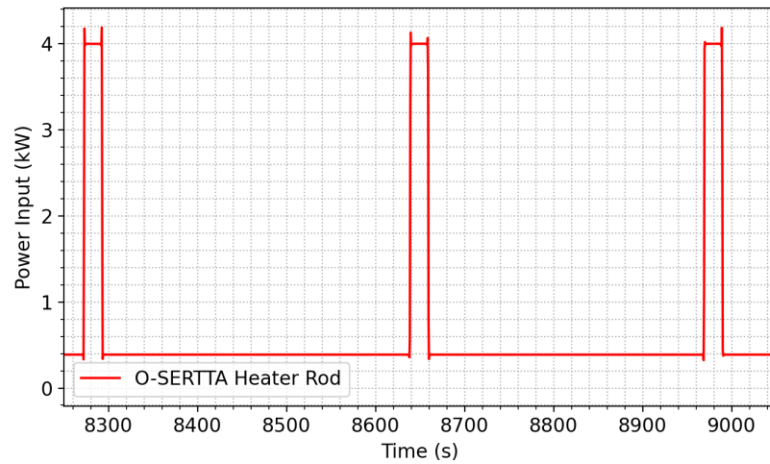


Figure 6.6. Power level of O-SERTTA heater rod during pulsing.

6.4 Parametric Study

The parametric study was done for a simplified CFD model to eliminate the asymmetries existing in the experimental facility. The spider component is uniquely shaped and could be omitted from O-SERTTA if accommodation of LVDTs was not necessary. As such, for the simplified geometry, it was modified as shown in Figure 6.7. The azimuthally symmetric slots at the turnaround region were also removed to be fully symmetrical. The last geometrical change is the removal of a minor expansion in the heated region of the riser. In the experiment, this change is from the core heater inner radius to the upper flow pipe coupling component, which is the green component shown in Figure 3.5. The flow expansion is minor and not expected to credibly impact flow. Other than the geometrical simplifications, the model is conserved in terms of physics and meshing.

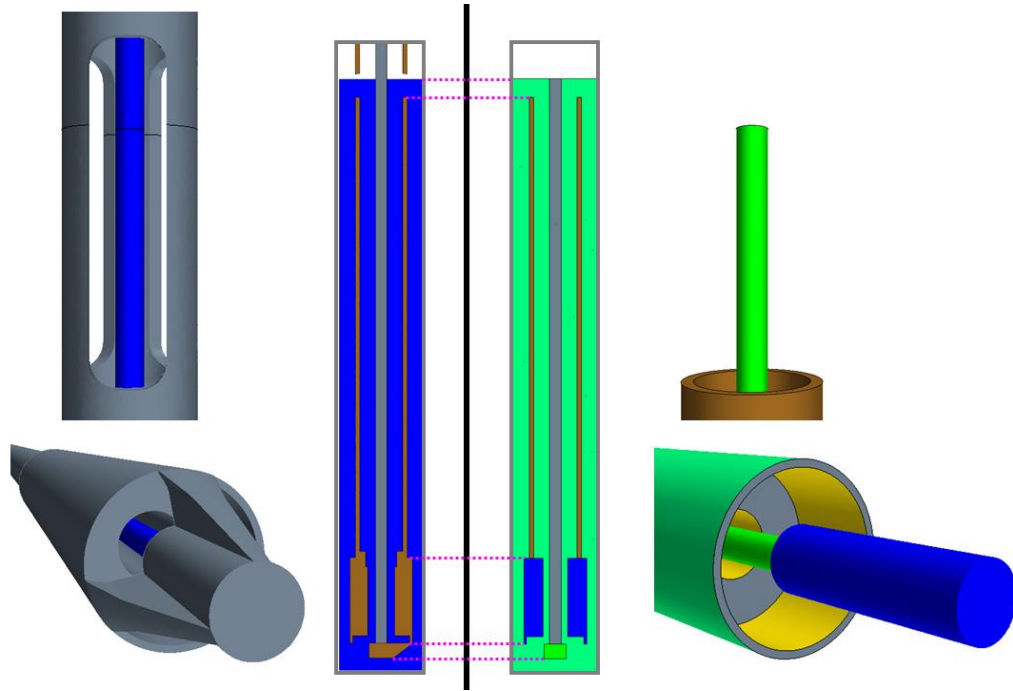


Figure 6.7. Comparison between O-SERTTA CFD model and simplified CFD model.

6.4.1 Parametric Modifications

The simplified geometry was used to evaluate geometrical effects on the system flow rates and temperature distributions. Modifications of the design are illustrated in Figure 6.8. The core configuration was left unchanged during the study because the geometrical concerns are of the flow pipe above the heated region. Literature does not demonstrate how the radius of the flow pipe impacts flow. The analytical model suggests the flow rate increases as the riser flow area decreases, however, at some point the frictional forces outweigh the benefit of increased fluid temperature differential which occurs when the riser flow area is smaller. These results are shown by the CFD modeling discussed later.

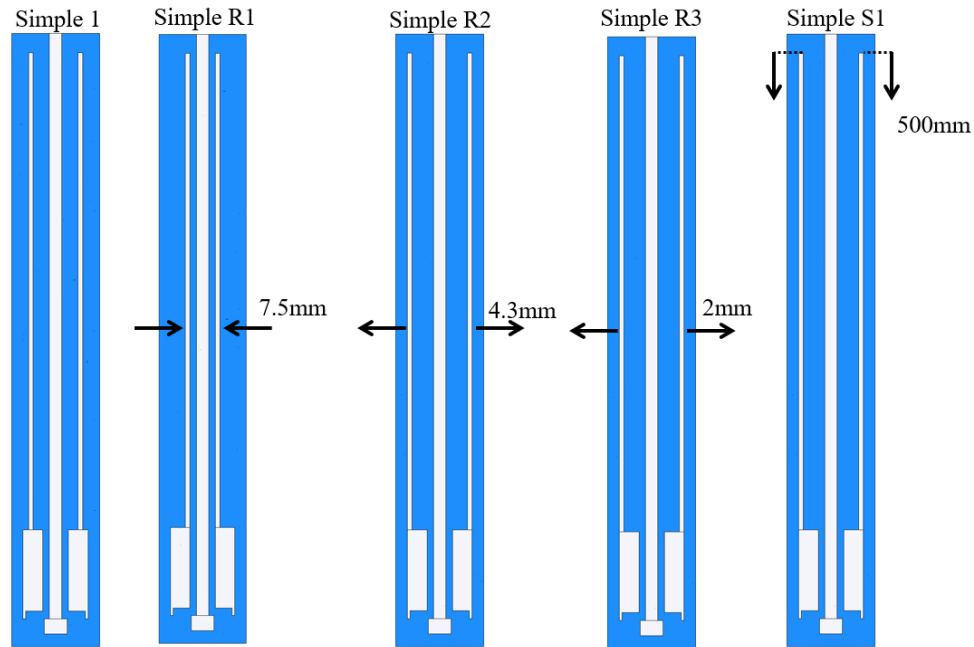


Figure 6.8. Geometrical modifications for CFD sensitivity study.

Results presented are based on the simulations shown in Table 6.4. There are four different solutions using the O-SERTTA geometry and five using the simplified geometry. A description of the variation to the geometry or physics is shown for each simulation. The notation for the model solutions is used in the results section.

Table 6.4. Various model nomenclature and the variation details.

Model	Geometry Type	Heat Transfer Coefficient	Variation
O-SERTTA 1	O-SERTTA	$h(0)=1$, $h(200)=10$	None, V&V model
O-SERTTA 2	O-SERTTA	$h=1$	None
O-SERTTA 3	O-SERTTA	$h=10$	None
O-SERTTA 4	O-SERTTA	$h=10$	Xu correction
Simple 1	Simplified	$h=10$	Normalized
Simple R1	Simplified	$h=10$	Radius equal to test section area
Simple R2	Simplified	$h=10$	Radius equal to core heater tube
Simple R3	Simplified	$h=10$	Radius 0.002m larger
Simple S1	Simplified	$h=10$	Lower flow pipe

6.5 Dimensionless Quantities

The dimensionless temperature is given by the following formula for the testing conditions.

$$\theta = \frac{T - T_{atm}}{T_{sat} - T_{atm}} \quad 6.1$$

The saturation temperature is water saturation temperature at atmospheric pressure and the atmospheric temperature is given in Table 6.2. The flow results are given by velocities, mass flow rates, or the classical Reynolds number as a function of fluid density (ρ), viscosity (μ), speed (u), and the hydraulic flow diameter (d_h).

$$Re = \frac{\rho u d_h}{\mu} \quad 6.2$$

The hydraulic diameter is four times the cross-sectional flow area divided by the wetted perimeter. Velocities are reported for the experimental comparison with the CFD results. The experimental measurement output is in velocity, so a direct comparison is made. Mass flow rates are given for the numerical uncertainties because it is the direct output of most of the monitors set up in the simulation domain.

7 RESULTS

The verification and validation of the O-SERTTA CFD model is shown and discussed. The calculations from the verification procedures outlined earlier are plotted. The LUNA fiber optic sensors the micro strain (μs) instead of temperature. Assuming only thermal strain is imparted on the fiber optic sensors allows for direct conversion of strain to temperature. The results of the calibration curve for conversion of strain to temperature is discussed. Validation error of the CFD simulation is shown to justify the model's ability to predict the flow phenomena. A discussion of the parametric study follows with mention of geometrical parameters, the two-layer correction in the turbulence closure, and initial condition dependence. Most of the results shown are casted in dimensionless forms to display generality of the application to more broad situations as discussed in the analysis methods section.

7.1 Verification

General code verification using the analytical model, residual monitoring, and engineering quantity criteria was done for all the models presented. The GCI study, however, was only conducted for the model representative of O-SERTTA. It is assumed from the similarities between the O-SERTTA model and the simplified or alternate condition models that the numerical uncertainty quantified is approximately the same for each model. Several mesh validity and quality verification tools are available in STAR-CCM+. The cell skewness angle, face validity, cell quality, volume change, and Chevron quality metrics were evaluated across the domain for each simulation. The root mean square (RMS) residual values were monitored to indicate convergence and observe a drop of at least two orders of magnitude as suggested by the ASME V&V 20 Standard [90]. The mass flow rate in the riser compared to the mass flow rate in the downcomer were verified monitored to ensure a physical solution was being simulated. The GCI study follows the procedure shown in Table 5.5. As a demonstration of the equation and associated values the Table 7.1 shows the procedure followed for the mass flow rate of three different mesh sizes. The results are shown on plots later in the section.

Table 7.1. Mass flow rate Roache's Grid Convergence Index (GCI) value examples.

Parameter	Equation	Value	Unit
h_3	$h = \left[\frac{V}{N} \right]^{1/3}$	0.00126	Dimensionless
h_2		0.00118	Dimensionless
h_1		0.00108	Dimensionless
r_{21}	$r = \frac{h_{coarse}}{h_{fine}}$	1.09119	Dimensionless
r_{32}		1.07313	Dimensionless
p	$q(p) = \ln \left(\frac{r_{21}^p - s}{r_{32}^p - s} \right)$	6.94000	Dimensionless
$q(p)$	$q(p) = \ln \left(\frac{r_{21}^p - s}{r_{32}^p - s} \right)$	0.21500	Dimensionless
s	$s = 1 * \text{sign} \left(\frac{\varepsilon_{32}}{\varepsilon_{21}} \right)$	1	Dimensionless
φ_{ext}	$\varphi_{ext}^{21} = \frac{r_{21}^p \varphi_1 - \varphi_2}{r_{21}^p - 1}$	0.02770	kg/s
ea_{32}	$e_a^{21} = \varphi_1 - \varphi_2 $	0.00220	kg/s
ea_{21}		0.00266	kg/s
GCI_{21}	$GCI_{fine}^{21} = \frac{Fs * e_a^{21}}{r_{21}^p - 1}$	0.00400	kg/s
GCI_{32}		0.00436	kg/s

7.1.1 Determining Center of Heat Sink

The method for quantifying the center of the heat sink is like calculating the center of heat as shown in Figure 6.5. Assuming the heat transfer coefficient is approximately constant around the primary exterior, the heat flux is highest where the temperature between the atmosphere and the fluid temperatures near the wall is the highest. The center of the heat flux is determined by taking the integral of the difference in temperature of the fluid from the atmosphere taken with respect to the baseline. The baseline is the minimum difference in temperature between the fluid and atmosphere, which is in the bottom stagnant region of the concentric tube thermosyphon design. Considering experimental data from O-SERTTA, the center of the heat sink can be quantified by the axial location at which the integral reaches half the maximum value. The change in temperature respective of the

baseline and the quantified integral based on a binomial fit of the experimental data is shown in Figure 7.1.

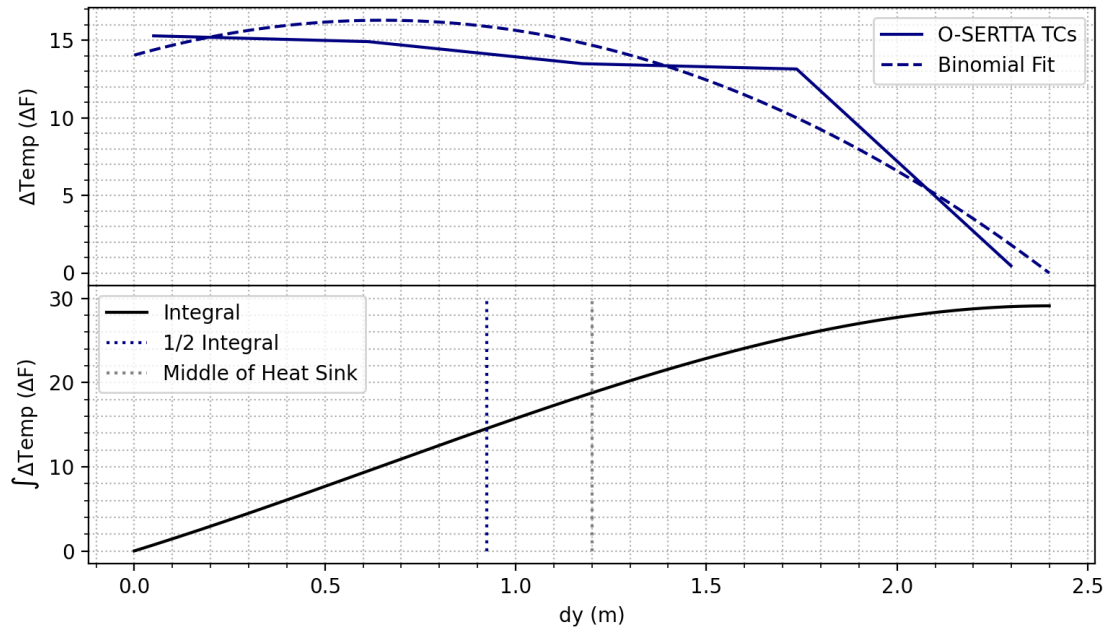


Figure 7.1. Center of heat removal determination plots.

The middle of the heat sink is estimated as the $\frac{1}{2}$ integral which is the estimate of the center of heat flux profile. The axial location of the $\frac{1}{2}$ integral is 0.925 m from the gas plenum as compared to the physical center of the primary boundary which is 1.2 m. This approach results in a Δz higher than if the center of the heat sink was taken as the physical center. This is expected because heat flux is higher near the top of the primary boundary where heat is being removed at higher rates.

7.1.2 Mesh Independence

The GCI study was conducted to evaluate the numerical uncertainty of relevant temperature and flow distributions. Locations in the simulation domain representative of the O-SERTTA instrumentation are included in the GCI study for validation comparison. Since all the internal components of the O-SERTTA system are modeled as a grouped component in the CFD model aside from the heater rod, the grouped components are the flow pipe. The temperatures shown are averaged around the circumference of the respective radial location discussed. The flow pipe surface temperatures averaged around

the circumference are shown in Figure 7.2. The temperatures are shown as a function of distance into the water from the gas plenum (dy).

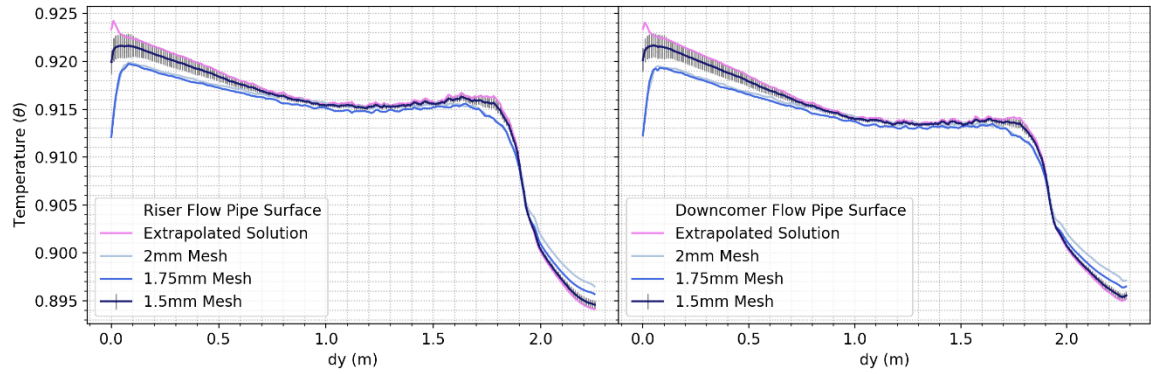


Figure 7.2. Flow pipe CFD near surface temperatures with numerical uncertainty.

The solution of the flow pipe surface temperatures has a maximum numerical uncertainty in the dimensionless temperature of about 0.003 (0.43 °F) which is about 0.15% uncertainty. The surface temperature of the heater rod is less independent of the mesh size. The dimensionless temperature with respect to distance from the gas plenum is shown in Figure 7.3.

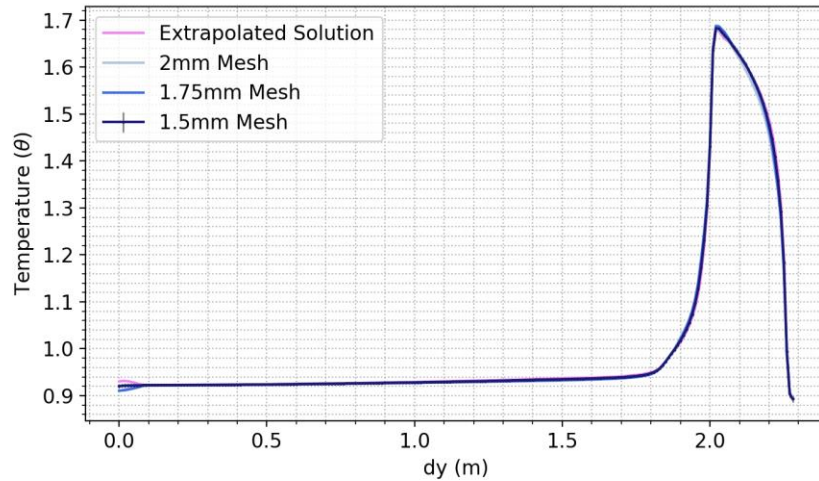


Figure 7.3. Rod CFD surface temperatures with numerical uncertainty.

The maximum uncertainty in the dimensionless temperature here is about 0.0125 (1.78 °F). This corresponds to 0.91% numerical uncertainty and is considered small compared to the

high temperatures of the domain. Even though the surface temperature solutions of the rod are less mesh independent than the flow pipe surface temperatures, the solution is still considered to be mesh independent. The solid continua surface temperatures have small numerical uncertainties because of the unstructured meshing techniques used for the models. The cells near the surfaces of the solid continua are constructed with a prism layer mesher which generates orthogonal prismatic cells next to the surfaces. The prism layer cells are smaller than the base cell size of the fluid domain and are thus more refined, even when the base size is its largest (2 mm). The mesh independence for the solid surface temperatures is also attributed to the fact that unstructured meshing in STAR-CCM+ creates smaller cells as the volume and thickness of the continua decreases. The rod and flow pipe thicknesses are thin which results in fine cells within the solid volumes. The fluid cell surface temperatures exposed to atmospheric convection as a function of distance from the gas plenum are shown in Figure 7.4.

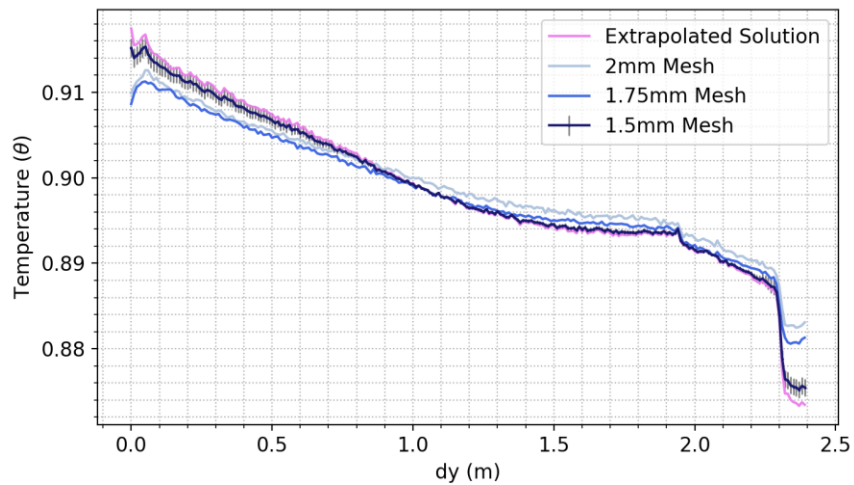


Figure 7.4. CFD temperatures near the atmospheric boundary with numerical uncertainty.

The maximum numerical uncertainty seen in this non dimensional temperature distribution is 0.012 (1.7 °F). While this uncertainty is higher than the previous surface temperature solutions the uncertainty is considered small for the application. The higher uncertainty of the atmospheric wall boundary is likely due to the higher heat transfer at the boundary. The difference in temperature between the solid continua and the fluid continua in the domain is more than a factor of 10 lower than the temperature difference between the fluid cells

near the atmospheric wall boundary in most locations of the domain. The higher heat transfer rates require more fine mesh resolution to improve the accuracy of the solution.

The fluid temperatures in the center of the riser and downcomer channels are less mesh independent than the surface temperatures because the base cell sizes are smaller than the prism layer cells in unstructured meshing. The fluid temperatures in the middle of the riser are shown in Figure 7.5 with respect to the distance from the gas plenum.

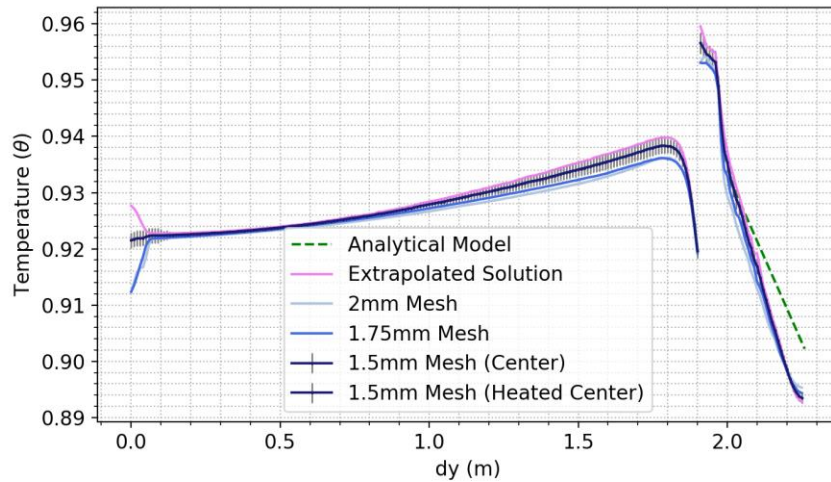


Figure 7.5. CFD riser centerline temperatures with numerical uncertainty.

The maximum uncertainty for the riser centerline temperatures is about 0.033 (4.7 °F). The uncertainty is about 2.4% with respect to the average fluid temperature. The highest uncertainties in the center of the riser channel are in the region where the flow area is higher, i.e., the unheated portion of the riser. As mentioned previously, with unstructured meshing the cells are more refined where the volumes and thicknesses of some continua are smaller. The numerical uncertainty in the reduced area region is lower because the riser channel thickness is smaller than in the unheated region. The same general trend occurs in the downcomer channel as illustrated by Figure 7.6.

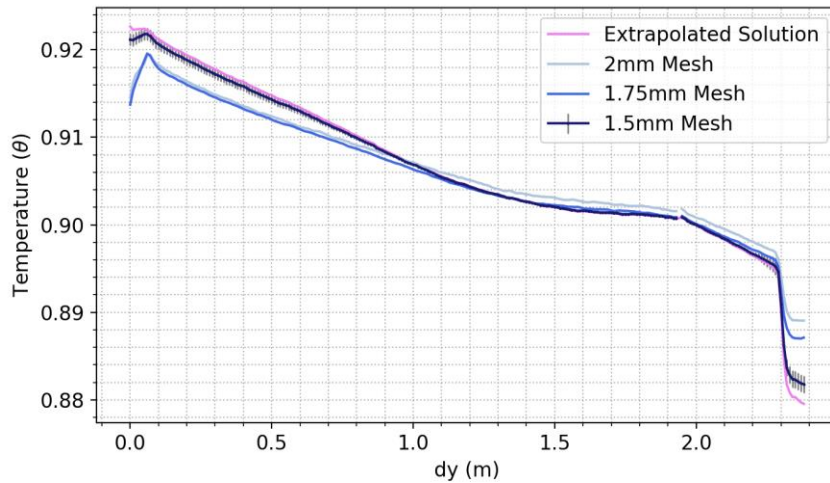


Figure 7.6. CFD downcomer centerline temperatures with numerical uncertainty.

The maximum uncertainty in the downcomer centerline temperature distribution is 0.025 (3.6 °F), corresponding to a percent uncertainty of 1.8%. Again, since the area is reduced in the lower axial location where the core heater exists in O-SERTTA, the unstructured mesh creates finer cells. The uncertainty in the shown temperature distributions is highest at the atmospheric wall boundary. The uncertainty peaks in the downcomer portion and slightly reduces as distance from the gas plenum increases. The unstructured meshing technique likely causes this decrease as well because the uncertainty becomes smaller when the downcomer area is reduced, resulting in finer prism layer cells and cells near the boundary.

The global flow rates were monitored with several cell monitors throughout the riser and downcomer channels of the simulation. The average flow rates were obtained by taking the mean of the monitors calculating local flow rates throughout the domain. The uncertainty of the flow is calculated similarly to the temperature distribution numerical uncertainties at the steady state status defined by Table 6.1. The time-dependent flow rates are shown in Figure 7.7.

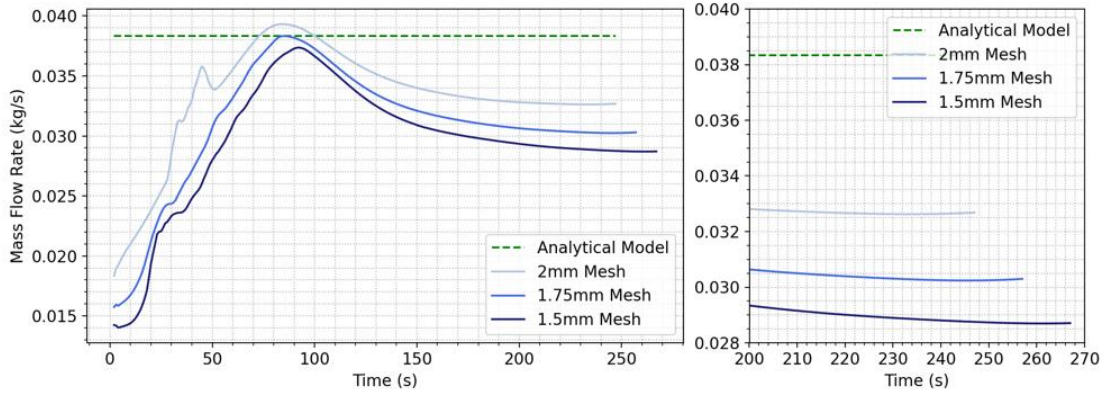


Figure 7.7. CFD mass flow rate with numerical uncertainty and analytical comparison.

The analytical model solution is shown for comparison. The flow rates have the highest numerical uncertainty in the developing phase of simulation. As the simulation progresses the uncertainties smooth as the flow rates begin to converge on the defined steady state operation. When the flow is said to reach steady state, the numerical uncertainty is approximately 0.003 kg/s (9.7%). This numerical flow uncertainty is said to be satisfactory for the application as further refinement of the mesh is expected to change a negligible amount compared to the validation comparison. The validation of the shown temperature and flow simulation solutions is discussed in the next section.

7.2 Validation

7.2.1 LUNA Fiber Optic Sensor Calibration

The fiber optic cables were set to capture measurements at a frequency of 2.25 Hz. All other instrumentation retrieved measurements at 2 Hz. Temperature estimations using the fiber optic sensors were obtained via an in-situ calibration correction using a closely located thermocouple. LUNA fiber optic sensors measure the change in strain along the length of the fiber. Since the aluminum sheathes are held still, it is assumed strain changes in the fiber are caused by thermal factors only. The location closest to a thermocouple on the fiber taped to the outer surface of the flow pipe was used for calibration. The measurements of the thermocouple and the nearby fiber optic sensor during transient procedures are shown in Figure 7.8.

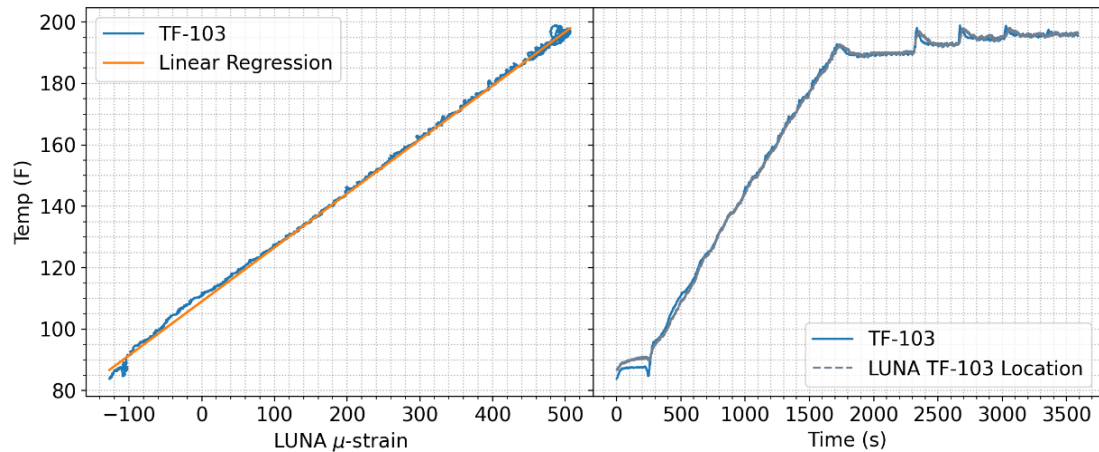


Figure 7.8. O-SERTTA LUNA fiber and TC-103 calibration plots.

The calibration curve was construed over the entire heat up procedure of O-SERTTA, steady-state operation, and pulse transient operation which can be seen on the right plot. The calibration curve shows good agreement and is deemed satisfactory for quantifying the temperature measurements with the LUNA fiber optic sensors based on strain. The fiber optic sensor measurements use the given calibration curve for the remaining presented results.

7.2.2 Pulse Operation

The pulse operation of O-SERTTA was done to quantify the flow rates in the system since steady state operational flow rates are immeasurable. Flow is confirmed by pulsing the heater rod and observing the LUNA fiber optic sensor and thermocouple responses. The response of the outer surface thermocouples is shown in Figure 7.9, where the top thermocouple is TC-103 and the bottom is TC-107. The thermocouple behavior during a 20 second long 4 kW pulse is shown. Note the heat map here is the simulation results shown for clarity on how the facility looks.

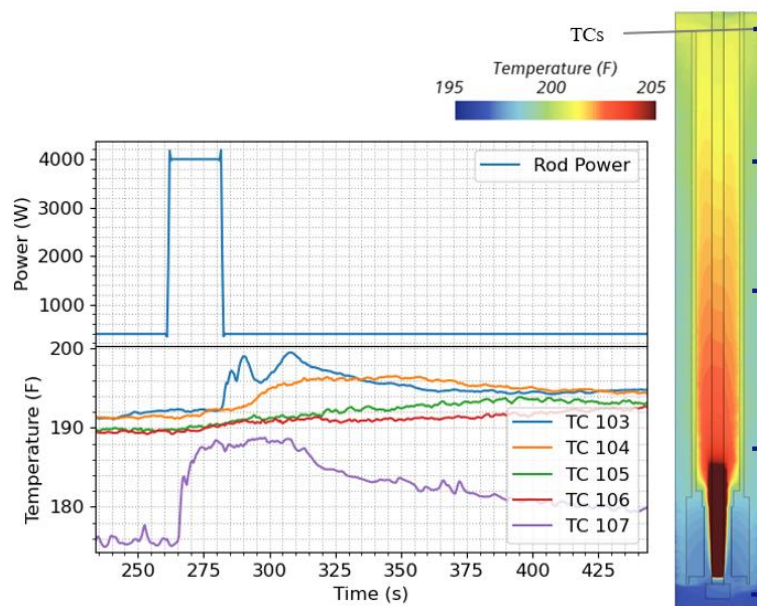


Figure 7.9. O-SERTTA close look at thermocouple response during pulse.

The bottom thermocouple (TC-107) responds the quickest after the power is increased. It is known that increasing the power of natural circulation systems increases flow rates. As the flow is accelerated, the warmer fluid in the downcomer channel relative to the thermally stratified region flows into the active region of TC-107. This explains why the initial spike in the thermocouple is seen before the remaining thermocouples. Approximately 15 seconds after the TC-107 temperature spike, the highest thermocouple TC-103 observes the thermal pulse that has traveled up the riser channel. After another 5 seconds, the downstream thermocouple (TC-104) measures an increase in temperature. By this point the thermal slug is not observable by the remaining two thermocouples because of conduction and thermal dissipation. The behavior seen here is indicative of the flow expected in the system.

The LUNA fiber optic sensor in the riser channel can capture the movement of the thermal slugs even better than the thermocouples. By monitoring the LUNA response along the entire length of the fiber throughout the pulse period, apparent thermal slugs can be observed. Examples of the fiber responses during 6 different pulses are shown in the 3D heat map plots in Figure 7.10. Using the center of heat (COH) detection method discussed earlier, the flow velocities can be estimated. The pulses are 20 seconds in duration and the

left column corresponds to 1 kW pulses, the middle column to 3 kW pulses, and the right column to 4 kW pulses.

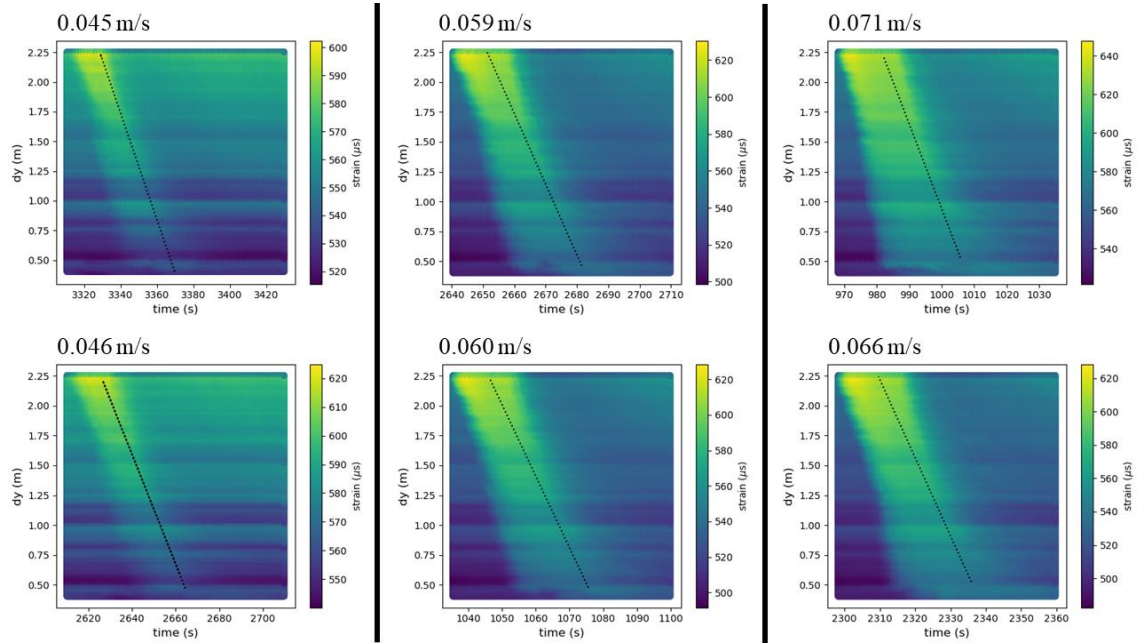


Figure 7.10. O-SERTTA fiber optic sensor heat map plots during power pulsing.

As can be seen from the estimated velocities, the higher-powered pulsing results in higher flow rate estimates, which is expected. The uncertainty in the velocity estimates is difficult to quantify because of the novelty of the center of heat method for in-situ applications. However, when comparing many different pulses of different durations and power levels displays a trend that could provide a reasonable estimate of the steady state flow rates of O-SERTTA. For 15 different pulses the estimated COH velocities are plotted as a function of the added energy above the steady state input energy shown in Figure 7.11. The added energy is calculated by the product of pulse duration and power increase. The results demonstrate higher velocities are seen when more energy is added to the system. Increasing power increases flow rates in NCLs and the analytical model suggests the following relation is true (Equation 4.16).

$$\dot{V} \propto (P)^{1/3} \quad 7.1$$

The analytical regression is based on this form and is shown in Figure 7.11.

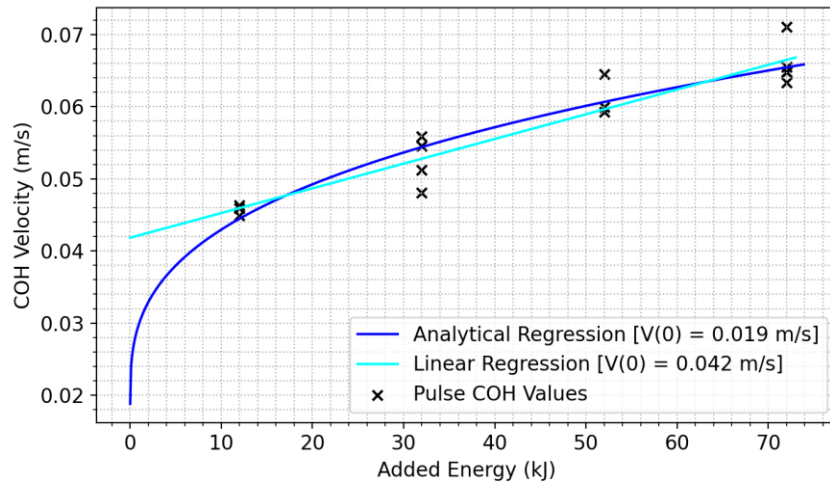


Figure 7.11. O-SERTTA center of heat pulse velocity estimate plot.

The relationship between flow rate and power is for steady state operation. Although the pulse operation of O-SERTTA is transient, the shape of the analytical regression line based on the analytical relation shows an initial spike seen at the initial stages of power increase as shown by the CFD simulations, Figure 7.7. The initial jump in flow rate is expected for sudden power increase because the fluid volume near the heated section is subject to a spike in temperature which rapidly reduces the density of the fluid near the heater. The heated portion of the O-SERTTA facility is not well instrumented so the transient temperature distribution of fluid in the heated section and flow rates are uncertain. Without an in depth understanding of the conditions during pulsing operation it is not certain that the steady state flow rate value would correspond to the analytical regression line at 0 added energy. However, the O-SERTTA steady state flow rate is likely to be bound by the 0 added energy velocity estimate and the linear regression. It is known that flow rates increase with increasing power in NCLs and thus the estimates with added energy would be an overestimate. This results in an experimental steady state estimate between 0.019 and 0.042 m/s.

7.2.3 Pseudo Steady State

To quantify the validation error, the experimental and computational quantities were compared for the steady states corresponding to Table 6.1. Both the experimental operation

and simulation are not true steady state but rather a pseudo steady state. The measurements from the thermocouples on the wall surface exposed to the atmosphere are compared to the simulation temperature distributions at the same spatial location in the simulation. The simulation values are volume averaged over the section where the experimental thermocouples respective to the volume of the thermocouple. As stated earlier the simulation results are averaged around the circumference of the simulation, whereas the experimental measurements are circumferential location. The experimental and computational temperature distributions in the downcomer channel during pseudo steady state are shown in Figure 7.12. A heat map of the simulation distribution is also shown in relation to the axial locations plotted on the graph.

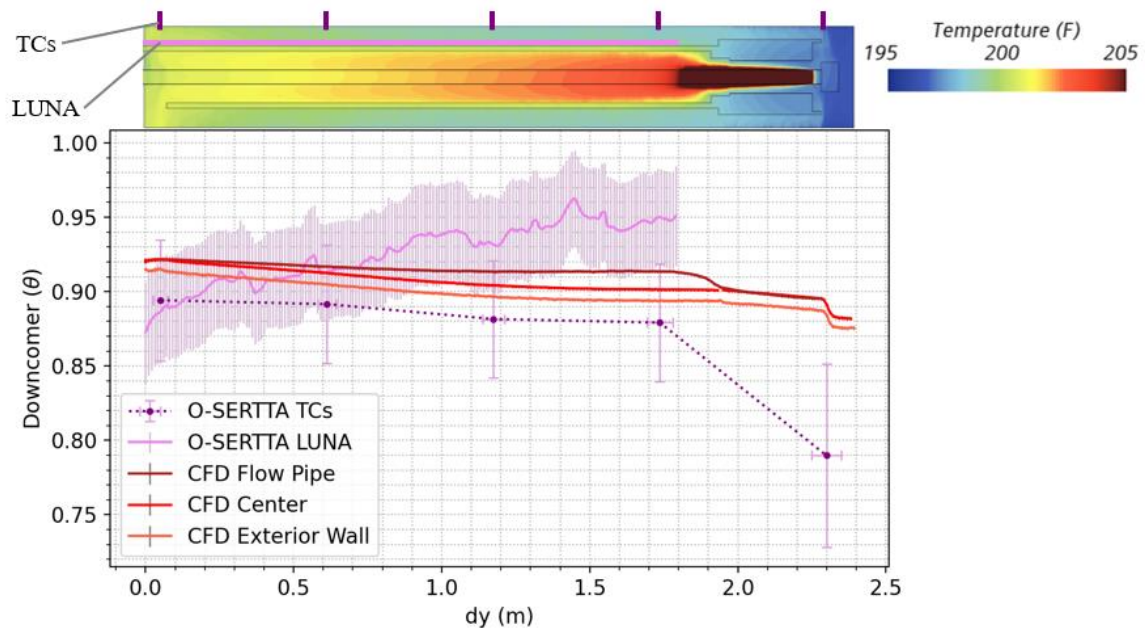


Figure 7.12. CFD and O-SERTTA downcomer temperature distributions with CFD heat map.

The temperature differential as the distance from the gas plenum increases is smaller as predicted by the CFD model compared to experimental results. In particular, the last thermocouple, which is in the thermally stratified region near the bottom of the downcomer section, measures a much lower temperature than predicted by the CFD model. The reason this is seen is because of initial starting temperatures. The CFD model initial temperature is never as low as the pseudo steady state value measured by the thermocouple furthest

from the gas plenum. The validation comparison for the wall thermocouples is shown in Figure 7.13, along with the experimental uncertainty of the thermocouple measurements.

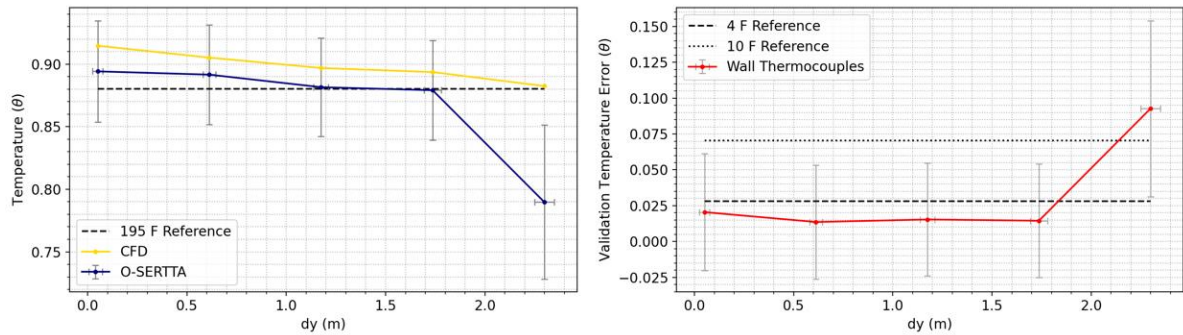


Figure 7.13. Outer wall thermocouple validation error and uncertainty.

As shown by the validation error, aside from the thermocouple in the thermally stratified region, the error is less than 5 °F. Although the last thermocouple has an error above 10 °F, the general trend of slightly decreasing temperatures as distance from the gas plenum increases is seen in both the experimental measurements and CFD results. The validation error for the LUNA fiber optic sensor fixed to the flow pipe surface in the downcomer is also low, which can be seen in Figure 7.14.

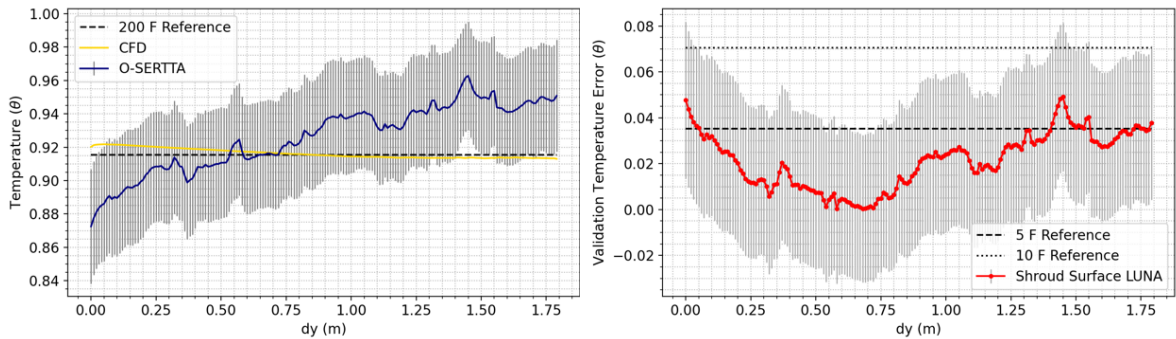


Figure 7.14. LUNA fiber validation error and uncertainty for downcomer flow pipe surface.

The experimental uncertainty of the fiber optic sensor is shown with the comparison with the CFD results. While the experimental measurements demonstrate a slight increase in temperature as dy increases, the CFD results predict a relatively flat profile. The experimental uncertainty may illustrate the fact that the measured increasing temperature

could be a result of the uncertainty. Despite the disagreeing trend, the validation error is still in good agreement with errors around 5 °F.

The riser temperature profiles have good agreement between the CFD and experimental results. The results are shown in a similar manner to the downcomer temperature profile results in Figure 7.15. The theoretical estimate as calculated by the analytical model is also shown in relation to the CFD model prediction for the change in fluid temperature across the heat source.

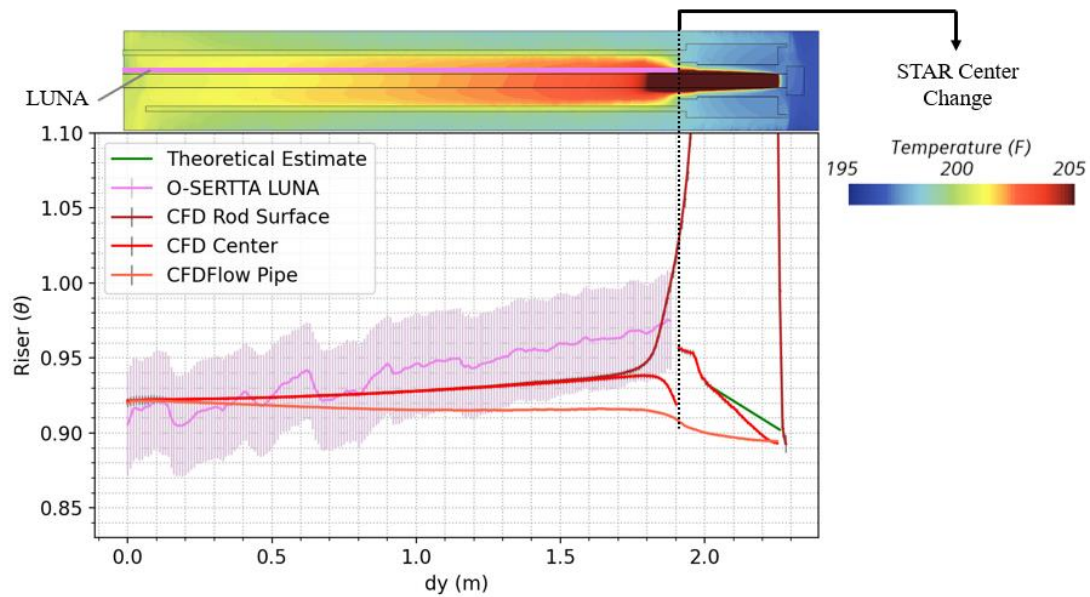


Figure 7.15. CFD and O-SERTTA riser temperature distributions with CFD heat map.

The only fluid temperatures monitored in the riser of the experiment is with the LUNA fiber optic sensor attached to the heater rod. When comparing experimental to the CFD results, similar trends are seen as shown by the validation comparison in Figure 7.16.

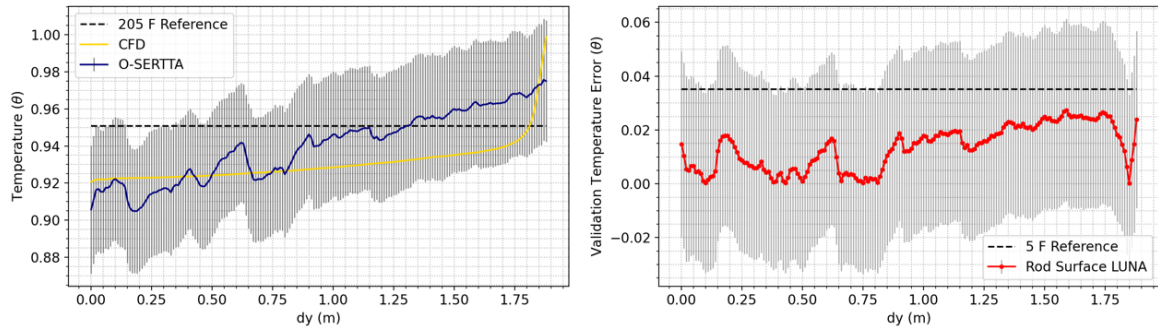


Figure 7.16. LUNA fiber optic sensor validation error and uncertainty in the riser.

The temperature near the rod surface is shown to increase as dy increases according to both experimental and computational results. Near the end of the fiber optic sensor length, divergence begins to occur as the CFD model measurements come closer to the heated section of the heater rod. The CFD model overpredicts the rod surface temperatures when compared to the experimental measurements which can be seen in the rod surface thermocouple comparison in Figure 7.17.

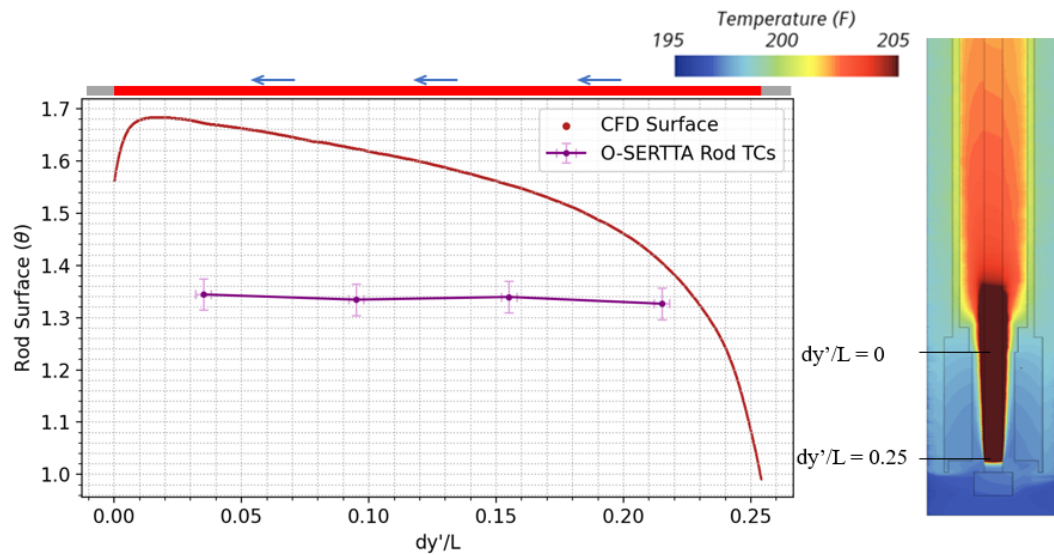


Figure 7.17. CFD and O-SERTTA rod surface temperature distributions with CFD heat map.

The simulation and experimental results both show rod surface temperatures above 1.3 (254 °F) which is above water saturation temperature. The experimental thermocouples are separated from the fluid by a thin flow pipe. As such the exact surface temperature would

only be a few degrees lower and it would be expected that some bubble formation would be occurring on the rod surface. The simulation results show higher temperatures with a peak temperature of about 1.7 (311 °F). The validation error for the thermocouples becomes more substantial around the peak temperatures predicted by the CFD model, which can be seen in Figure 7.18.

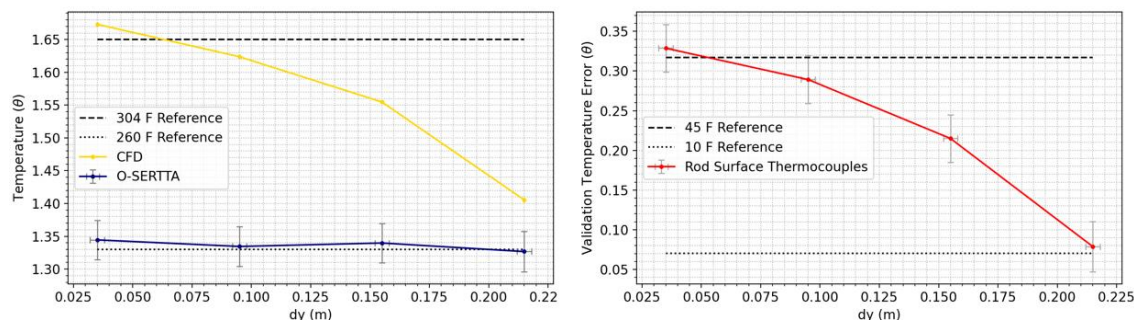


Figure 7.18. Rod surface thermocouple validation error and uncertainty.

The validation error ranges from 0.38 to 0.08 (50 °F to 10 °F). With respect to the average fluid temperatures, the rod validation error peaks at 29% error. The heat transfer coefficient of the model is underestimated because of single phase water modeling. There may be some boiling in the experiment. It is known from literature review section that the heat transfer coefficient increases in the initial stages of bubble formation, which can be seen in Figure 7.19.

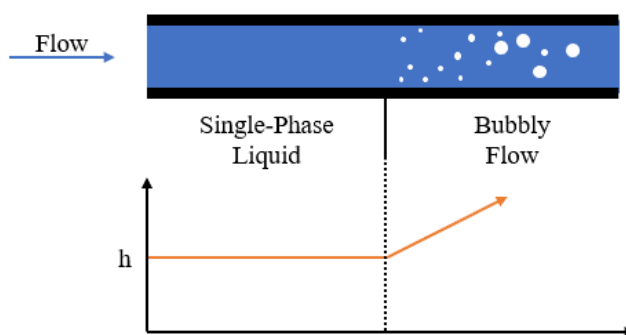


Figure 7.19. Heat transfer coefficient depending on flow regime.

The CFD model applied a heat flux at the rod surface to match the experimental heat flux. Since the single-phase model would not predict any bubble formation, the temperature difference between the fluid and the rod surface becomes higher since the heat transfer

coefficient is underpredicted. While this disagreement causes higher validation error, the bulk flow characteristics appear to be only dependent on the heat flux rather than the heat transfer coefficient at the heated surface of the rod.

The velocities obtained from the simulation are lower than the analytical model solution. There are many assumptions made with the analytical model that could be the cause of the overpredicted velocity values. The main discrepancy is likely to be the fact the analytical model is one dimensional and does not account for conduction across the flow pipe. As stated in the literature review, natural circulation has been observed to be a 3D phenomenon. The estimated velocity of O-SERTTA based on the three different methods are shown in Figure 7.20. The O-SERTTA COH value corresponds to the method using the center of heat detection in the experiment.

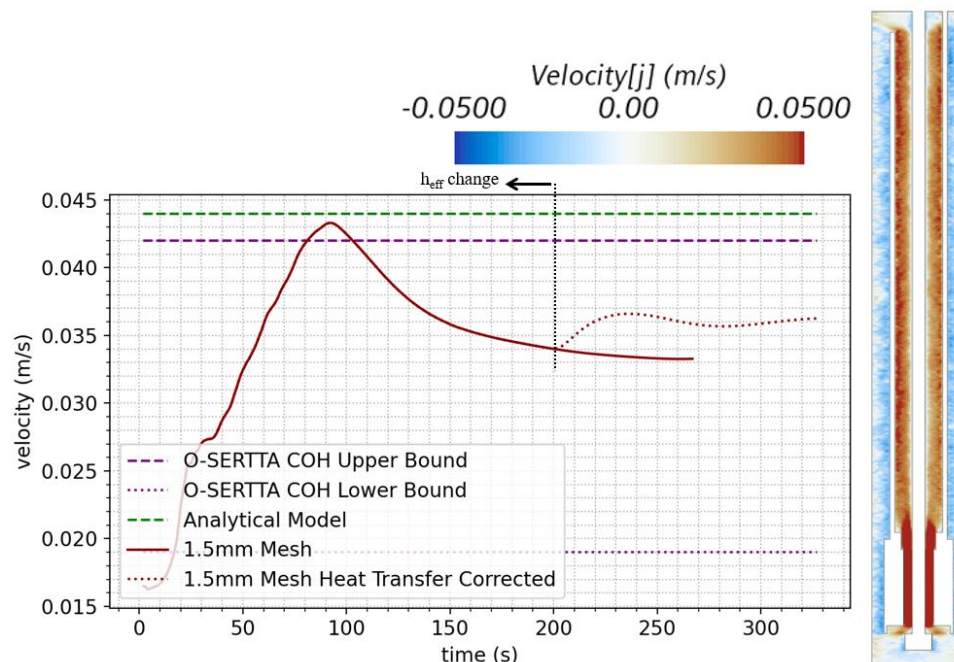


Figure 7.20. CFD and O-SERTTA velocity estimates with CFD velocity contour map.

The CFD predicted velocities after the heat transfer coefficient is corrected align with the experimental upper bound prediction with an error of only about 0.009 m/s. However, as shown by the GCI study, the extrapolated velocity of the simulation will be lower than the mesh shown here. Thus, the CFD velocity here serves as an upper bound result of the CFD simulations. The validation error has high uncertainty because of the experimental velocity

estimation methodology. Center of heat detection using fiber optic sensors is new and limited data are available for uncertainties in annular flow channels.

The CFD model demonstrates the code's ability to characterize the behavior of the O-SERTTA system and presumably other internally heated concentric tube thermosyphon designs. Introducing systematic alterations in the model produces expected results. The heat transfer coefficient to the atmosphere was increased at 200 seconds of simulation time and a noticeable increase in the predicted velocity occurs. This is expected behavior because more heat removal results in a higher temperature differential which drives flow in the loop. The system is said to be steady state for the last 190 seconds of simulation time which is approximately the time it takes for one flow cycle. The verification and validation of the O-SERTTA model justifies the ability of the model to predict how parametric changes would affect flow rates. Note that this model discussed in later sections follows the naming conventions in Table 6.4.

7.3 Sensitivity Study

The parametric study was conducted to provide a better understanding of how different geometrical parameters, initial conditions, and modeling methods affect the solution results. During construction of the O-SERTTA facility, it was uncertain what radial and axial parameters of the flow pipe above the heated section should be chosen to optimize flow rates. As discussed in the literature review, the conduction between the downcomer and riser creates a dependency that does not exist in other flow loops. The parametric study will help identify what geometry optimizes flow rates to inform similar designs relevant for nuclear applications.

Much of the literature suggests that natural circulation systems often have more than one steady state, or no steady state. While the CFD model predicts very small oscillatory behaviors during operation, per the criteria set in Table 6.1 the models still achieve a pseudo steady state. The steady state temperature distributions and flow rates are compared between models with different initial conditions or modeling parameters. It was discussed

in the theory section that there are two correction factors that can be used with the realizable k - ε turbulence model. The Xu correction factor is generally recommended for buoyancy driven flow, however, the Wolfstein correction was used for validation since has been shown to be accurate for internal flows [96]. The differences in the predicted results are shown for the two models. Results demonstrating the sensitivity of the simulation solution on the initial heat transfer coefficient are also shown.

7.3.1 Modeling Parameters

The flow results for the remaining plots are given as the Reynolds (Re) number in the heated region of the geometry. This is because during the geometrical changes, the unheated region axial or radial parameters are modified but the heated region remains the same. Additionally, the Re number is of concern since the goal is to maximize the flow near the test section. The Re numbers as a function of time for the four different O-SERTTA geometry simulations and the simplified geometry simulation are shown in Figure 7.21. The naming conventions are described in Table 6.4.

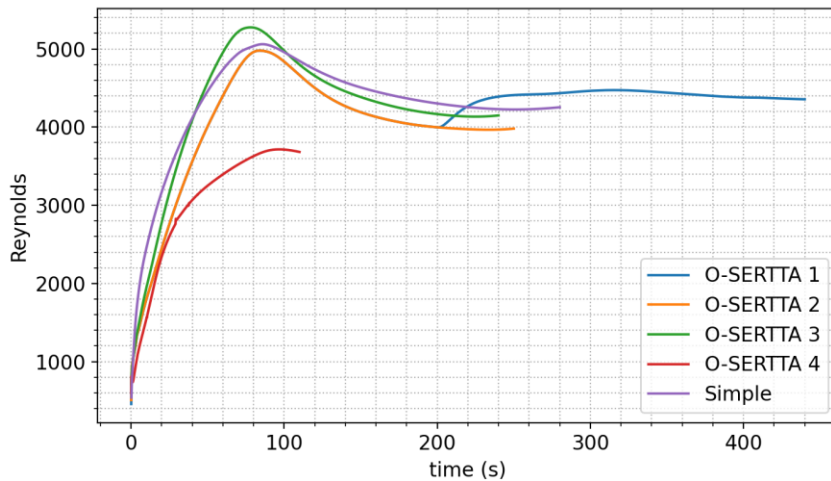


Figure 7.21. Reynolds number of various O-SERTTA models and the simplified geometry.

The first notable difference is the lower predicted flow rates using the Xu correction in the turbulence model. The Xu correction (O-SERTTA 4) underpredicts flow by 15.5% compared to the validated solution using the Wolfstein correction (O-SERTTA 1). The Xu

correction also converged slower which is why the solution was only extrapolated to just over 100 seconds. The initial condition dependence on the heat transfer coefficient influences the flow rates by about 5% (O-SERTTA 1 compared to O-SERTTA 3). However, it appears the solutions would have a smaller difference if time was extrapolated longer. This can be inferred by the slight changes being seen after steady state is reached. The O-SERTTA simulations with the higher heat transfer coefficients appear to be converging to values within the numerical uncertainties shown in Figure 7.7 and Figure 7.20. The simplified geometry model is observed to be slightly higher than its O-SERTTA model equivalent. This could be because of reduced frictional losses from the O-SERTTA azimuthally symmetric components.

The temperature distributions of these models give some insight on how the different modeling parameters affect the resultant profiles. The gas plenum is modeled as a pressure outlet with values controlling the pressure and temperature of backflow conditions. Although the flow at the pressure outlet in the simulation is limited, the backflow temperature influences the overall temperature distributions. To ensure that these conditions do not affect the flow rates the pressure outlet backflow temperature was controlled with two different methods: a constant temperature above the average fluid temperature (only O-SERTTA 2) and an average temperature of the volume near the pressure outlet to minimize heat transfer. The centerline temperatures for the O-SERTTA runs and the simplified run are shown in Figure 7.22.

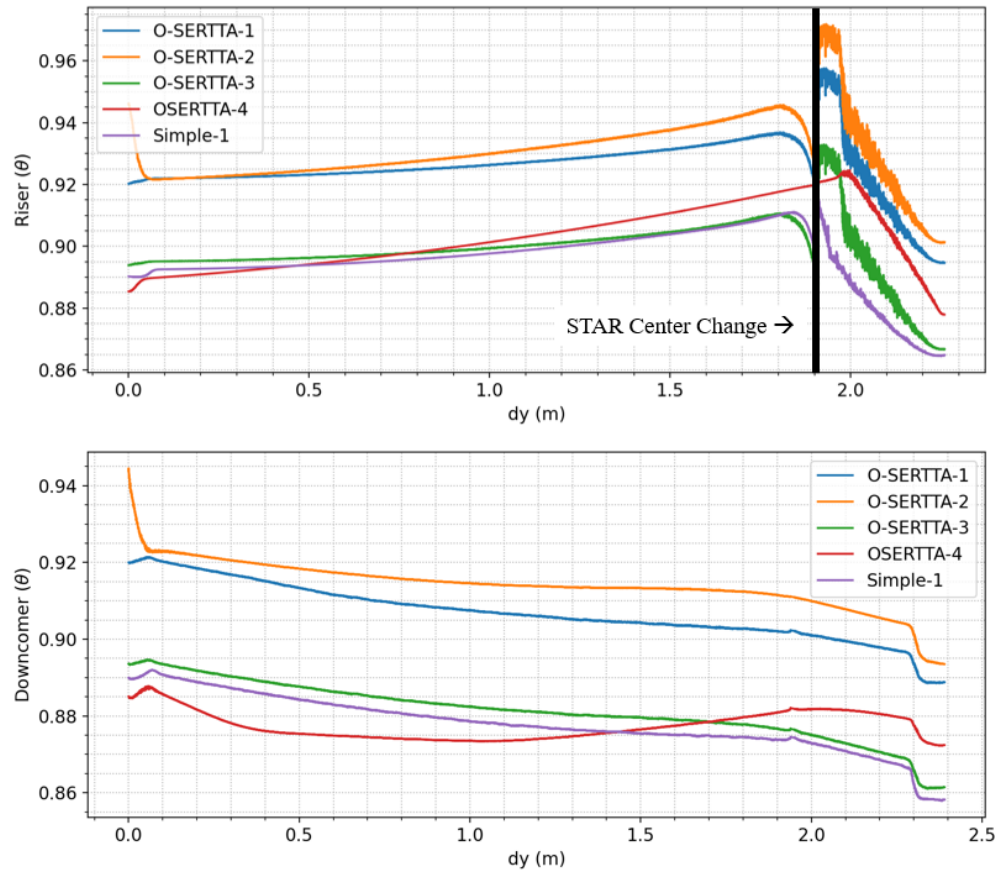


Figure 7.22. Centerline temperatures of various O-SERTTA models and the simplified geometry.

The same general trend in the riser channel for the different simulations can be seen in the first temperature plot. There is a slight increase in temperature as you get closer to the heated portion where there is a drop in temperature across the heated portion. The main differences shown in the centerline plots are again with the Xu correction (O-SERTTA 4). The Xu correction predicts the temperature near the heated portion where the STAR centerline monitor changes is more uniform. This is expected because the flow rates are lower which allows more time for mixing as flow moves around the loop. The other O-SERTTA models and the simplified model have the same trends and only slight discrepancies in the magnitudes. O-SERTTA 1 and 2 are bound to have higher temperature profiles because of the heat up period in the first 200 seconds when the heat transfer coefficient is lower.

The downcomer centerline temperatures show similar characteristics to the riser temperatures where all the models aside from the Xu correction model have the same trend. With slower flow rates, there is more time allowed for heat transfer across the flow pipe into the downcomer. This can explain why there is a slight increase in temperature seen by the O-SERTTA 4 model as distance from the plenum increases. One discrepancy that exists between the other models is the increase in temperature near the gas plenum shown by O-SERTTA 2. This was because the pressure outlet boundary condition backflow temperature was set higher than the other models. As shown in the Reynolds number plots and the remaining temperature distributions, this does not appear to have a significant effect on the remainder of the CFD solution compared to the other O-SERTTA models.

7.3.2 Geometrical

The first parametrization was done for the radius of the flow pipe above the heated portion of the test section. The velocities based on how the flow pipe radius was modified is shown in Figure 7.23.

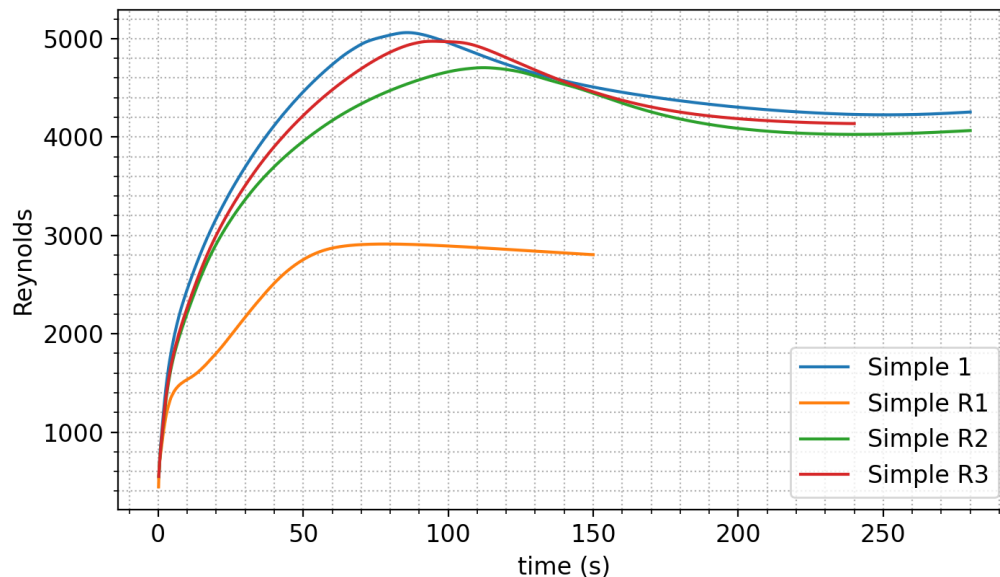


Figure 7.23. Reynolds dependence on radius for simplified CFD model.

The reduction in velocity when the radius is at a minimum (R1) is expected. It was predicted the velocity would increase as the flow pipe radius decreases until the frictional

forces from the reduced area channel outweighed the benefits of the reduced body forces from reduced volume. Aside from the minimum radius, the other radial perturbations are not apparently better than the others. Literature would suggest that the area of the downcomer channel should be equal to the area of the riser channel to optimize flow rates. This is not true for the CFD simulation as the peak velocities are seen when the downcomer channel area is higher than the riser. This could be because when the riser flow area is reduced, less mass of fluid is resisting the flow.

The results can be compared with the analytical model to show similar predictions. The comparison between CFD and analytical solution is in Figure 7.24

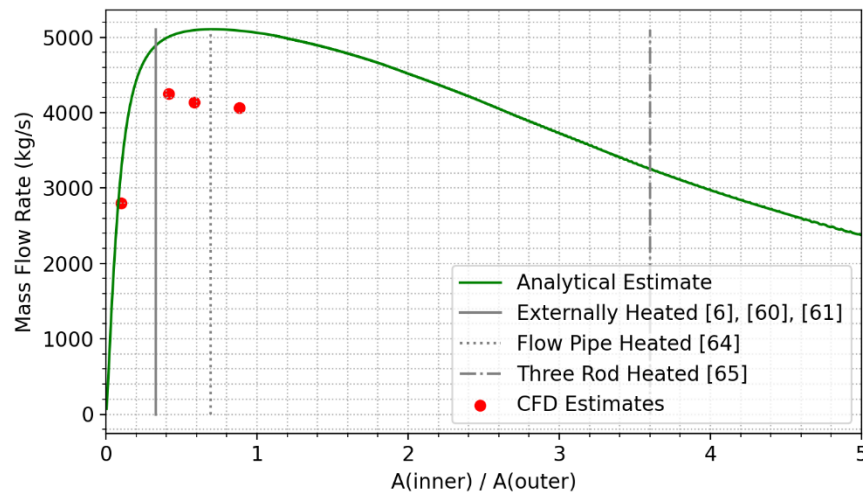


Figure 7.24. Reynolds number based on flow pipe diameter for analytical model, experimental facilities, and CFD models.

The CFD models do not predict significantly different flow rates in the three CFD models with values of the ratio inner area to outer area between 0.4 and 1. This region on the analytical model prediction curve is also where little influence is expected to occur. When the ratio is lowered near 0.1 the CFD simulation results agree well with the analytical model prediction. The results from CFD are limited but the agreement with the analytical model suggests there is a range of this ratio where little impact on flow rates occur based on the ratio. Outside of that range frictional forces are suggested to negatively impact flow rates.

7.3.3 Flow Pipe Height Dependence

The last geometrical modification to be made is with the flow pipe height. If the height of the flow tube could be lowered, there could be much more room to accommodate instrumentation. The flow pipe height was reduced by 27%. The resulting Reynolds number compared to the non-modified simple geometry is shown in Figure 7.25.

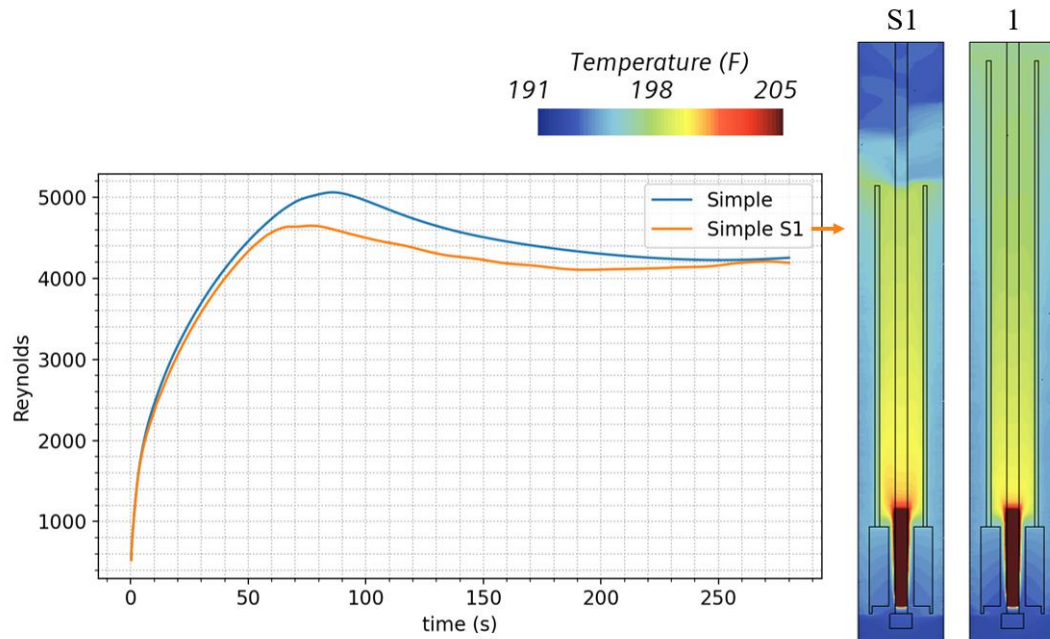


Figure 7.25. Flow pipe height impact on CFD flow rate.

Based on the analytical model for flow, lowering the flow pipe height should not impact flow rates since all the assumptions still hold true. The analytical model agrees with the CFD results in that the flow rates are relatively maintained when a significant decrease in flow pipe height is applied. This information has a lot of value because one of the main concerns about this design is the small spatial constraints. Since flow rates are not reduced as the flow pipe height is lowered, more space is created for the accommodation of instrumentation.

Variations in geometry impact flow rates, in some cases, substantially. The geometry variations did not lead to any flow instabilities. The same general temperature trends are seen for the different variations which can be seen in Figure 7.26.

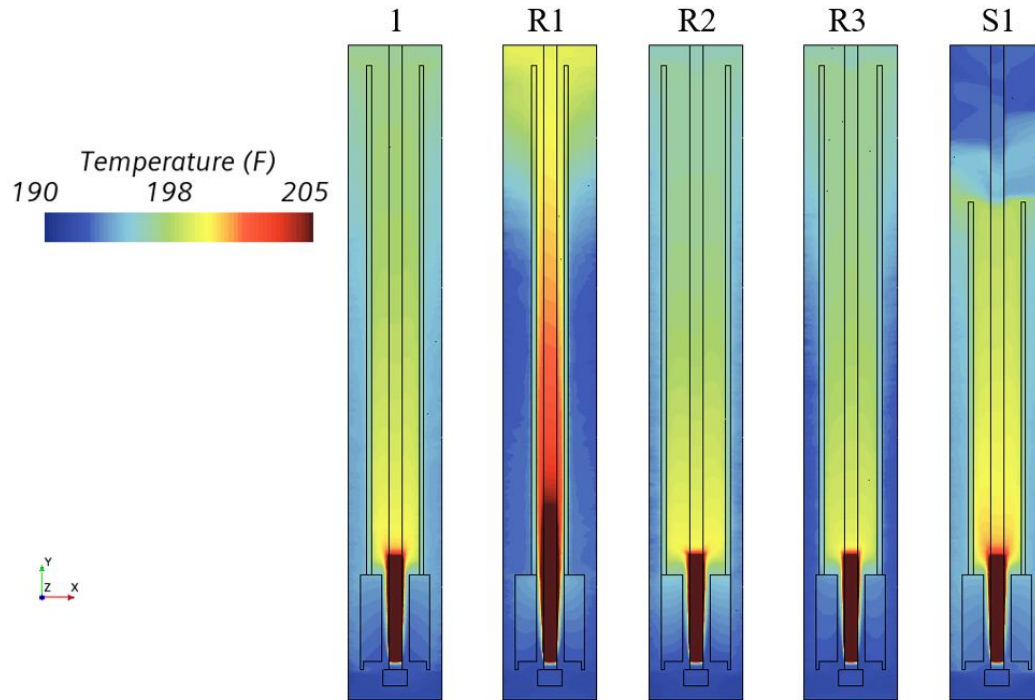


Figure 7.26. Various simplified CFD geometry temperature profiles.

The predicted stability of the design is valuable for design because the radial and axial flow pipe parameters are not super sensitive aside from the conditions discussed. Aspect ratios are used in the literature to describe the geometries of tall natural circulation loops. The following is a table of some aspect ratios and the respective mass flow rates. R_t and D_t are the riser and downcomer thicknesses, respectively. The “a” subscript denotes the respective areas. L is the length and MFR is the mass flow rates.

Table 7.2. Aspect ratios and respective mass flow rates for simplified CFD models.

Name	R_t/D_t	R_a/D_a	L/R_t	L/D_t	Re
1	0.95808	0.384551	146.1725	140.045	4253
R1	0.253649	0.078109	353.4923	89.66295	2803
R2	1.895752	0.842831	109.2489	207.1088	4065
R3	1.303573	0.550367	126.3874	164.7552	4136
S1	0.95808	0.384551	107.0366	102.5497	4193

The aspect ratios should be the first thing to think about when designing a concentric tube thermosyphon because it gives insight into how thin the channels need to be respective to each other and the length of the system.

8 CONCLUSION

This research has demonstrated STAR-CCM+ can adequately analyze the thermal hydraulic performance of an internally heated concentric tube thermosyphon design. The accuracy was quantitatively described using V&V procedures. The validation errors and uncertainties of the flow rates and temperature distributions were determined acceptable for the proposed purpose. High fidelity thermal hydraulic data were provided for a concentric tube thermosyphon design at atmospheric pressure and a power of 400 W. The O-SERTTA experiment is estimated to be able to achieve a Reynolds number of 4400 near the test specimen with the associated temperatures, pressure, and power described. The flow rate validation error is about 12% when comparing with experimental results. However, it must be noted that the experimental flow rate uncertainty is likely to be high. Future efforts using fiber optic sensors to measure thermal slug velocities should focus on an uncertainty quantification methodology. Currently there is no clear-cut way to determine the experimental uncertainty in these regards, which is a limitation to this study.

The parametric study gave insight into several design considerations that are expected to improve the performance of internally heated concentric tube thermosyphons. The main considerations are shown in Figure 8.1. The leftmost schematic is the arrangement of O-SERTTA. The fiber optic sensor was fed through small gauge tubing to limit any resistance larger diameter tubing would cause. This limited the depth of the sensor into the test section. If a larger sheath was inserted into the test section, more of the lead cable to the sensor could have been put into the riser as shown on the right. Cooling coils were not put into the downcomer because of similar concerns with flow resistance. If the cooling coils were arranged as shown in the middle, the effective heat transfer coefficient would improve as compared to having the coils on the exterior. Improving heat removal would lead to performance improvements.

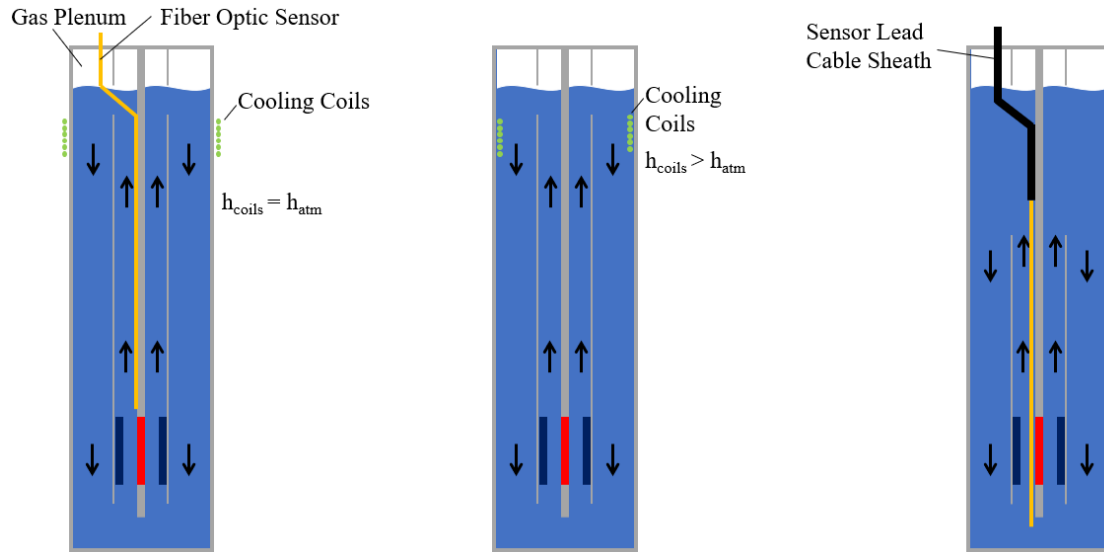


Figure 8.1. Internally heated concentric tube thermosyphon design improvement considerations.

The radius of the flow pipe that is above the heated portion is not predicted to have a high impact on flow rates until flow restriction becomes dominant. The O-SERTTA riser area to downcomer area ratio (R_A/D_A) is 0.38. It was shown that increasing R_A/D_A to 0.55 and 0.84 only reduced the resultant flow rates by 3.8% and 5.4%, respectively. The flow rates decreased by about 36% when R_A/D_A was set to 0.078. Based on these results it's conclusive that R_A/D_A should not be much smaller than 0.38 because at some point decreasing adds too much friction. Inference from this result would mean R_A/D_A should not be too large either since it is expected that flow restriction would occur in the downcomer channel as well. Past research on concentric tube thermosyphons has pointed out that R_A/D_A should be near 1 to optimize flow rates. It appears that is not the case for an internally heated design with annular flow paths as it can be seen the flow rates decrease when R_A/D_A is increased from 0.38 to 0.84. Although it is difficult to estimate the optimal R_A/D_A , conclusions can be made about a general range. When R_A/D_A is between 0.38 and 0.84 the flow rates all fall within 6% difference.

Based on the flow pipe radial parametric study, a design consideration could be implemented which would increase heat removal substantially without negatively impacting flow rates. Currently the O-SERTTA heat sink is the atmosphere on the outer

surface of the facility with coils wrapped around the external surface. The outer surface coils provide the same heat removal as the atmosphere. Small gauge tubing could be fed from the pipe cap to helically wrap around in the downcomer channel near the turnaround region. Flowing water or air through the tubing would provide a much better heat sink than what is currently implemented in the O-SERTTA design. The drawback to this would be increased flow resistance in the downcomer because of a smaller cross-sectional flow area. It was shown in the radial parametric study that reducing the downcomer flow area by 27% only lowered the flow rates by 5.4%. Although the riser area increased when the downcomer flow area was decreased, it can be assumed that the tubing could fill 27% of the downcomer cross-sectional flow area while maintaining flow rates that are decreased by only 5.4%. When the fact that coolant will be flowing through the tubes is incorporated into the analysis, the buoyancy buildup is expected to increase substantially which is the driver of flow.

Lowering the flow pipe height by 27% decreased the flow rates in the test section by only 1.4%. When the flow pipe was lowered the fluid above the turnaround section mixes. In this mixing region a high gauge sheath could be fed through without impacting the flow because it would be tethered to the top of the tank outside of the riser. The large diameter sheath would allow larger/more instrumentation for temperature, pressure, or other measurements. This is of value if fiber optic sensors are used because the diameters of the lead cables to the fiber optic sensor adapter are more than 2 orders of magnitude larger than the diameters of the sensors themselves. Therefore, in smaller flow areas the cables would impact flow so they cannot be put in the flow path. When the flow pipe is lowered higher diameter sheaths can be used to feed instrumentation components, allowing instruments like fiber optic sensors to be located deeper into the fluid test section to provide data information.

Future performance data should be provided for LWR technology conditions. With higher temperatures and pressures, power can be higher which will increase flow rates. This research has demonstrated internally concentric tube thermosyphons provide consistent flow, but LWR conditions are preferred for irradiation experiments. Modeling helical heat

exchanger coils in the downcomer in the future will be useful to analyze the improved performance compared to no internal coils. The modeling would benefit from multiphase approaches if there were bubble formations occurring in the experiment. The heater rod was modeled by stainless steel but, the rod composition is much more complex. Further modeling of the rod could also better address the large validation error associated with the rod surface temperature.

BIBLIOGRAPHY

- [1] D. C. Crawford, D. L. Porter, S. L. Hayes, M. K. Meyer, D. A. Petti and K. Pasamehmetoglu, "An approach to fuel development and qualification," *Journal of Nuclear Materials*, vol. 371, pp. 232-242, 2007.
- [2] C. M. Petrie, D. C. Sweeney, R. H. Howard, D. K. Felde and J. L. McDuffee, "Single-phase, natural circulation annular flow measurements for cartridge loop irradiation experiments," *Nuclear Engineering and Design*, vol. 370, no. 110900, pp. 1-15, 2020.
- [3] J. McDuffee and K. Thoms, "A review of molten salt irradiation experiments.," *Trans. Am. Nucl. Soc.*, pp. 313-316, 2019.
- [4] C. Jensen, D. Wachs, J. Carmack and N. Woolstenhulme, "Review of Transient Testing of Fast Reactor Fuels in the Transient REactor Test Facility (TREAT)," Idaho National Laboratory (INL), Idaho Falls, IAEA-CN-245-523.
- [5] V. N. Efimov, I. Zhemkov, A. N. Kozolup, V. I. Polyakov, V. T. Stepanov, Y. E. Synda, V. V. Orlov, A. I. Filin, A. G. Sila-Novitsky and A. A. Pikalov, "The vor-60 loop-channel design for testing the brest reactor fuel.," IAEA-TECDOC-1348, International Atomic Energy Agency, 2003.
- [6] R. H. Chen, G. H. Su, S. Z. Qiu and K. Fukuda, "Prediction of CHF in a concentric-tube open thermosiphon using artificial neural network and genetic algorithm," *Heat Mass Transfer*, vol. 46, pp. 345-353, 2010.
- [7] M. A. Islam, M. Monde and Y. Mitsutake, "CHF characteristics and correlations of concentric-tube open thermosyphon working with R22," *International Journal of Heat and Mass Transfer*, vol. 48, pp. 4615-4622, 2005.
- [8] S. E. Bays, G. J. Youinou, M. Lillo and P. Gilbreath, "ATR Compendium: Irradiation Test Capabilities," *Nuclear Technology*, vol. 201, no. 3, pp. 191-208, 2018.
- [9] J. McDuffee, N. Cetiner, N. D. Bull Ezell, L. Qualls and K. Thoms, "Evaluation of Flowing Salt Irradiation Facilities with High Neutron Flux," Oak Ridge National Laboratory, ORNL/TM-2018/954, 2018.

- [10] N. Woolstenhulme, C. Baker, C. Jensen, D. Chapman, D. Imholte, N. Oldham, C. Hill and S. Snow , "Development of Irradiation Test Devices for Transient Testing," *Nuclear Technology*, vol. 205, no. 10, pp. 1251-1265, 2019.
- [11] D. Wachs, K. Weaver, M. Farmer, J. McDuffee, P. Sabharwal, S. Malloy, D. Rao, D. Wooten, T. Hill and S. McDeavitt, "Development of experimental capabilities for fuels and materials testing in the versatile test reactor," *Transactions of the American Nuclear Society*, vol. 119, no. 1, pp. 507-508, 2018.
- [12] J. L. McDuffee, D. K. Felde and J. J. Carbajo, "Design and Testing for a New Thermosyphon Irradiation Vehicle," Oak Ridge National Laboratory, ORNL/TM-2017/399, 2017.
- [13] C. L. Pope, C. B. Jensen, D. M. Gerstner and J. R. Parry, "Transiant Reactor Test (TREAT) Facility Design and Experiment Capability," *Nuclear Technology*, pp. 1-10, 2019.
- [14] N. Woolstenhulme, "Status Report on Development of MARCH Module Designs," Idaho National Laboratory, INL/EXT-19-55844, 2019.
- [15] D. W. Kamerman, "Loss of Coolant Accident Commissioning Test Plan," Idaho National Laboratory, INL/MIS-19-55237, 2019.
- [16] N. Woolstenhulme, C. Jensen, C. Folsom, R. Armstrong, J. Yoo and D. Wachs, "Thermal-Hydraulic and Engineering Evaluations of New LOCA Testing Methods in TREAT," *Nuclear Technology*, vol. 207, pp. 637-652, 2021.
- [17] D. B. Trauger and J. A. Conlin Jr. , "Circulating Fused-Salt Fuel Irradiation Test Loop," *Nuclear Science and Engineering*, vol. 9, pp. 346-356, 1961.
- [18] G. W. Keilholtz, J. G. Morgan and W. E. Browning, "Effect of Radiation on Corrosion of Structural Materials by Molten Fluorides," *Nuclear Science and Engineering*, vol. 5, pp. 15-20, 1959.
- [19] D. B. Trauger, "Some Major Fuel-Irradiation Test Facilities of the Oak Ridge National Laboratory," ORNL-3574, 1964 .
- [20] M. Surendran, "Computational Fluid Dynamic Modeling of Natural Convection in Vertically Heated Rods," Utah State University , 2016.

- [21] A. Kraus and R. Hu, "CFD Simulation of Natural Convection Cooling After a Loss-of-Flow Transient," Argonne National Laboratory, Chicago, 2015.
- [22] R. M. Fanning, "Design and Analysis of a Passive Heat Removal System for a Small Modular Reactor Using STAR CCM+," Missouri University of Science and Technology, 2017.
- [23] Y. Zhang, P. Gao, C. Tian, X. He, C. Chen and Z. Wang, "Coupling one-dimensional user code and STAR-CCM+ for simulation of rolling conditions in a rod bundle channel," *Annals of Nuclear Energy*, vol. 151, no. 107888, 2021.
- [24] M. Jeltsov, D. Grishchenko and P. Kudinov, "Validation of Star-CCM+ for liquid metal thermal-hydraulics using TALL-3D experiment," *Nuclear Engineering and Design*, vol. 341, pp. 306-325, 2019.
- [25] Y. Lin, P. Gao, X. Chen, S. Bello, C. Tian and C. Zhao, "Experimental investigation on instability characteristics of loss of heat sink accident in a natural circulation system," *Annals of Nuclear Energy*, vol. 155, pp. 1-8, 2021.
- [26] P. K. Vijayan, A. K. Nayak and N. Kumar, Single-Phase, Two-Phase and Supercritical Natural Circulation Systems, Cambridge: Elsevier Ltd. , 2019.
- [27] E. Ramos, M. Sen and C. Trevino, "A steady-state analysis for variable area one- and two-phase thermosyphon loops," *International Journal of Heat and Mass Transfer*, vol. 28, no. 9, pp. 1711-1719, 1985.
- [28] K. S. Chen and Y. R. Chang, "Steady-state analysis of two-phase natural circulation loop," *International Journal of Heat and Mass Transfer*, vol. 31, no. 5, pp. 931-940, 1988.
- [29] P. Welander, "Note on the Self-Sustained Oscillations of a Simple Thermal System," *Tellus*, vol. 9, no. 3, pp. 419-420, 1957.
- [30] H. F. Creveling, J. F. De Paz, J. Y. Baladi and R. J. Schoenhals, "Stability characteristics of a single-phase free convection loop," *Journal of Fluid Mechanics*, vol. 67, no. 1, pp. 65-84, 1975.
- [31] J. B. Keller, "Periodic oscillations in a model of thermal convection," *Journal of Fluid Mechanics*, vol. 26, no. 3, pp. 599-606, 1966.

- [32] P. K. Vijayan, A. K. Nayak, D. S. Pilkhwal, D. Saha and V. Venkat Raj, " Effect of loop diameter on the stability of single-phase natural circulation in rectangular loops," in *5th Int. Topical Meeting on Nuclear Reactor Thermal Hydraulics, NURETH-5*, Salt Lake City, 1992.
- [33] P. Vijayan, "Experimental observations on the general trends of the steady state and stability behaviour of single-phase natural circulation loops," *Nuclear Engineering and Design*, vol. 215, pp. 139-152, 2002.
- [34] K. Chen, "On the Oscillatory Instability of Closed-Loop Thermosyphons," *Journal of Heat Transfer*, vol. 107, no. 4, pp. 826-832, 1985.
- [35] K. Chen, S. Tsai and Y. Yang, "Heat Transfer Performance of a Double-Loop Separate-Type Heat Pipe: Measurement Results," *Energy conversion and management*, vol. 35, no. 12, pp. 1131-1141, 1994.
- [36] S. S. Goodrich and W. R. Marcum, "Natural convection heat transfer and boundary layer transition for vertical heated cylinders," *Experimental Thermal and Fluid Science*, vol. 105, pp. 367-380, 2019.
- [37] S. K. S. Boetcher, *Natural Convection from Circular Cylinders*, New York: Springer, Cham, 2014.
- [38] M. Al-Arabi and M. Khamis, "Natural Convection Heat Transfer from Inclined Cylinders," *International Journal of Heat and Mass Transfer*, vol. 25, no. 1, pp. 3-15, 1982.
- [39] A. K. Mohanty and M. R. Dubey, "Buoyancy induced flow and heat transfer through a vertical annulus," *International Journal of Heat and Mass Transfer*, vol. 39, no. 10, pp. 2087-2093, 1996.
- [40] J. Mustafa , S. Husain and M. Altamush Siddiqui, "Experimental studies on natural convection of water in a closed-loop vertical annulus.," *Experimental Heat Transfer*, vol. 30, no. 1, pp. 25-45, 2017.
- [41] M. Khalid Usmani, M. Altamush Siddiqui, S. Alam, A. Jairajpuri and M. Kamil, "Heat transfer studies during natural convection boiling in an internally heated

- annulus," *International Journal of Heat and Mass Transfer*, vol. 46, pp. 1085-1095, 2003.
- [42] M. Altamush Siddiqui, M. Kamil, M. Asif and A. Jairajpuri, "Experimental study of boiling incipience in a closed loop vertical annular thermosiphon," *Applied Thermal Engineering*, vol. 30, pp. 1333-1340, 2010.
- [43] S.-Y. Chun, H.-J. Chung, S.-K. Moon, S.-K. Yang, M.-K. Chung, T. Schoesse and M. Aritomi, "Effect of pressure on critical heat flux in uniformly heated vertical annulus under low flow conditions," *Nuclear Engineering and Design*, vol. 203, pp. 159-174, 2001.
- [44] J.-W. Park, W.-P. Baek and S. H. Change, "Critical heat flux and flow pattern for water flow in annular geometry," *Nuclear Engineering and Design*, vol. 172, pp. 137-155, 1997.
- [45] T. Schoesse, M. Aritomi, Y. Kataoka, S.-R. Lee, Y. Yoshioka and M. K. Chung, "Critical Heat Flux in a Vertical Annulus under Low Upward Flow and near Atmospheric Pressure," *Journal of Nuclear Science and Technology*, vol. 34, no. 6, pp. 559-570, 1997.
- [46] M. Alipour, R. Hosseini and A. Rezaia, "Radius ratio effects on natural heat transfer in concentric annulus.," *Experimental Thermal and Fluid Science*, vol. 49, pp. 135-140, 2013.
- [47] S. K. Ravi and A. K. Singh, "Natural Convection of a Polar Fluid in Vertical Concentric Annuli," *Int. J. of Appl. Math and Mech.*, vol. 7, no. 1, pp. 70-81, 2011.
- [48] W. Qi and Y. Shiming, "Experimental Study of Natural Convection Heat Transfer of Air Layers in Vertical Annuli Under High Rayleigh Number Conditions," *Heat Transfer - Asian Research*, vol. 28, no. 1, pp. 50-57, 1999.
- [49] L. Maudou, G. H. Choueiri and S. Tavoularis, "An Experimental Study of Mixed Convection in Vertical, Open-Ended, Concentric and Eccentric Annular Channels," *Journal of Heat Transfer*, vol. 135, no. 072502, pp. 1-9, 2013.

- [50] E. M. Mokheimer and M. A. I. El-Shaarawi, "Critical Values of Gr/Re for Mixed Convection in Vertical Eccentric Annuli with Isothermal/Adiabatic Walls," *Journal of Heat Transfer*, vol. 126, no. 3, pp. 479-482, 2004.
- [51] E. M. Mokheimer and M. A. I. El-Shaarawi, "Developing Mixed Convection in Vertical Eccentric Annuli," *Heat and Mass Transfer*, vol. 41, no. 2, pp. 176-187, 2004.
- [52] S. Husain and A. Siddiqui, "Experimental and numerical analyses of natural convection flow in a partially heated vertical annulus," *Numerical Heat Transfer*, vol. 70, no. 7, pp. 763-775, 2016.
- [53] M. S. El-Genk and D. V. Rao, "Buoyancy Induced Instability of Laminar Flows in Vertical Annuli-II," *International Journal of Heat and Mass Transfer*, vol. 33, pp. 2161-2172, 1990.
- [54] M. Shahi, A. H. Mahmoudi and F. Talebi, "A numerical investigation of conjugated-natural convection heat transfer enhancement of a nanofluid in an annular tube driven by inner heat generating solid cylinder," *International Communications in Heat and Mass Transfer*, vol. 38, pp. 533-542, 2011.
- [55] S. Wang, A. Faghri and T. Bergman, "Transient Natural Convection in Vertical Annuli: Numerical Modeling and Heat Transfer Correlation," *Numerical Heat Transfer*, vol. 61, pp. 823-836, 2012.
- [56] A. H. Malik, A. Shah and S. Khushnood, "CFD Analysis of Heat Transfer Within a Bottom Heated Vertical Concentric Cylindrical Enclosure," *Journal of Physics*, vol. 439, pp. 1-15, 2013.
- [57] S. A. Lal and V. A. Kumar, "Numerical Prediction of Natural Convection in a Vertical Annulus Closed at Top and Opened at Bottom," *Heat Transfer Engineering*, vol. 34, no. 1, pp. 70-83, 2013.
- [58] Y. Inaba, Y. Zhang, T. Takeda and Y. Shiina, "Natural Convection Heat Transfer of High Temperature Gas in an Annulus Between Two Vertical Concentric Cylinders," *Heat Transfer Asian Research*, vol. 34, no. 5, pp. 293-308, 2005.

- [59] R. S. Anand, C. P. Jawahar, A. B. Solomon and E. Bellos, "A review of experimental studies on cylindrical two-phase closed thermosyphon using refrigerant for low-temperature applications," *International Journal of Refrigeration*, vol. 120, pp. 296-313, 2020.
- [60] N. Seki, S. Fukusako and K. Koguchi, "Single-Phase Heat Transfer Characteristics of Concentric-Tube Thermosyphon," *Heat and Mass Transfer*, vol. 14, pp. 189-199, 1980.
- [61] M. A. Islam, M. Monde, M. Z. Hasan and Y. Mitsutake, "Experimental study of critical heat flux in concentric-tube open thermosyphon," *International Journal of Heat and Mass Transfer*, vol. 41, pp. 3691-3704, 1998.
- [62] M. S. El-Genk and D. V. Rao, "Buoyancy Induced Instability of Laminar Flows in Vertical Annuli-I (Flow Visualization and Heat Transfer Experiments)," *International Journal of Heat and Mass Transfer*, vol. 33, pp. 2145-2159, 1990.
- [63] M. Monde, Y. Mitsutake and S. Kobo, "Critical Heat Flux during Natural Convective Boiling on Uniformly Heated Inner Tubes in Vertical Annular Tubes Submerged in Saturated Liquid," *Transactions of the Japan Society of Mechanical Engineers*, vol. 60, pp. 940-945, 1994.
- [64] J. L. Steimke, "Natural Convection Heat Transfer for a Concentric Tube Thermosiphon," *Journal of Heat Transfer*, vol. 107, no. 3, pp. 583-588, 1985.
- [65] J. McDuffee, J. J. Carbajo and D. Felde, "Design, fabrication, and modeling of a two-phase thermosyphon experimental facility for fuels and materials irradiation," *Proceedings of the 2014 Water Reactor Fuel Performance Meeting, Top Fuel, and LWR Fuel Performance Meeting*, p. 784, 2014.
- [66] J. J. Carbajo, "Comparison of RELAP5-3D and TRACE Results for a Boiling/Condensing Experimental Facility," *Transactions of the American Nuclear Society*, vol. 110, pp. 685-688, 2014.
- [67] Y. A. Cengel, A. J. Ghajar and M. Kanoglu, *Heat and Mass Transfer: Fundamentals and Applications 4th Edition*, New York: McGraw Hill Higher Education, 2011.

- [68] T. Yener, S. C. Yener and R. Mutlu, "Convection Coefficient Estimation of Still Air Using an Infrared Thermometer and Curve-Fitting," *Journal of Engineering Technology and Applied Sciences*, vol. 4, no. 2, pp. 95-103, 2019.
- [69] American Society of Mechanical Engineers, Test uncertainty, New York: ASME PTC 19.1, 2014.
- [70] Y. Zvirin, "A review of natural circulation loops in pressurized water reactors and other systems.," *Nuclear Engineering and Design*, vol. 67, no. 2, pp. 203-225, 1982.
- [71] H. F. Creveling, J. F. DePaz, J. Y. Baladi and R. J. Schoenhals, "Stability Characteristics of a Single-Phase Free Convection Loop," *Journal of Fluid Mechanics*, vol. 67, pp. 65-84, 1975.
- [72] Y. Zvirin, P. R. Jeuck III, C. S. Sullivan and R. B. Duffey, "Experimental and Analytical Investigation of a PWR Natural Circulation Loop," in *ANS/ENS Thermal Reactor Safety Meeting*, Knoxville, 1980.
- [73] Y. Zvirin, "The Effect of Dissipation on Free Convection Loops," *International Journal of Heat and Mass Transfer*, vol. 22, pp. 1539-1546, 1979.
- [74] E. E. Lewis, Nuclear Power Reactor Safety, New York: Wiley, 1977.
- [75] The International Association for the Properties of Water and Steam, IAPWS, 4 January 2021. [Online]. Available: iapws.org.
- [76] W. Jones and B. Launder, "The prediction of laminarization with a two equation model of turbulence," *International Journal of Heat and Mass Transfer*, pp. 301-314, 1972.
- [77] T.-H. Shih, W. W. Liou, A. Shabbir, Z. Yang and J. Zhu, "A New k- ϵ Eddy Viscosity Model for High Reynolds Number Turbulent Flows," *Computers Fluids*, vol. 24, no. 3, pp. 227-238, 1995.
- [78] K. Van Maele and B. Merci, "Application of two buoyancy-modified k- ϵ turbulence models to different types of buoyant plumes," *Fire Safety Journal*, vol. 41, no. 2, pp. 122-138, 2006.
- [79] P. R. Spalart and S. R. Allmaras, "A One-Equation Turbulence Model for Aerodynamic Flows," *AIAA*, no. 92-0439, 1992.

- [80] D. Wilcox, *Turbulence Modeling for CFD*, 2nd edition: DCW Industries, INC, 1998.
- [81] M. Abdollahzadeh, M. Esmailpour, R. Vizinho, A. Younesi and J. C. Pascoa, "Assessment of RANS turbulence models for numerical study of laminar-turbulent transition in convection heat transfer," *International Journal of Heat and Mass Transfer*, vol. 115, pp. 1288-1308, 2017.
- [82] F. R. Menter, "Improved Two-Equation k-omega Turbulence Models for Aerodynamic Flows," NASA Technical Memorandum, 103975, 1992.
- [83] P. R. Spalart and C. L. Rumsey, "Effective Inflow Conditions for Turbulence Models in Aerodynamic Calculations," *AIAA Journal*, vol. 45, no. 10, pp. 2544-2533, 2007.
- [84] S. Sarkar and L. Balakrishnan, "Application of a Reynolds-stress turbulence model to the compressible shear layer," ICASE Report 90-18, NASA CR 182002, 1990.
- [85] W. Xu, Q. Chen and F. T. Nieuwstadt, "A new turbulence model for near-wall natural convection," *International Journal of Heat and Mass Transfer*, vol. 41, pp. 3161-3176, 1998.
- [86] M. Wolfstein, "The velocity and temperature distribution in one-dimensional flow with turbulence augmentation and pressure gradient," *International Journal of Heat and Mass Transfer*, vol. 12, pp. 301-318, 1969.
- [87] T. Jongen, "Simulation and Modeling of Turbulent Incompressible Flows," Lausanne EPFL, Ph.D. Thesis, 1998.
- [88] S. Kenjeres, S. B. Gunarjo and K. Hanjalic, "Contribution to elliptic relaxation modelling of turbulent natural and mixed convection," *International Journal of Heat and Fluid Flow*, vol. 26, no. 4, pp. 569-586, 2005.
- [89] A. M. Savill, "Further progress in the turbulence modelling of by-pass transition," *Engineering Turbulence Modelling and Experiments*, pp. 583-592, 1993.
- [90] The American Society of Mechanical Engineers, "Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer," ASME V&V 20-2009, An American National Standard, 2009.

- [91] P. J. Roache, "Building PDE Codes to be Verifiable and Validatable," *Computing in Science and Engineering*, Vols. 1521-9615/04, pp. 30-38, 2004.
- [92] OMEGA, "Thermocouple types," A Spectris Company, 17 April 2019. [Online]. Available: <https://www.omega.com/en-us/resources/thermocouple-types>. [Accessed 17 February 2022].
- [93] LUNA, "ODiSI-B Sensor Strain Gage Factor Uncertainty," LUNA Innovations, EN-FY1602, 2016.
- [94] W. Wagner and A. Prub, "The IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use," *Journal of Physical and Chemical Reference Data*, vol. 31, no. 2, p. 387, 2002.
- [95] A. Jderu, M. A. Soto, M. Enachescu and D. Ziegler, "Liquid Flow Meter by Fiber-Optic Sensing of Heat Propagation," *Sensors MDPI*, vol. 21, no. 355, 2021.
- [96] K. J. Hsieh and F. S. Lien, "Numerical modeling of buoyancy-driven turbulent flows in enclosures," *International Journal of Heat and Fluid Flow*, vol. 25, pp. 659-670, 2004.
- [97] Siemens Digital Industries Software, "Finite Volume Discretization," 2021. [Online]. Available: <https://docs.sw.siemens.com/documentation/external/PL20201109101148301/en-US/userManual/userguide/html/index.html#page/STARCCMP/GUID-A806D14C-6833-458E-89B7-8B273EE048B7.html#wwID0EZJLBD>.
- [98] S. V. Patankar and D. B. Spalding, "A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows," *International Journal of Heat and Mass Transfer*, vol. 15, no. 10, pp. 1787-1806, 1972.
- [99] H. Reichardt, "Vollstaendige Darstellung der turbulenten Geschwindigkeitsverteilung in glatten Leitungen," *Angew. Math. Mech*, vol. 31, no. 7, pp. 208-219, 1951.
- [100] B. A. Kader, "Temperature and Concentration Profiles in Fully Turbulent Boundary Layers," *International Journal of Heat and Mass Transfer*, vol. 24, pp. 1541-1544, 1981.

- [101] W. Rodi, "Experience with two-layer models combining the k-epsilon model with a one-equation model near the wall," in *American Institute of Aeronautics and Astronautics*, AIAA Meeting Papers, 1991.

9 NOMENCLATURE

A	area
\mathbf{a}	area vector
b_i	systematic uncertainty
C_M	Sarkar coefficient
C_μ	critical coefficient
C_t	model coefficient
$C_{\varepsilon 1}$	model coefficient
$C_{\varepsilon 2}$	model coefficient
c	specific heat capacity
c	speed of sound
c_p	specific heat capacity at constant pressure
D	experimental real-world data value
\mathbf{D}	deformation tensor
d	distance to wall
d_H	hydraulic diameter
E	total energy
E	validation comparison error
\hat{e}_s	flow directional normal unit vector
\hat{e}_z	vertical unit vector
F	force
\mathbf{f}_b	body forces
\mathbf{f}_g	body forces from gravity
f	friction coefficient
f_μ	variable damping function
G_b	buoyancy production
G_k	turbulence production
g	gravitational constant
H	total enthalpy
H_{sc}^V	subcooled vapor enthalpy

H_{sh}^L	superheated liquid enthalpy
h	static enthalpy
h	heat transfer coefficient
h'	representative grid size
$h_{combined}$	combined convective and radiative heat transfer coefficient
h_{rad}	radiative heat transfer coefficient
I	identity tensor
I	current
J	convective or diffusive flux
k	thermal conductivity
k	turbulent kinetic energy
k'	expansion factor
k_b	bottom of test section flow loss coefficient
k_c	contraction flow loss coefficient
k_e	expansion flow loss coefficient
k_t	top of test section flow loss coefficient
L	length
l_ε	length scale function
\dot{m}	mass flow rate
\hat{n}	normal unit vector
P	power
P_k	turbulent kinetic energy production
P_ε	turbulent dissipation production
p	pressure
p'	apparent order
p_{piezo}	piezometric pressure
p_{static}	reference static altitude pressure
\dot{Q}	heat transfer rate
q	heat flux (q'')
\dot{q}	sum heat flux

\bar{q}	mean heat flux
q_c''	heat flux at heat sink location
q_h''	heat flux at heat source location
\dot{q}_{bb}''	black body local surface heat flux
R	flow resistance parameter
S	modulus of the mean strain rate tensor
\mathbf{S}	mean strain rate tensor
S	simulation solution value
S_c	heat sink end length
S_h	heat source length
S_{hl}	heat sink start length
S_u	unit source
S_E	energy source
s	one-directional spatial parameter
s_{ik}	random standard uncertainty
\mathbf{T}	viscous shear stress tensor
\mathbf{T}_{RANS}	Reynolds Averaged Navier-Stokes stress tensor
T_e	large eddy time scale
T_∞	ambient temperature
T_s	surface temperature
T_{sat}	saturation temperature
t	time
u	velocity normal to flow direction
u_{num}	numerical uncertainty
u_{input}	uncertainty due to simulation input
u_D	uncertainty from experimental measurements
u_τ	shear stress-based velocity scale
\forall, V	volume
\vec{V}	velocity vector
\dot{V}	volumetric flow rate

V	voltage
\mathbf{v}	normal velocity vector
W	modulus of mean vorticity tensor
\mathbf{W}	mean vorticity tensor
y^+	y-plus wall value
α	thermal diffusivity
β	thermal expansion coefficient
Γ	diffusion coefficient
γ	geometrical area parameter
Δ	delta (change in)
δ	comparison error
ε	emissivity
ε	turbulent dissipation rate
θ	dimensionless temperature
κ	von Karman constant
λ	wall-proximity indicator
μ	dynamic viscosity
μ_t	turbulent eddy viscosity
$\mu\varepsilon$	micro strain
ν	kinematic viscosity
ξ	wetted perimeter of flow section
ρ	density
ρ_{ref}	reference density
Σ	summation
σ	Stefan Boltzmann constant
$\boldsymbol{\sigma}$	stress tensor
τ_w	shear stress at wall
Υ_M	Sarkar compressibility modification function
Φ	dissipation function

ϕ	solution variable
φ	solution value
∇	gradient
\otimes	cross product
Annular	geometry between two concentric circles
Buoyancy	upward force exerted by fluid opposing weight of immersed object
Cartridge	self-contained apparatus used in irradiation experiments
Concentric tube thermosyphon	closed thermosyphon with concentric tube within cylindrical casing
Core heater	component surrounding O-SERTTA heater rod
Downcomer	portion of fluid where velocity is in direction of gravity gradient
External loop	irradiation experiment design where most equipment is outside of irradiation site
Flow Pipe	concentric tube separating riser and downcomer
Lower tube	tube surrounding core heater of O-SERTTA
Riser	portion of the fluid where velocity is against gravity gradient
Spider	three-pronged attachment at the bottom of O-SERTTA
Thermosyphon	method of passive heat exchange based on natural convection
CAD	computer aided design
CFD	computational fluid dynamics
CHF	critical heat flux
CHT	conjugate heat transfer
CS	control surface
CV	control volume
DNS	direct numerical simulation
GCI	Roache's Grid Convergence Index
Gr	Grashof number
HBWR	Halden boiling water reactor

HFIR	high flux isotope reactor
IAPWS	International Association for the Properties of Water and Steam
LOCA	loss of coolant accident
LVDT	linear voltage differential transformer
LWR	light water reactor
MARCH	minimal activation retrievable capsule holder
MMS	method of manufactured solutions
NCL	natural circulation loop
OSU	Oregon State University
Pr	Prandtl number
PWR	pressurized water reactor
Ra	Rayleigh number
RANS	Reynolds-Averaged Navier Stokes
Re	Reynolds number
Ri	Richardson number
SERTTA	static environment rodlet transient test apparatus
SST	shear stress transport
SWF	standard wall functions
TREAT	Transient REActor Test Facility
VTR	Versatile Test Reactor

A. APPENDIX

A.1 Numerical Flow Solution

The finite volume method subdivides the solution domain into a finite number of control volumes based on the meshing. The objective is to obtain a set of linear algebraic equations based on the discrete versions of the integral conservation equations. The set of linear equations are comprised of many unknown values corresponding to the number of cells in the domain. The governing equations are discretized in space and time to be solved with a multigrid solver. For unsteady problems, the physical time interval to be analyzed is subdivided into sub-intervals called time-steps. The general transport equation, given by A.1, writes out the closed set of equations in a generic form over a control volume.

$$\frac{d}{dt} \int_V \rho \phi dV + \int_A \rho \mathbf{v} \phi \cdot d\mathbf{a} = \int_A \Gamma \nabla \phi d\mathbf{a} + \int_V S_\phi dV \quad \text{A.1}$$

The variables and the key terms of the transport equation are shown in Table A.1.

Table A.1. Transport equation variables and terms.

Variable	Description	Notes
ϕ	Property	The transport of a scalar property
A	Area	Surface area of the control volume
$d\mathbf{a}$	Area vector	The surface area vector of the control volume
Γ	Diffusion Coefficient	Proportionality constant between the molar flux due to molecular diffusion and the gradient in the concentration of the species
S_ϕ	Source	The source quantity of the scalar property
$\frac{d}{dt} \int_V \rho \phi dV$	Transient Term	The time rate of change of fluid property ϕ inside the control volume
$\int_A \rho \mathbf{v} \phi \cdot d\mathbf{a}$	Convective Flux	The net rate of decrease of fluid property ϕ across the control volume boundaries due to convection
$\int_A \Gamma \nabla \phi d\mathbf{a}$	Diffusive Flux	The net rate of increase of fluid property ϕ across the control volume boundaries due to diffusion
$\int_V S_\phi dV$	Source Term	The generation or destruction of fluid property ϕ inside the control volume

To illustrate the discretization of the general transport equation, consider the two neighboring polyhedral cells shown in Figure A.1.

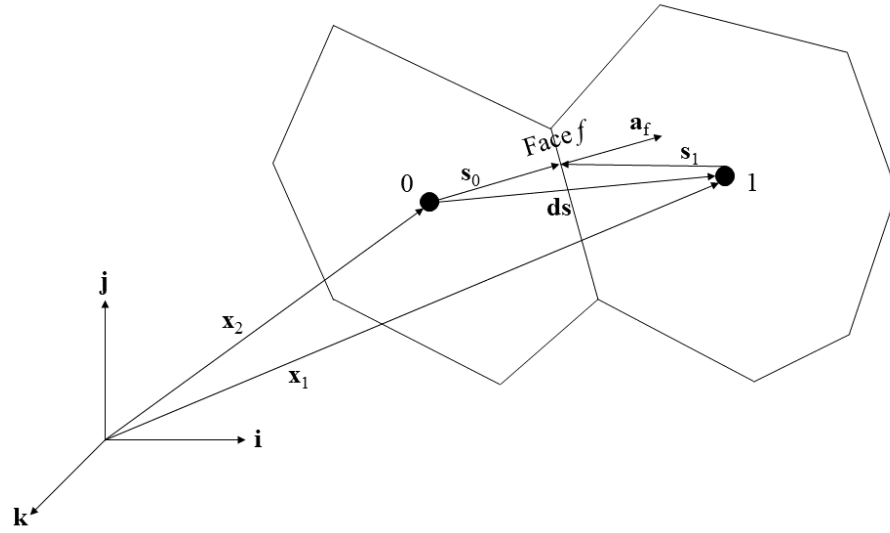


Figure A.1. Cell discretization illustration of two neighboring cells.

The surface integrals of Equation A.1 are evaluated using quadrature approximations. The integral is expressed with variable values at one location on the cell face. Using the second-order midpoint rule, the integral is evaluated as the product of the cell face center value and the cell face area, shown in Equation A.2.

$$\int_A \mathbf{J}^\phi \cdot d\mathbf{a} \approx \sum_f \mathbf{J}_f^\phi \cdot \mathbf{a}_f \quad \text{A.2}$$

\mathbf{J}^ϕ is the convective or diffusive flux of the fluid property ϕ . The second term in A.2 is the summation of a product over all the faces (f) of the cell, where \mathbf{a}_f is the surface area vector of face f . The cell face center values are unknown, so interpolation is used to approximate the values based on the cell center values.

The volume integrals in Equation A.2 are approximated by the product of the mean value of the source term at the center of the cell and the volume of the cell, which is accurate to the second order given by the following equation.

$$\int_V S_\phi dV = S_{\phi_0} V_0 \quad \text{A.3}$$

The cell face center is the area weighted center, and the cell center is the volume center. The simplified transport equation is given in A.4.

$$\frac{d}{dt}(\rho\phi V)_0 + \sum_f [\rho\phi(\mathbf{v} \cdot \mathbf{a})]_f = \sum_f (\Gamma \nabla \phi \cdot \mathbf{a})_f + (S_\phi V)_0 \quad \text{A.4}$$

The convective term at the face is rearranged in terms of the mass flow rate at the face (\dot{m}_f).

$$[\rho\phi(\mathbf{v} \cdot \mathbf{a})]_f = (\dot{m}\phi)_f = \dot{m}_f \phi_f \quad \text{A.5}$$

The fluid property, ϕ , at the face must be known. For the case of known boundary conditions, if the equations are applied at the boundary, ϕ_f is replaced by the boundary value, ϕ_b . The values are computed from the cell using a second-order numerical scheme. For second-order upwind schemes, the following relations are used.

$$(\dot{m}\phi)_f = \begin{cases} \dot{m}_f \phi_{f,0} & \text{for } \dot{m}_f \geq 0 \\ \dot{m}_f \phi_{f,1} & \text{for } \dot{m}_f \leq 0 \end{cases} \quad \text{A.6}$$

The face value quantities of the fluid property, ϕ , are linearly interpolated from the cell center values on either side of the face.

$$\begin{aligned} \phi_{f,0} &= \phi_0 + \mathbf{s}_0 \cdot (\nabla \phi)_{r,0} & \text{where } \mathbf{s}_0 &= \mathbf{x}_f - \mathbf{x}_0 \\ \phi_{f,1} &= \phi_1 + \mathbf{s}_1 \cdot (\nabla \phi)_{r,1} & \text{where } \mathbf{s}_0 &= \mathbf{x}_f - \mathbf{x}_1 \end{aligned} \quad \text{A.7}$$

The diffusive flux through internal cell faces of a cell are discretized as follows.

$$D_f = (\Gamma \nabla \phi \cdot \mathbf{a})_f = \Gamma_f [(\phi_1 - \phi_0) \vec{\alpha} \cdot \mathbf{a} + \overline{\nabla \phi} \cdot \mathbf{a} - (\overline{\nabla \phi} \cdot \mathbf{ds}) \vec{\alpha} \cdot \mathbf{a}] \quad \text{A.8}$$

For the diffusive flux at a boundary face, the subscript 1 is replaced with the subscript denoting the boundary face, f . Additional details of the diffusive flux equation are given in Table A.2.

Table A.2. Diffusive flux term details.

Variable	Description	Notes
$\vec{\alpha}$	$\frac{\mathbf{a}}{\mathbf{a} \cdot \mathbf{ds}}$	The surface area vector \mathbf{a} divided by the product of the surface area vector and the cell center distance, \mathbf{ds}
\mathbf{ds}	$\mathbf{x}_{f,1} - \mathbf{x}_0$	Subscript f is used at a boundary face and 1 is the neighboring internal cell
$\overline{\nabla \phi}$	$\frac{(\nabla \phi_0 + \nabla \phi_1)}{2}, \nabla \phi_0$	The gradient of fluid property ϕ . The first term is for internal cell faces and the second term is for boundary faces
Γ_f	Diffusion Coefficient	Harmonic average of the face diffusivity cell values

The last two terms of A.8 represent the secondary gradient contribution which are essential for maintaining accuracy in non-orthogonal meshes. The formulation for the diffusive term assumes the centroids of cells 0 and 1 (shown in Figure A.1) lie on opposing sides of the face. The convention used assumes the face area vector points out of cell 0. Meshing should be valid in terms of the skewness angle, which is the angle between the vectors \mathbf{a}_f and \mathbf{ds} in Figure A.1. The skewness angle presents a non-physical solution if greater than 90 degrees. If the diffusive fluxes are calculated at the boundary, the substitutions to be made are shown in Table A.2.

Time is an additional coordinate in transient simulations. The temporal discretization is required in addition to spatial discretization where the total time interval is subdivided into time-steps. The governing solutions must be solved at each time-step, where the solution at some time requires the solutions from previous time-steps. Time-integration schemes are distinguished by the number of time-levels used for integration and which time-levels the fluxes and sources are integrated over. A first-order temporal scheme, called the Euler implicit scheme, approximates the transient terms in the governing equations using the solution at the current time-step in addition to the solution from the previous time-step.

$$\frac{d}{dt}(\rho\phi V)_0 = \frac{(\rho\phi V)_0^{n+1} - (\rho\phi V)_0^n}{\Delta t} \quad \text{A.9}$$

As more time-step solutions become available, higher-order temporal schemes can be applied by using solutions from previous two or more time levels (n) using backward differentiation formulas. As an example, the following formula is applied for using solutions from the previous two time levels.

$$\frac{d}{dt}(\rho\chi\phi V)_0 = \frac{\frac{3}{2}(\rho\chi\phi V)_0^{n+1} - 2(\rho\chi\phi V)_0^n + \frac{1}{2}(\rho\chi\phi V)_0^{n-1}}{\Delta t} \quad \text{A.10}$$

9.1.1 Gradients

Variable gradients are required at cell centers and cell-face centers for construction of variable values at cell faces, secondary gradients calculations for diffusion terms, pressure gradients calculations for pressure-velocity coupling, and strain-rate and rotation-rate

calculations for turbulent flows. The Hybrid Gauss-Least Squares Method [97] is given by A.11.

$$\nabla\phi = \sum_f (\phi_n - \phi_0)w_f^0 \quad \text{A.11}$$

Where the left side of the equation is the gradient of some fluid property ϕ . The additional terms are explained in Table A.3.

Table A.3. Hybrid Gauss-Least Squares Method term definitions.

Variable	Equation	Description and notes
w_f^0	$\beta w_f^{lsq} + (1 - \beta)w_f^G$	β is the geometric Gauss-LSQ gradient blending factor field function.
w_f^{lsq}	$\left[\sum_f \frac{ds \otimes ds}{ds^2} \right]^{-1} \frac{ds}{ds^2}$	The least squares method gradient calculation. Subscript f is used at a boundary face and ds is the distance between cell centroids. The centroid of neighboring cell (x_n) and the quantified cell (x_0) addressed through face f .
ds	$x_n - x_0$	The centroid of neighboring cell (x_n) and the quantified cell (x_0) addressed through face f .
w_f^G	$\frac{A_f}{V_0 + V_n}$	The green-gauss method gradient calculation. A_f is the face f area-vector, V_n and V_0 are the respective cell volumes.

The gradient blending factor (w_f^0) is used to determine how the two gradient method options are used to calculate the gradient. A blending factor of 1 equates to using the least squares method, 0 equates to using the green-gauss method, and in between 0 and 1 equates to using a combination of the two. The cell value of the blending factor is computed based on input values and mesh quantities. The blending value is 1 if the least-squares tensor eigenvalues ratio of the cell is larger than the specified minimum value, the tangent of the skewness angle is smaller than the Normalized Flat Cells Curvature Factor times the aspect ratio of the cell, the skewness angle is smaller than the maximum safe skewness angle. Additionally, the blending value is 0 if it is a Chevron-cell and 1 otherwise. Chevron cells exist when thin, slender cells meet at a shared face at an angle where the line joining the cell centers does not pass through the shared face.

The gradients must be limited because when used for reconstruction, the gradient face values can fall outside the range of cell values found in neighboring cells. Gradient limiters

are used to find the minimum and maximum bounds of the neighboring cell values to limit the reconstruction gradients. The face value after being reconstructed is given by A.12.

$$\phi_{f,0} = \phi_0 + \mathbf{s}_0 \cdot (\nabla\phi)_{r,0} \quad \text{A.12}$$

The term \mathbf{s}_0 is the difference between the face centroid (\mathbf{x}_f) and the cell centroid (\mathbf{x}_0) locations. The last term is the reconstruction gradient $((\nabla\phi)_{r,0})$, which ensures the reconstructed face value does not exceed the minimum and maximum of the neighboring cell centroid values. The scale factor (α) expresses the ratio of the limited and unlimited values and can be used in the following relation.

$$(\nabla\phi)_{r,0} = \alpha \nabla\phi \quad \text{A.13}$$

For each cell face of the reconstructed cell the following are defined for quantifying the scale factor using the Venkatakrishnan method.

$$\Delta_f = \phi_{f,0} - \phi_0 \quad \text{A.14}$$

$$r_f = \begin{cases} \frac{\Delta_f}{\Delta_{max}} & \text{for } \Delta_f > 0 \\ \frac{\Delta_f}{\Delta_{min}} & \text{for } \Delta_f \leq 0 \end{cases} \quad \text{A.15}$$

$$\alpha_f = \frac{2r_f + 1}{r_f(2r_f + 1) + 1} \quad \text{A.16}$$

The values Δ_{max} and Δ_{min} are the difference in the maximum, or minimum quantities of the concerning cell and the neighboring cells with respect to the cell of concern.

The algebraic system for the transported variable, ϕ , is given by the following.

$$\frac{a_p}{\omega} \Delta\phi_p + \sum_n a_n \Delta\phi_n = r \quad \text{A.17}$$

The summation is over all the neighbors, n , of cell p . The coefficients a_p and a_n are obtained from the discretized terms and ω is an under-relaxation factor. The right-hand side of the equation is termed the residual, which represents the discretized form of the original equation at some iteration shown in A.18.

$$r = \frac{d}{dt}(\rho\phi V)_0 + \sum_f [\rho\phi(\mathbf{v} \cdot \mathbf{a})]_f - \sum_f (\Gamma \nabla\phi \cdot \mathbf{a})_f - (S_\phi V)_0 \quad \text{A.18}$$

The residual becomes zero when the discretized equation is satisfied exactly. Non-linearity in CFD problems require iterative solutions. The outer iteration loop controls the solution

update, and the inner iteration loop governs the iterative solution of the linear system. The outer iterations are repeated multiple times which makes it sufficient to solve the linear system only approximately at each outer iteration. The iterative solution of the linear system is accomplished using a multigrid solver.

9.1.2 Segregated Flow Solver

Segregated flow solver approaches the conservation equation solutions in a sequential manner. A pressure-velocity coupling algorithm is employed where the mass conservation constraint on the velocity field is fulfilled by solving a correction equation. The pressure-correction equation is constructed from the continuity and momentum equations such that the predicted velocity field is sought to fulfill the continuity equation, which is achieved by correcting pressure. This method is also known as the predictor-corrector approach.

Considering the cells shown in Figure A.1, the discretized momentum equation is shown in Equation A.19 for some cell 0.

$$\frac{\partial}{\partial t}(\rho \mathbf{v} V)_0 + \sum_f [\rho \mathbf{v} \mathbf{v} \cdot \mathbf{a}]_f = - \sum_f (p \mathbf{I} \cdot \mathbf{a})_f + \sum_f \mathbf{T} \cdot \mathbf{a} \quad \text{A.19}$$

The pressure in the momentum equations is a gradient, which must be evaluated at each cell face.

$$p_f = \frac{\bar{a}_0 p_{f0} + \bar{a}_1 p_{f1}}{\bar{a}_0 + \bar{a}_1} \quad \text{A.20}$$

The average of the momentum coefficients for all the components of momentum for cells 0 and 1 are \bar{a}_0 and \bar{a}_1 , respectively. The pressure terms for cells 0 and 1 are interpolated from cell values and reconstruction gradients according to A.12. The continuity equation can be re-written in terms of a mass flux correction (\dot{m}'_f) to couple the pressure and velocity.

$$\sum_f \dot{m}_f = \sum_f (\dot{m}_f^* + \dot{m}'_f) = 0 \quad \text{A.21}$$

The uncorrected face mass flux (\dot{m}_f^*) is computed after the discrete momentum equations have been solved. An initial guess of the pressure field (p^*) allows for the momentum equations to be solved but not the other continuity equations. The uncorrected mass flux at an interior face is shown in Equation A.22.

$$\dot{m}_f^* = \rho_f \mathbf{a} \cdot \left(\frac{\mathbf{v}_0^* + \mathbf{v}_1^*}{2} \right) - Y_f \quad \text{A.22}$$

The cell velocities (\mathbf{v}_0^* and \mathbf{v}_1^*) are solved for in the discrete momentum equations. Linear interpolation from the cell center velocities to the face results in nonphysical pressure gradients. To mitigate this, the Rhie-and-Chow dissipation term (Y_f) is introduced.

$$Y_f = Q_f (p_1 - p_0 - \overline{\nabla p_f^*} \cdot ds) \quad \text{A.23}$$

$$Q_f = \rho_f \left(\frac{V_0 + V_1}{a_0 + a_1} \right) \boldsymbol{\alpha} \cdot \mathbf{a} \quad \text{A.24}$$

The volume-weighted average of the cell gradients of pressure ($\overline{\nabla p_f^*}$) is computed using a volume weighted interpolation between the gradient values of the two cells. In other words, the cell 0 volume (V_0) and the cell 1 volume (V_1) are weighted about face f to interpolate between the gradient values of the cells. The mass flow correction is shown in A.25.

$$\dot{m}_f' = Q_f (p_0^* - p_1^*) + \frac{\dot{m}_f^*}{\rho_f} \left(\frac{\partial \rho}{\partial p} \right)_T p_{upwind}' \quad \text{A.25}$$

$$p_{upwind}' = \begin{cases} p_0' & \text{for } \dot{m}_f^* > 0 \\ p_1' & \text{for } \dot{m}_f^* < 0 \end{cases} \quad \text{A.26}$$

Combining Equations A.21 and A.25 gives the discrete pressure correction equation in A.27, where n is iterated over the neighboring cells.

$$p_p' + \sum_n a_n p_n' = r \quad \text{A.27}$$

$$r = - \sum_n \dot{m}_f^* \quad \text{A.28}$$

The segregated solver uses the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) algorithm [98], which is summarized in Figure A.2.

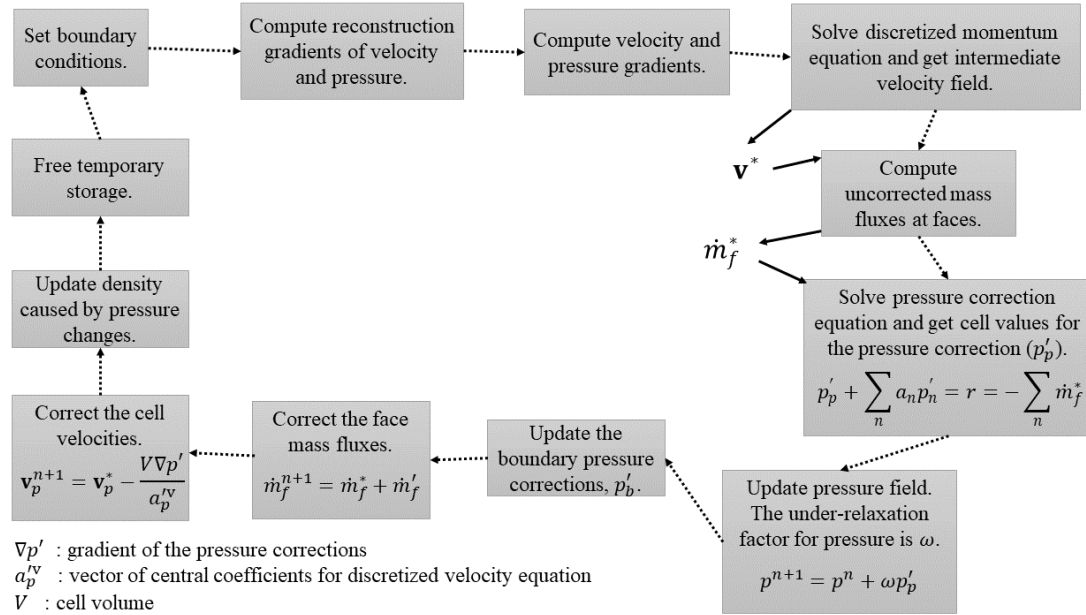


Figure A.2. SIMPLE algorithm overview.

9.1.3 Algebraic Multigrid Methodology

As the mesh size of CFD models grows, conventional iterative solution algorithms converge significantly slower. The Algebraic Multigrid (AMG) method is employed in STAR-CCM+ to accelerate solver convergence. AMG derives a coarse system of the model without direct reference to the model's geometrical mesh. The coarse system equations are derived from arithmetic combinations of the fine grid equation coefficients. Iterating through the several iterations of the fine grid, the computations are transferred from the fine system to the coarse linear system. This allows for efficient reduction of the numerical error. The AMG method agglomerates the cells to form a coarse system, transfers the residuals from fine level to coarse level quantities (restriction), and then transfers the correction from the coarse level back to the fine mesh level (prolongation).

Basic iteration schemes like Gauss-Seidel are greatly accelerated by multigrid techniques. Simple correction sweeps over the coarse meshes have a significant impact on algorithm efficiency. Fixed and flexible cycling strategies are used with the AMG solver. Fixed cycles go through a complete multigrid cycle according to residual values. The fixed cycle begins by smoothing the fine grid solution by applying a number of iterative relaxation

sweeps to get a new set of corrections. The residuals are then restricted, that is transferred to the next coarser mesh level where a new cycle is applied. Following the cycle in the coarse level, the corrections are transferred back to the fine level. The final fine level residuals are then smoothed again, and the process is repeated based on the cycle type. Three common fixed cycle types are shown in Figure A.3 for four different mesh resolutions.

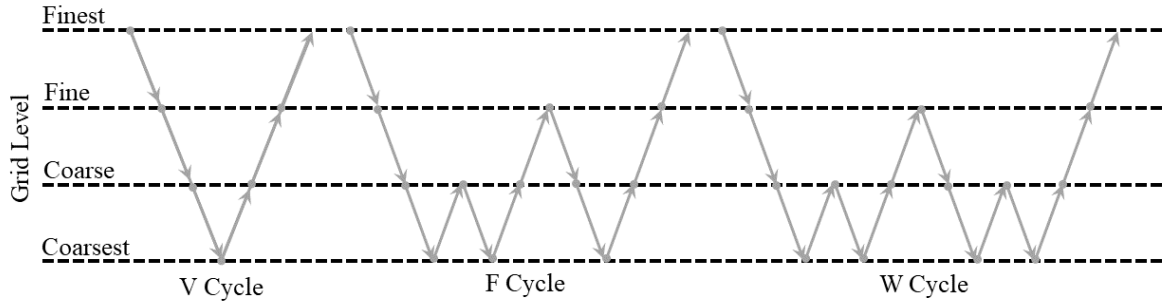


Figure A.3. Fixed cycling strategies.

Flexible cycles save computational time by using a non-stiff cycling strategy based on monitoring residuals. Rather than transferring the restricted solutions to the fine grid in a regular pattern, the residual rate of reduction dictates if the solution continues at the course level. If the residual reduction is high enough the calculations are continued at the course level mesh, otherwise the solution moves to a finer level like the fixed cycle strategies.

The governing discretized equations can be represented as a linear system given in A.29.

$$Ax = b \quad \text{A.29}$$

This equation applies to each computational cell; Equation A.17 can be used as an example of the linear system. Matrix A represents the coefficients of the linear system ($\frac{a_p}{\omega}$ and a_n), the vector x represents the unknowns ($\Delta\phi_n$), and the vector b is the residuals (r). The resultant linear system of equations is solved using the Jacobi or Gauss-Seidel decomposition relaxation schemes. The schemes are given an approximate solution (x^k) and seeks to calculate a better approximation (x^{k+1}) in a repeated procedure. The error at iteration k (e^k) is quantified as follows and shows that continuing iterations until the residual is small, also causes the error to be small.

$$e^k = x - x^k \quad \text{A.30}$$

$$r^k = b - Ax^k \rightarrow r^k = Ae^k \quad \text{A.31}$$

The Jacobi and Gauss-Seidel iterative methods involve passing through each cell in sequence and updating the value of x_i (the unknown property vector) in each cell based on the coefficients of the neighboring cells, n .

$$x_i = \frac{1}{A_{i,i}} \left(b - \sum_{\text{neighbors}, n} A_{i,n} x_n \right) \quad \text{A.32}$$

The Jacobi method uses the old values of x_n from the prior iteration while the Gauss-Seidel method uses the updated values available from the current iteration. Gauss-Seidel results in better solution convergence.

To accelerate convergence of the AMG solver, the Bi-Conjugate Gradient Stabilized (BiCGStab) method is used to solve nonsymmetric linear systems. To solve the linear system in Equation A.29, the procedure in Figure A.4 is followed. The robustness and speed of convergence for linear systems is improved with this method.

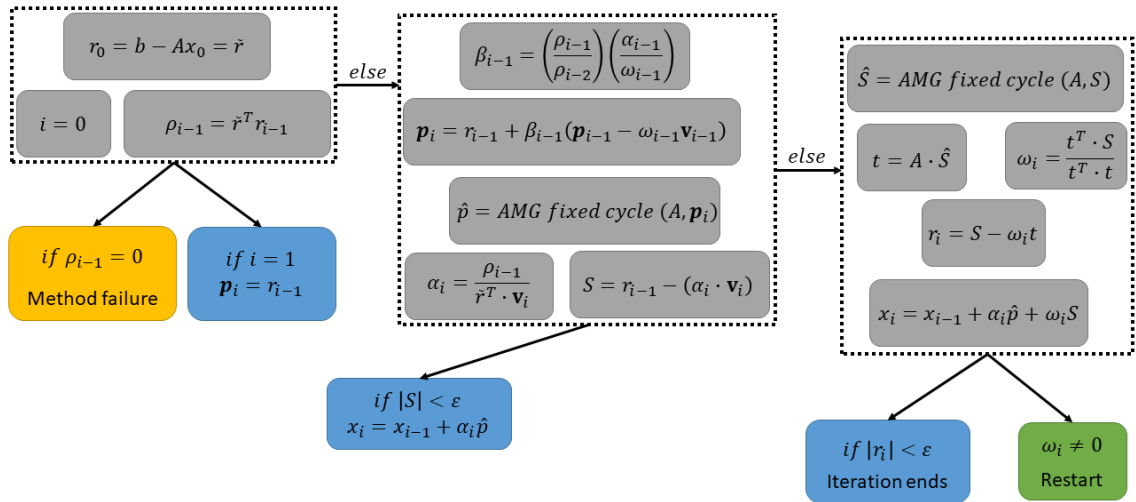


Figure A.4. Bi-Conjugate Gradient Stabilized method overview.

A.2 Wall Treatment

Wall treatment in CFD simulations is important to consider. For flows with high velocities, the boundary layer is very thin. The velocities in this thin layer will be smaller due to fluid viscosity. The velocities change rapidly as distance from the wall increases in the boundary

layer. To capture the effects of the boundary layer, the appropriate wall meshing and the y^+ wall treatment should be used when modeling the physics. STAR-CCM+ offers low-wall, high-wall, and all-wall y^+ treatment for capturing the boundary layer physics. Flow in the boundary layer can be laminar or turbulent, depending on Reynolds (Re) number. For low Re, the streamwise velocity changes uniformly away from the wall. Turbulent boundary layers are split into three sections along the flow direction: the viscous, log-law, and buffer layers. The three layers can be visualized in Figure A.5.

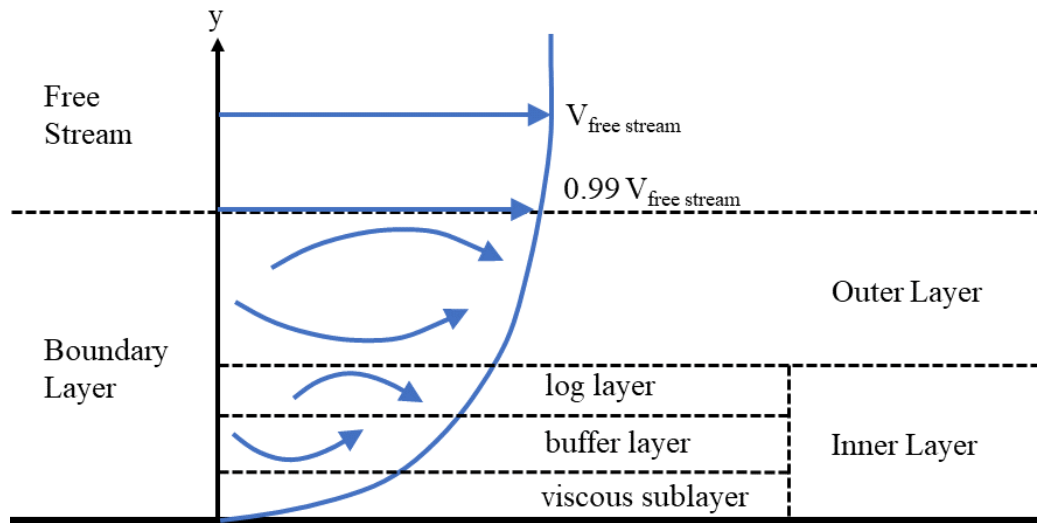


Figure A.5. Turbulent boundary layers.

Turbulent effects dominate in the outer layer. The viscous sublayer is fluid in contact with the wall dominated by viscous effects, the log layer is dominated equally by viscous and turbulent effects, and the buffer layer is transitional between the log and viscous layers. No specific modeling of near-wall turbulence is required when using DNS. However, using RANS turbulence models requires additional physics modeling of the near-wall turbulence and wall treatment. Near-wall modeling deals with effects of the walls on turbulence in the viscous sublayer and buffer layer. The two-layer approach in the $k-\varepsilon$ model uses damping functions to account for wall blocking effects. The wall treatment provides boundary conditions for the flow and energy equations. Wall functions are used to quantify the turbulent quantities in the centroids of near wall cells.

9.1.4 Wall Functions

Wall functions provide approximations to velocity, temperature, and turbulence quantities in the inner turbulent boundary layer. Standard functions are used specifically for the viscous sublayer or the log layers. Blended functions are continuous functions that cover all three inner turbulent layers. The buffer layer is represented by blending the viscous sublayer and the log layer. The non-dimensional definitions of the wall functions are given in Table A.4.

Table A.4. Wall functions definitions and equations.

Variable	Description	Non-dimensional equation
y	Wall distance	$y^+ = \frac{y\rho u_*}{\mu}$, μ is viscosity
u	Wall-tangential velocity component of the velocity vector	$u^+ = \frac{u}{u_*}$, u_* is the velocity scale
T	Temperature	$T^+ = \rho C_p u_* \frac{T - T_w}{q''}$, q'' is wall heat flux, T_w is wall temperature
μ_t	Turbulent eddy viscosity	$\mu_t^+ = \frac{\mu_t}{\mu}$
\tilde{v}	Modified diffusivity	$\tilde{v}^+ = \frac{\rho \tilde{v}}{\mu}$
$P_{\tilde{v}}$	Production of modified diffusivity	$P_{\tilde{v}}^+ = \frac{P_{\tilde{v}}}{u_*^2} = -\frac{1+C_{b2}}{\sigma_{\tilde{v}}} \frac{(\tilde{v}^+)^2}{(y^+)^2}$
k	Turbulent kinetic energy	$k^+ = \frac{k}{u_*^2}$
P_k	Production of turbulent kinetic energy	$P_k^+ = \frac{P_k \mu}{\rho u_*^4} = \gamma \left(\mu_t^+ \frac{\partial u^+}{\partial y^+} \right) + (1 - \gamma) \frac{1}{\kappa y^+}$
ε	Dissipation rate	$\varepsilon^+ = \frac{\varepsilon \mu}{\rho u_*^4} = \frac{(k^+)^{3/2}}{l_\varepsilon^+ y^+}$ where $l_\varepsilon^+ = \frac{l_\varepsilon}{y}$
ω	Specific dissipation rate	$\omega^+ = \frac{\omega \mu}{\rho u_*^2} = \sqrt{\frac{6}{\beta_1 (\gamma^+)^2} + \frac{1}{\sqrt{\beta^* \kappa} y^+}}$

The velocity scale (u_*) is approximated for the near-wall flow region using the following formula with the realizable k - ε turbulence model.

$$u_* = \gamma \sqrt{\frac{\mu |\mathbf{v}_{tan}|}{\rho \gamma}} + (1 - \gamma) C_\mu^{1/4} k^{1/2} \quad \text{A.33}$$

The wall-tangential velocity vector (\mathbf{v}_{tan}), turbulence model coefficient (C_μ), and damping function (γ) are calculated by other equations.

$$\gamma = e^{-\frac{Re_d}{11}} \quad \text{A.34}$$

The wall functions for velocity describe the distribution of the non-dimensional wall-tangential velocity (u^+) in the three inner turbulent boundary layers. The realizable k - ε model uses Reichardt's law [99], given by the following.

$$u^+ = \frac{1}{\kappa} \ln(1 + \kappa y^+) + C \left(1 - e^{-\frac{y^+}{y_m^+}} - \frac{y^+}{y_m^+} e^{-by^+} \right) \quad \text{A.35}$$

$$C = \frac{1}{\kappa} \ln \left(\frac{E'}{\kappa} \right), b = \frac{1}{2} \left(y_m^+ \frac{\kappa}{C} + \frac{1}{y_m^+} \right) \quad \text{A.36}$$

$$y_m^+ = \max[3, 267 * (2.64 - 3.9\kappa) E'^{0.0125}] - 0.987 \quad \text{A.37}$$

The variable κ is the von Karman constant (0.42) and E' is the log law offset ($E = 9$) divided by the roughness function (f). The non-dimensional temperature quantity in the inner turbulent boundary layers is based on Kader's law [100].

$$T^+ = e^{-\Gamma} \text{Pr} y^+ + e^{-1/\Gamma} \text{Pr}_t \left[\frac{1}{\kappa} \ln(E' y^+) + P \right] \quad \text{A.38}$$

$$\Gamma = \frac{0.01c(\text{Pr} y^+)^4}{1 + \frac{5}{c} \text{Pr}^3 y^+}, c = e^{f-1} \quad \text{A.39}$$

$$P = 9.24 \left[\left(\frac{\text{Pr}}{\text{Pr}_t} \right)^{3/4} - 1 \right] \left[1 + 0.28 e^{-0.007 \frac{\text{Pr}}{\text{Pr}_t}} \right] \quad \text{A.40}$$

The term P governs velocity at the thermal profile intersection of the log layer and viscous sub-layer. The turbulence wall functions describe the non-dimensional turbulent quantities in the inner boundary layer. No wall functions are used for the non-dimensional modified diffusivity ($\tilde{\nu}^+$) and the turbulent kinetic energy (k^+) quantities due to strong dependence of the flow Reynolds number. However, the production terms of these two quantities ($P_{\tilde{\nu}}^+$ and P_k^+) can be described with wall functions.

9.1.5 Wall Treatment for Flow and Energy

The flow and energy models are given boundary conditions at the wall to close the equations. Considering a simplified 2D mesh shown in Figure A.6, where the x-axis is parallel to the wall and the y-axis is the wall-normal direction.

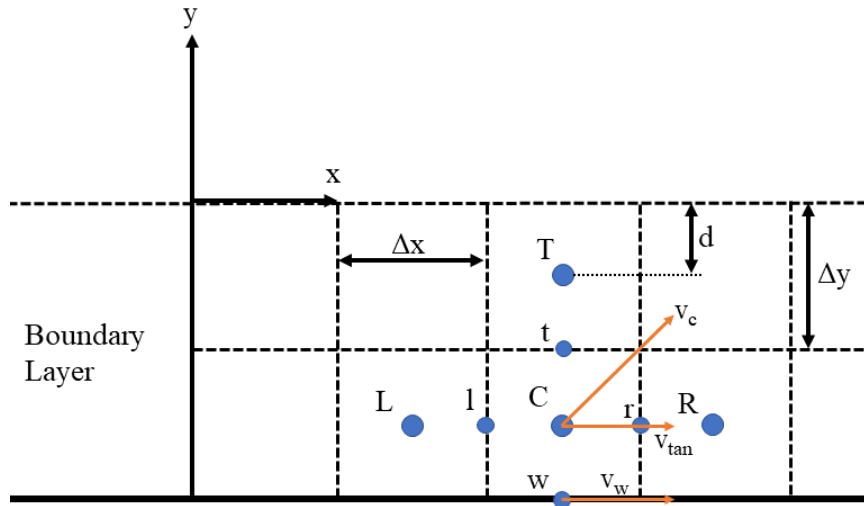


Figure A.6. Simple 2D schematic for wall treatment.

The steady-state transport equation with no external forces for a scalar quantity is given by the following.

$$\frac{\partial}{\partial x}(\rho \mathbf{v} \phi) = \frac{\partial}{\partial y} \left(\Gamma \frac{\partial \phi}{\partial y} \right) + S_\phi \quad \text{A.41}$$

Integrating over a cell volume near the wall, cell C in Figure A.6, gives the following equation.

$$[(\rho \mathbf{v} \phi)_r - (\rho \mathbf{v} \phi)_l] \Delta y = \left[\left(\Gamma \frac{\partial \phi}{\partial y} \right)_t - \left(\Gamma \frac{\partial \phi}{\partial y} \right)_w \right] \Delta x + S_\phi \Delta x \Delta y \quad \text{A.42}$$

The subscripts denote the cell face location respectively. The wall-normal gradient term corresponds to wall shear stress (τ_w) in the momentum equations or it corresponds to the wall heat flux (\dot{q}_w'') in the energy equations. The relations based on flow regime (laminar or turbulent) are given in Table A.5.

Table A.5. Wall specifications for momentum and energy equations.

Variable	Description	Condition	Definition
τ_w	The wall shear stress.	Laminar	$\tau_w = \left\{ \mu [(\nabla \mathbf{v}_{tan})_C - (\nabla \mathbf{v}_{tan})_C^T] - \frac{2}{3} (\nabla \mathbf{v}_{tan})_C \mathbf{I} \right\} \mathbf{n}$ $\mathbf{v}_{tan} = (\mathbf{v}_C - \mathbf{v}_w) - [(\mathbf{v}_C - \mathbf{v}_w) \mathbf{n}] \mathbf{n}$
		Turbulent	$\tau_w = \left \rho u_\tau^2 \frac{\hat{\mathbf{v}}_{tan}}{ \hat{\mathbf{v}}_{tan} } \right $
\dot{q}_w''	The wall heat flux.	Laminar	$\dot{q}_w'' = -k (\nabla T)_C $
		Turbulent	$\dot{q}_w'' = \rho C_p u_* \frac{\hat{T}_C - T_w}{\hat{T}_C}$
$\left(\Gamma \frac{\partial \phi}{\partial y} \right)_w$	The wall-normal gradient.	Momentum equations	$\left(\Gamma \frac{\partial \phi}{\partial y} \right)_w = \tau_w$
		Energy Equations	$\left(\Gamma \frac{\partial \phi}{\partial y} \right)_w = \dot{q}_w''$

The velocity scale (u_*) and the non-dimensional RANS averaged cell temperature (T^+) are given based on the wall treatment of the turbulence model.

9.1.6 Wall Treatment for Turbulence

The low-wall y^+ treatment is suitable for low Reynolds-number flows. This is the preferred method if fine meshing is not considered computationally expensive. The governing equations require no modification if there are enough nodes to accurately capture the physics. To maintain this condition, the wall y^+ value must be less than or equal to 1. The high-wall y^+ treatment requires modifications to the governing equations when the flow falls in the viscous regime of the boundary layer. The high-wall method assumes the near wall cells lie in the log-law regime of the boundary layer and must have wall y^+ values greater than 30. This method does not capture boundary layer physics well but significantly reduces computational costs. The all-wall y^+ treatment is a combination of the other two techniques. When wall cells fall in the buffer regime of the boundary layer, a blending function calculates the turbulent quantities. The all-wall method works effectively when fine resolution at the wall is possible in some locations of the simulation, but course resolution is required in other locations of the simulation. The two-layer all-wall y^+ treatment is used for the $k-\varepsilon$ realizable model. The approach is identical to the all-wall

treatment, but specific values of ε are imposed at the centroids of the near-wall cells to provide consistency with the two-layer formulation.

Using the RANS turbulence models requires the wall treatment to provide boundary conditions to the flow, energy, and turbulence solvers in laminar and turbulent flow regimes. The RANS wall treatment calculates the wall shear stress (τ_w), then the wall heat flux (\dot{q}_w'') and imposes the turbulent quantities on the centroids of the near-wall cells.

$$\mu_\tau = \frac{u_*}{u^+} |\hat{\mathbf{v}}_{tan}| \quad \text{A.43}$$

The velocity scale (u_*) is given by Equation A.33 and the non-dimensional wall-tangential velocity component of the velocity vector (u^+) is give in Table A.4. The averaged tangential velocity vector ($\hat{\mathbf{v}}_{tan}$) is quantified from the RANS equations. The wall turbulent heat flux is calculated as a function of the non-dimensional RANS averaged cell temperature (T^+). The conditions applied to the transport equations for turbulence at the wall are shown in Table A.6.

Table A.6. Final conditions for turbulence models.

Condition	Description	Equations
Diffusivity ($\tilde{\nu}$)	Modified diffusivity at wall is 0. Near-wall cell modified diffusivity is based on imposed production.	$\tilde{\nu} = 0$ $\tilde{\nu} = f(P_{\tilde{\nu}})$
Turbulent kinetic energy (k)	Wall-normal velocity gradient is 0 at wall. Near-wall cell turbulent kinetic energy is based on imposed production (P_k) and ε .	$\left. \frac{\partial k}{\partial y} \right _w = 0$ $k = f(P_k, \varepsilon)$
Dissipation rates (ε and ω)	The turbulent (ε) and specific (ω) dissipation rates set the gradients at the wall equal to 0. Input values are imposed for ε and ω .	$\left. \frac{\partial \varepsilon}{\partial y} \right _w = 0$ $\left. \frac{\partial \omega}{\partial y} \right _w = 0$
Reynolds stress tensor (\mathbf{R})	The wall-normal velocity gradient is 0. The transport equation is solved using imposed strain rate tensor (\mathbf{S}) and ε .	$\left. \frac{\partial \mathbf{R}}{\partial y} \right _w = 0$ $\mathbf{R} = f(\mathbf{S}, \varepsilon)$

9.1.7 Wall Distance

Wall distance is a parameter that represents the distance from a cell centroid to the nearest wall face with a non-slip boundary condition. Various physical models require this

parameter to account for near-wall effects. STAR-CCM+ uses the implicit tree or partial differential equation (PDE) method. The implicit tree method calculates the exact wall distance to the centroid based on the mesh. The PDE method makes a close estimate of the wall distance based on a Poisson-type pure diffusion equation. The implicit tree method is more accurate and is used in this research.

B. APPENDIX

The equations and all results in this document were generated automatically from STAR-CCM+ or were obtained using python programming language. The following python functions were used to perform the calculations shown in this document.

```
import csv
import numpy
import matplotlib.pyplot
import scipy.integrate

def sortData(listdy, listtemp):
    temp = [j for _, j in sorted(zip(listdy, listtemp))]
    dy = sorted(listdy)
    return dy, temp

def moveAverage(listdy, listtemp, window_size):
    d = 0
    mov_average_dy = []
    while d < len(listdy) - window_size + 1:
        window = listdy[d : d + window_size]
        window_average = sum(window) / window_size
        mov_average_dy.append(window_average)
        d += 1

    c = 0
    mov_average_temp = []
    while c < len(listtemp) - window_size + 1:
        window_temp = listtemp[c : c + window_size]
        window_average_temp = sum(window_temp) / window_size
        mov_average_temp.append(window_average_temp)
```

```

c += 1

return mov_average_dy, mov_average_temp

#This function takes in the surface temperature file strings from simulation and puts the
data in lists.
#The lists are the y-axis values and corresponding temperatures.
#The lists are sorted and a moving average is applied based on the given criteria
(moving_avg_val).
#:param: csv_file: opened surface temp file
#     moving_avg_val: value for how many data values the moving average is calculated
over
#     name: the name for which to be used if plotting functions separately
#     plot: conditional operator for whether or not to print plots
#:return: list of averaged surface temperatures
#     [0]: downcomer fluid/shroud interface temperature
#     [1]: downcomer fluid/shroud interface corresponding axial location
#     [2]: downcomer fluid/atm interface temperature
#     [3]: downcomer fluid/atm interface corresponding axial location
#     [4]: riser fluid/shroud interface temperature
#     [5]: riser fluid/shroud interface corresponding axial location
#     [6]: riser fluid/rod interface temperature
#     [7]: riser fluid/rod interface corresponding axial location
def surfaceTemps(csv_file, moving_avg_val, name, plot):

    surface_temps = csv.reader(csv_file, delimiter=',')
    data = []
    for i in surface_temps:
        data.append(i)
    dc_fluid_shroud_temp = []
    dc_fluid_shroud_dy = []

```

```

r_fluid_shroud_temp = []
r_fluid_shroud_dy = []

r_fluid_rod_temp = []
r_fluid_rod_dy = []

dc_fluid_atm_temp = []
dc_fluid_atm_dy = []

for i in range(len(data)):
    try:
        dc_fluid_shroud_dy.append(abs(float(data[i][0]) - 2.35))
        dc_fluid_shroud_temp.append(float(data[i][1]))
    except:
        pass
    try:
        r_fluid_shroud_dy.append(abs(float(data[i][2]) - 2.35))
        r_fluid_shroud_temp.append(float(data[i][3]))
    except:
        pass
    try:
        dc_fluid_atm_dy.append(abs(float(data[i][4]) - 2.35))
        dc_fluid_atm_temp.append(float(data[i][5]))
    except:
        pass
    try:
        r_fluid_rod_dy.append(abs(float(data[i][6]) - 2.35))
        r_fluid_rod_temp.append(float(data[i][7]))
    except:
        pass

```

```
dc_fluid_shroud_dy, dc_fluid_shroud_temp = sortData(dc_fluid_shroud_dy,
dc_fluid_shroud_temp)
```

```
dc_fluid_shroud_dy, dc_fluid_shroud_temp = moveAverage(dc_fluid_shroud_dy,
dc_fluid_shroud_temp, moving_avg_val)
```

```
dc_fluid_atm_dy, dc_fluid_atm_temp = sortData(dc_fluid_atm_dy, dc_fluid_atm_temp)
dc_fluid_atm_dy, dc_fluid_atm_temp = moveAverage(dc_fluid_atm_dy,
dc_fluid_atm_temp, moving_avg_val)
```

```
r_fluid_shroud_dy, r_fluid_shroud_temp = sortData(r_fluid_shroud_dy,
r_fluid_shroud_temp)
```

```
r_fluid_shroud_dy, r_fluid_shroud_temp = moveAverage(r_fluid_shroud_dy,
r_fluid_shroud_temp, moving_avg_val)
```

```
r_fluid_rod_dy, r_fluid_rod_temp = sortData(r_fluid_rod_dy, r_fluid_rod_temp)
r_fluid_rod_dy, r_fluid_rod_temp = moveAverage(r_fluid_rod_dy, r_fluid_rod_temp,
moving_avg_val)
```

```
if plot == True:
```

```
fig = matplotlib.pyplot.gcf()
matplotlib.pyplot.plot(dc_fluid_atm_dy, dc_fluid_atm_temp, label='Fluid/Atm Temp')
matplotlib.pyplot.plot(dc_fluid_shroud_dy, dc_fluid_shroud_temp,
label='Fluid/Shroud (down)')
```

```
matplotlib.pyplot.plot(r_fluid_shroud_dy, r_fluid_shroud_temp, label='Fluid Shroud
(riser)')
```

```
matplotlib.pyplot.xlabel('dy (m)', fontsize=12)
```

```
matplotlib.pyplot.ylabel('temp (F)', fontsize=12)
```

```
matplotlib.pyplot.minorticks_on()
```

```
matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
```

```

matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)
fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\STAR\\'+str(name)+' surface temps.png', dpi=200)
matplotlib.pyplot.close()

fig = matplotlib.pyplot.gcf()
matplotlib.pyplot.plot(r_fluid_rod_dy, r_fluid_rod_temp, label='Fluid/Rod')
matplotlib.pyplot.xlabel('dy (m)', fontsize=12)
matplotlib.pyplot.ylabel('temp (F)', fontsize=12)
matplotlib.pyplot.minorticks_on()
matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)
fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\STAR\\'+str(name)+' rod surface temp.png',
dpi=200)
matplotlib.pyplot.close()

return_list = dc_fluid_shroud_temp, dc_fluid_shroud_dy, dc_fluid_atm_temp,
dc_fluid_atm_dy, r_fluid_shroud_temp, \
    r_fluid_shroud_dy, r_fluid_rod_temp, r_fluid_rod_dy

return return_list

#This function takes in the channel centerline temperature file strings from simulation and
puts the data in lists.
#The lists are the y-axis values and corresponding temperatures.

```


#The lists are sorted and a moving average is applied based on the given criteria (moving_avg_val).

#:param: csv_file_center: opened centerline temps file

moving_avg_val: value for how many data values the moving average is calculated over

name: the name for which to be used if plotting functions separately

plot: conditional operator (True/False) for whether or not to print plots

#:return: list of averaged centerline temperatures

[0]: downcomer centerline temperature

[1]: downcomer centerline corresponding axial location

[2]: downcomer choked section centerline temperature

[3]: downcomer choked section centerline corresponding axial location

[4]: riser centerline temperature

[5]: riser centerline corresponding axial location

[6]: riser heated section centerline temperature

[7]: riser heated section centerline corresponding axial location

def centerlineTemps(csv_file_center, moving_avg_val, name, plot):

center_temps = csv.reader(csv_file_center, delimiter=',')

data2 = []

for i in center_temps:

data2.append(i)

dc_center_temp = []

dc_center_dy = []

dc_center_choked_temp = []

dc_center_choked_dy = []

r_center_heat_temp = []

r_center_heat_dy = []

```

r_center_temp = []
r_center_dy = []

for i in range(len(data2)):
    try:
        dc_center_dy.append(abs(float(data2[i][0]) - 2.35))
        dc_center_temp.append(float(data2[i][1]))
    except:
        pass
    try:
        dc_center_choked_dy.append(abs(float(data2[i][2]) - 2.35))
        dc_center_choked_temp.append(float(data2[i][3]))
    except:
        pass
    try:
        r_center_heat_dy.append(abs(float(data2[i][4]) - 2.35))
        r_center_heat_temp.append(float(data2[i][5]))
    except:
        pass
    try:
        r_center_dy.append(abs(float(data2[i][6]) - 2.35))
        r_center_temp.append(float(data2[i][7]))
    except:
        pass

r_center_dy, r_center_temp = sortData(r_center_dy, r_center_temp)
r_center_dy, r_center_temp = moveAverage(r_center_dy, r_center_temp,
moving_avg_val)

```

```

r_center_heat_dy,      r_center_heat_temp      =      sortData(r_center_heat_dy,
r_center_heat_temp)

```

```

r_center_heat_dy,      r_center_heat_temp      =      moveAverage(r_center_heat_dy,
r_center_heat_temp, moving_avg_val)

```

```

dc_center_dy, dc_center_temp = sortData(dc_center_dy, dc_center_temp)
dc_center_dy,  dc_center_temp  =  moveAverage(dc_center_dy,  dc_center_temp,
moving_avg_val)

```

```

dc_center_choked_dy,  dc_center_choked_temp  =  sortData(dc_center_choked_dy,
dc_center_choked_temp)

```

```

dc_center_choked_dy, dc_center_choked_temp = moveAverage(dc_center_choked_dy,
dc_center_choked_temp, moving_avg_val)

```

```

if plot == True:
    fig = matplotlib.pyplot.gcf()
    matplotlib.pyplot.plot(dc_center_dy, dc_center_temp, label='Downcomer Center')
    matplotlib.pyplot.plot(dc_center_choked_dy,          dc_center_choked_temp,
label='Downcomer Center (Choked)')
    matplotlib.pyplot.plot(r_center_dy, r_center_temp, label='Riser Center')
    matplotlib.pyplot.plot(r_center_heat_dy,  r_center_heat_temp,  label='Riser  Center
(Choked)')
    matplotlib.pyplot.xlabel('dy (m)', fontsize=12)
    matplotlib.pyplot.ylabel('temp (F)', fontsize=12)
    matplotlib.pyplot.minorticks_on()
    matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
    matplotlib.pyplot.xticks(fontsize=12)
    matplotlib.pyplot.yticks(fontsize=12)
    matplotlib.pyplot.legend(fontsize=12)
    fig.set_size_inches(8, 4.5)

```

```

    matplotlib.pyplot.savefig('images\\STAR\\'+str(name)+'    centerline    temps.png',
dpi=200)
    matplotlib.pyplot.close()

```

```

    return_list    =    dc_center_temp,    dc_center_dy,    dc_center_choked_temp,
dc_center_choked_dy, r_center_temp, r_center_dy, \
        r_center_heat_temp, r_center_heat_dy

```

```

    return return_list

```

```

#This function takes in x and y data values and bounds them according to the given
#x-axis bounds.

```

```

#The lists are bounded and the x-axis reference (lower_bound) becomes zero

```

```

#:param: list_dy: axial data points

```

```

#    list_temp: corresponding temperatures

```

```

#    upper_bound: upper x-axis (dy distance for temps) bound

```

```

#    lower_bound: lower x-axis (dy distance for temps) bound

```

```

#:return: list of normalized short section where lower bound is now equal zero

```

```

#    list of corresponding temperatures

```

```

def heatedPortionLists(list_dy, list_temp, upper_bound, lower_bound):

```

```

    new_y = []

```

```

    new_temp = []

```

```

    for j in range(len(list_dy)):

```

```

        if list_dy[j] <= upper_bound and list_dy[j] >= lower_bound:

```

```

            new_y.append(list_dy[j] - lower_bound)

```

```

            new_temp.append(list_temp[j])

```

```

    return new_y, new_temp

```

```

#Calculates the change in temperature for some spatial profile (list_dy) and associated
temps (list_temp)

```

```

#The range_inlet and range_outlet are spatial ranges for the inlet and outlet, respectively.
#:param: list_dy: axial data points
#    list_temp: corresponding temperatures
#    upper_bound: two float number list, setting the range
#    lower_bound: two float number list, setting the range
#:return: float estimate of the change in temperature of the mean value of the temps within
the ranges.
def delTempHeated(list_dy, list_temp, range_inlet, range_outlet):
    out_temp_list = []
    in_temp_list = []
    for i in range(len(list_dy)):
        if list_dy[i] >= range_inlet[0] and list_dy[i] <= range_inlet[1]:
            in_temp_list.append(list_temp[i])
        elif list_dy[i] >= range_outlet[0] and list_dy[i] <= range_outlet[1]:
            out_temp_list.append(list_temp[i])
    delta_temp = numpy.mean(out_temp_list) - numpy.mean(in_temp_list)

    return delta_temp

#This function takes in the 3D cfd metric spreadsheets and returns lists of the values with
respect to time.
#The plot option allows choice to print the results
#:param: csv_file: opened csv metrics file
#    name: the name for which to be used if plotting functions separately
#    plot: conditional operator (True/False) for whether or not to print plots
#:return: list of averaged centerline temperatures
#    [0]: time in seconds
#    [1]: riser velocity in m/s
#    [2]: riser velocity in heated section in m/s
#    [3]: heat transfer calculated from rod in W
#    [4]: heat transfer calculated to atmosphere in W

```

```

#      [5]: average fluid temperature
#      [6]: downcomer mfr kg/s
#      [7]: riser mfr kg/s
#      [8]: heat transfer through plenum
def getMetrics(csv_file, name, plot):

    model_metrics = csv.reader(csv_file, delimiter=',')

    data = []
    for i in model_metrics:
        data.append(i)
    data.pop(0)
    time = []
    riser_vel = []
    riser_heat_vel = []
    rod_ht = []
    atm_ht = []
    avg_temp = []
    dc_mfr = []
    riser_mfr = []
    top_ht = []
    for i in data:
        time.append(float(i[0]))
        try:
            riser_vel.append(float(i[1]))
        except:
            riser_vel.append(riser_vel[-1])
        try:
            riser_heat_vel.append(float(i[2]))
        except:
            riser_heat_vel.append(riser_heat_vel[-1])

```

```

try:
    rod_ht.append(float(i[3]))
except:
    rod_ht.append(rod_ht[-1])
try:
    atm_ht.append(float(i[4]))
except:
    atm_ht.append(atm_ht[-1])
try:
    avg_temp.append(float(i[5]))
except:
    avg_temp.append(avg_temp[-1])
try:
    dc_mfr.append(float(i[6]))
except:
    dc_mfr.append(dc_mfr[-1])
try:
    riser_mfr.append(float(i[7]))
except:
    riser_mfr.append(riser_mfr[-1])
try:
    top_ht.append(float(i[8]))
except:
    top_ht.append(top_ht[-1])

if plot == True:
    fig = matplotlib.pyplot.gcf()
    matplotlib.pyplot.plot(time, rod_ht, label='Rod Heat')
    matplotlib.pyplot.plot(time, atm_ht, label='Atm Heat')
    matplotlib.pyplot.xlabel('time (s)', fontsize=12)
    matplotlib.pyplot.ylabel('heat transfer (W)', fontsize=12)

```

```

matplotlib.pyplot.minorticks_on()
matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)
fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\STAR\\'+str(name)+' rod and atm heat transfer.png',
dpi=200)
matplotlib.pyplot.close()

```

```

matplotlib.pyplot.plot(time, avg_temp, label='Avg Fluid Temp')
matplotlib.pyplot.xlabel('time (s)', fontsize=12)
matplotlib.pyplot.ylabel('temp (F)', fontsize=12)
matplotlib.pyplot.minorticks_on()
matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)
fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\STAR\\'+str(name)+' avg temp.png', dpi=200)
matplotlib.pyplot.close()

```

```

matplotlib.pyplot.plot(time, riser_vel, label='Riser Velocity Upper')
matplotlib.pyplot.plot(time, riser_heat_vel, label='Riser Velocity Heated')
matplotlib.pyplot.xlabel('time (s)', fontsize=12)
matplotlib.pyplot.ylabel('velocity (m/s)', fontsize=12)
matplotlib.pyplot.minorticks_on()
matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)

```



```

fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\STAR\\' + str(name) + ' velocity.png', dpi=200)
matplotlib.pyplot.close()

return_list = time, riser_vel, riser_heat_vel, rod_ht, atm_ht, \
    avg_temp, dc_mfr, riser_mfr, top_ht

return return_list

#Non-dimensionalizing function for a temperature list
#:param: temperature_list: temperature in any temperature to be put in terms of theta
#    tsat: saturation temperature
#    tinf: surrounding atmospheric temperature
#:return: list of non-dimensional temperature
def nonDimensionalTheta(temperature_list, tsat, tinf):
    theta = []
    for q in range(len(temperature_list)):
        theta.append((temperature_list[q] - tinf) / (tsat - tinf))
    return theta

#Non-dimensionalizing temperature uncertainty
#:param: temperature_list: temperature in any temperature to be put in terms of theta
#    u_list: associated uncertainty in the temperature list
#    tsat: saturation temperature
#    tinf: surrounding atmospheric temperature
#:return: list of non-dimensional temperature uncertainty
def nonDimensionalThetaU(temp_list, u_list, tsat, tinf):
    theta = []
    for q in range(len(temp_list)):
        t_err = (temp_list[q] + u_list[q] - tinf) / (tsat - tinf)
        t_meas = (temp_list[q] - tinf) / (tsat - tinf)

```

```

    theta.append(t_err - t_meas)
return theta

#normalizes a spatial list from 0 to 1
#version 1 is for if the initial value is not 0 but should be
#otherwise it normalizes with no spatial shift to 0, e.g. if initial value is already 0
#Non-dimensionalizing temperature uncertainty
#:param: dy: is the list to be normalized
#    version: 1 if initial value needs to be shifted to 0
#:return: list of normalizes spatial list
def normalDy(dy, version):
    if version == 1:
        list_min = min(dy)
        list_max = max(dy)
        for q in range(len(dy)):
            dy[q] = (dy[q] - list_min)/(list_max - list_min)
    else:
        list_max = version
        for q in range(len(dy)):
            dy[q] = dy[q] / list_max
    return dy

#This function takes in the SERTTA data and returns the values with respect to time.
#Default results can be plotted with the plot conditional
#:param: csv_file: the opened SERTTA data file
#    name: the name for which to be used if plotting functions separately
#    plot: conditional operator for whether or not to print plots
#:return: list of SERTTA data
#    [0]: time in seconds
#    [1]: power in watts
#    [2]: current in amperes

```

```

#      [3]: voltage in volts
#      [4]: tc_103 in F
#      [5]: tc_104 in F
#      [6]: tc_105 in F
#      [7]: tc_106 in F
#      [8]: tc_107 in F
#      [9]: ts_101 in F
#     [10]: ts_102 in F
#     [11]: ts_103 in F
#     [12]: ts_104 in F
#     [13]: tf_201 in F
#     [14]: tf_202 in F
#     [15]: tf_203 in F
#     [16]: tf_204 in F
#     [17]: tf_205 in F
#     [18]: ts_401 in F
#     [19]: ts_402 in F
#     [20]: ts_403 in F
#     [21]: ts_404 in F
#     [22]: ts_405 in F

def readSERTTA(csv_file, name, plot):

    read_sertta = csv.reader(csv_file, delimiter='\t')
    data = []
    for i in read_sertta:
        data.append(i)

    for i in range(len(data[0])):
        if str(data[0][i]) == 'Run_time':
            time_index = i
        elif str(data[0][i]) == 'CM-101':

```

```
    current_index = i
elif str(data[0][i]) == 'VM-101':
    voltage_index = i
elif str(data[0][i]) == 'TF-103':
    tc_103_index = i
elif str(data[0][i]) == 'TF-104':
    tc_104_index = i
elif str(data[0][i]) == 'TF-105':
    tc_105_index = i
elif str(data[0][i]) == 'TF-106':
    tc_106_index = i
elif str(data[0][i]) == 'TF-107':
    tc_107_index = i
elif str(data[0][i]) == 'TS-101':
    ts_101_index = i
elif str(data[0][i]) == 'TS-102':
    ts_102_index = i
elif str(data[0][i]) == 'TS-103':
    ts_103_index = i
elif str(data[0][i]) == 'TS-104':
    ts_104_index = i
elif str(data[0][i]) == 'TF-201':
    tf_201_index = i
elif str(data[0][i]) == 'TF-202':
    tf_202_index = i
elif str(data[0][i]) == 'TF-203':
    tf_203_index = i
elif str(data[0][i]) == 'TF-204':
    tf_204_index = i
elif str(data[0][i]) == 'TF-205':
    tf_205_index = i
```

```

elif str(data[0][i]) == 'TS-401':
    ts_401_index = i
elif str(data[0][i]) == 'TS-402':
    ts_402_index = i
elif str(data[0][i]) == 'TS-403':
    ts_403_index = i
elif str(data[0][i]) == 'TS-404':
    ts_404_index = i
elif str(data[0][i]) == 'TS-405':
    ts_405_index = i
elif str(data[0][i]) == 'TF-602':
    tf_602_index = i
elif str(data[0][i]) == 'TS-601':
    tf_601_index = i

```

```

time_seconds = []
power_watts = []
current_amps = []
voltage_volts = []
tc_103 = []
tc_104 = []
tc_105 = []
tc_106 = []
tc_107 = []
ts_101 = []
ts_102 = []
ts_103 = []
ts_104 = []
tf_201 = []
tf_202 = []

```

```
tf_203 = []
tf_204 = []
tf_205 = []
ts_401 = []
ts_402 = []
ts_403 = []
ts_404 = []
ts_405 = []
tf_601 = []
tf_602 = []

for i in data:
    try:
        time_seconds.append(float(i[time_index]))
        power_watts.append(float(i[voltage_index]) * float(i[current_index]))
        current_amps.append(float(i[current_index]))
        voltage_volts.append(float(i[voltage_index]))
        tc_103.append(float(i[tc_103_index]))
        tc_104.append(float(i[tc_104_index]))
        tc_105.append(float(i[tc_105_index]))
        tc_106.append(float(i[tc_106_index]))
        tc_107.append(float(i[tc_107_index]))
        ts_101.append(float(i[ts_101_index]))
        ts_102.append(float(i[ts_102_index]))
        ts_103.append(float(i[ts_103_index]))
        ts_104.append(float(i[ts_104_index]))
        tf_201.append(float(i[tf_201_index]))
        tf_202.append(float(i[tf_202_index]))
        tf_203.append(float(i[tf_203_index]))
        tf_204.append(float(i[tf_204_index]))
        tf_205.append(float(i[tf_205_index]))
```

```

ts_401.append(float(i[ts_401_index]))
ts_402.append(float(i[ts_402_index]))
ts_403.append(float(i[ts_403_index]))
ts_404.append(float(i[ts_404_index]))
ts_405.append(float(i[ts_405_index]))
tf_601.append(float(i[tf_601_index]))
tf_602.append(float(i[tf_602_index]))
except:
    pass

if plot == True:
    fig = matplotlib.pyplot.gcf()
    matplotlib.pyplot.plot(time_seconds, power_watts, label='Power')
    matplotlib.pyplot.xlabel('time (s)', fontsize=12)
    matplotlib.pyplot.ylabel('power (W)', fontsize=12)
    matplotlib.pyplot.minorticks_on()
    matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
    matplotlib.pyplot.xticks(fontsize=12)
    matplotlib.pyplot.yticks(fontsize=12)
    matplotlib.pyplot.legend(fontsize=12)
    matplotlib.pyplot.show()
    fig.set_size_inches(8, 4.5)
    matplotlib.pyplot.savefig('images\\SERTTA\\' + str(name) + 'power.png', dpi=200)
    matplotlib.pyplot.close()

fig = matplotlib.pyplot.gcf()
matplotlib.pyplot.plot(time_seconds, tc_103, label='TF 103')
matplotlib.pyplot.plot(time_seconds, tc_104, label='TF 104')
matplotlib.pyplot.plot(time_seconds, tc_105, label='TF 105')
matplotlib.pyplot.plot(time_seconds, tc_106, label='TF 106')
matplotlib.pyplot.plot(time_seconds, tc_107, label='TF 107')

```

```

matplotlib.pyplot.xlabel('time (s)', fontsize=12)
matplotlib.pyplot.ylabel('temp (F)', fontsize=12)
matplotlib.pyplot.minorticks_on()
matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)
fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\SERTTA\\' + str(name) + 'primary
thermocouples.png', dpi=200)
matplotlib.pyplot.close()

```

```

fig = matplotlib.pyplot.gcf()
matplotlib.pyplot.plot(time_seconds, ts_101, label='TS 104')
matplotlib.pyplot.plot(time_seconds, ts_102, label='TS 105')
matplotlib.pyplot.plot(time_seconds, ts_103, label='TS 103')
matplotlib.pyplot.plot(time_seconds, ts_104, label='TS 104')
matplotlib.pyplot.xlabel('time (s)', fontsize=12)
matplotlib.pyplot.ylabel('temp (F)', fontsize=12)
matplotlib.pyplot.minorticks_on()
matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)
fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\SERTTA\\' + str(name) + 'heater rod
thermocouples.png', dpi=200)
matplotlib.pyplot.close()

```

```

fig = matplotlib.pyplot.gcf()
matplotlib.pyplot.plot(time_seconds, ts_401, label='TS 401')

```



```

matplotlib.pyplot.plot(time_seconds, ts_402, label='TS 402')
matplotlib.pyplot.plot(time_seconds, ts_403, label='TS 403')
matplotlib.pyplot.plot(time_seconds, ts_404, label='TS 404')
matplotlib.pyplot.plot(time_seconds, ts_405, label='TS 405')
matplotlib.pyplot.plot(time_seconds, tf_201, label='TF 201')
matplotlib.pyplot.plot(time_seconds, tf_202, label='TF 202')
matplotlib.pyplot.plot(time_seconds, tf_203, label='TF 203')
matplotlib.pyplot.plot(time_seconds, tf_204, label='TF 204')
matplotlib.pyplot.plot(time_seconds, tf_205, label='TF 205')
matplotlib.pyplot.xlabel('time (s)', fontsize=12)
matplotlib.pyplot.ylabel('temp (F)', fontsize=12)
matplotlib.pyplot.minorticks_on()
matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)
fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\SERTTA\\' + str(name) + 'tank and shroud
thermocouples.png', dpi=200)
matplotlib.pyplot.close()

return_list = time_seconds, power_watts, current_amps, voltage_volts, tc_103, tc_104,
tc_105, tc_106, tc_107, ts_101, \
    ts_102, ts_103, ts_104, tf_201, tf_202, tf_203, tf_204, tf_205, ts_401, ts_402, ts_403,
ts_404, ts_405, tf_601, tf_602

return return_list

#Takes in the SERTTA data returned from readSERTTA function to get steady state
temperatures and standard
#deviations of the steady state values for experimental uncertainty.

```

#:param: ss_time: a list of lists where each individual list is a range of steady state operational time

data: the list returned from readSERTTA function which can be seen in description

#:return: list of SERTTA data

[0]: rod thermocouple steady state average values in F

[1]: wall fluid thermocouple steady state average values in F

[2]: outer atmospheric surface thermocouple steady state average values in F

[3]: rod thermocouple steady state average value standard deviation in F

[4]: wall fluid thermocouple steady state average value standard deviation in F

[5]: outer atmospheric surface thermocouple steady state average value standard deviation in F

def steadyTemps(ss_time, data):

data_time = data[0]

ts1 = data[9]

ts2 = data[10]

ts3 = data[11]

ts4 = data[12]

tc3 = data[4]

tc4 = data[5]

tc5 = data[6]

tc6 = data[7]

tc7 = data[8]

ts41 = data[18]

ts42 = data[19]

ts43 = data[20]

ts44 = data[21]

ts45 = data[22]

ts61 = data[23]

ts62 = data[24]

ts_rod_steady_array = [], [], [], []

```

tf_wall_steady_array = [], [], [], [], []
ts_atm_steady_array = [], [], [], [], []
tf_coils_array = [], []
for i in range(len(data_time)):
    for q in range(len(ss_time)):
        if data_time[i] >= ss_time[q][0] and data_time[i] <= ss_time[q][1]:
            ts_rod_steady_array[0].append(ts1[i])
            ts_rod_steady_array[1].append(ts2[i])
            ts_rod_steady_array[2].append(ts3[i])
            ts_rod_steady_array[3].append(ts4[i])

            tf_wall_steady_array[0].append(tc3[i])
            tf_wall_steady_array[1].append(tc4[i])
            tf_wall_steady_array[2].append(tc5[i])
            tf_wall_steady_array[3].append(tc6[i])
            tf_wall_steady_array[4].append(tc7[i])

            ts_atm_steady_array[0].append(ts41[i])
            ts_atm_steady_array[1].append(ts42[i])
            ts_atm_steady_array[2].append(ts43[i])
            ts_atm_steady_array[3].append(ts44[i])
            ts_atm_steady_array[4].append(ts45[i])

            tf_coils_array[0].append(ts61[i])
            tf_coils_array[1].append(ts62[i])

ts_rod_steady_avg = []
tf_wall_steady_avg = []
ts_atm_steady_avg = []

ts_rod_std_dev = []

```

```
tf_wall_std_dev = []
```

```
ts_atm_std_dev = []
```

```
ts_coils_steady_avg = []
```

```
ts_coils_steady_avg.append(numpy.mean(tf_coils_array[0]))
```

```
ts_coils_steady_avg.append(numpy.mean(tf_coils_array[1]))
```

```
ts_rod_steady_avg.append(numpy.mean(ts_rod_steady_array[0]))
```

```
ts_rod_steady_avg.append(numpy.mean(ts_rod_steady_array[1]))
```

```
ts_rod_steady_avg.append(numpy.mean(ts_rod_steady_array[2]))
```

```
ts_rod_steady_avg.append(numpy.mean(ts_rod_steady_array[3]))
```

```
ts_rod_std_dev.append(numpy.std(ts_rod_steady_array[0]))
```

```
ts_rod_std_dev.append(numpy.std(ts_rod_steady_array[1]))
```

```
ts_rod_std_dev.append(numpy.std(ts_rod_steady_array[2]))
```

```
ts_rod_std_dev.append(numpy.std(ts_rod_steady_array[3]))
```

```
tf_wall_steady_avg.append(numpy.mean(tf_wall_steady_array[0]))
```

```
tf_wall_steady_avg.append(numpy.mean(tf_wall_steady_array[1]))
```

```
tf_wall_steady_avg.append(numpy.mean(tf_wall_steady_array[2]))
```

```
tf_wall_steady_avg.append(numpy.mean(tf_wall_steady_array[3]))
```

```
tf_wall_steady_avg.append(numpy.mean(tf_wall_steady_array[4]))
```

```
tf_wall_std_dev.append(numpy.std(tf_wall_steady_array[0]))
```

```
tf_wall_std_dev.append(numpy.std(tf_wall_steady_array[1]))
```

```
tf_wall_std_dev.append(numpy.std(tf_wall_steady_array[2]))
```

```
tf_wall_std_dev.append(numpy.std(tf_wall_steady_array[3]))
```

```
tf_wall_std_dev.append(numpy.std(tf_wall_steady_array[4]))
```

```
ts_atm_steady_avg.append(numpy.mean(ts_atm_steady_array[0]))
```

```
ts_atm_steady_avg.append(numpy.mean(ts_atm_steady_array[1]))
```

```

ts_atm_steady_avg.append(numpy.mean(ts_atm_steady_array[2]))
ts_atm_steady_avg.append(numpy.mean(ts_atm_steady_array[3]))
ts_atm_steady_avg.append(numpy.mean(ts_atm_steady_array[4]))

```

```

ts_atm_std_dev.append(numpy.std(ts_atm_steady_array[0]))
ts_atm_std_dev.append(numpy.std(ts_atm_steady_array[1]))
ts_atm_std_dev.append(numpy.std(ts_atm_steady_array[2]))
ts_atm_std_dev.append(numpy.std(ts_atm_steady_array[3]))
ts_atm_std_dev.append(numpy.std(ts_atm_steady_array[4]))

```

```

return_list = ts_rod_steady_avg, tf_wall_steady_avg, ts_atm_steady_avg, \
    ts_rod_std_dev, tf_wall_std_dev, ts_atm_std_dev, ts_coils_steady_avg

```

```

return return_list

```

#Takes in the SERTTA data returned from readSERTTA function to get steady state power and standard

#deviations of the steady state power value for experimental uncertainty.

#:param: ss_time: a list of lists where each individual list is a range of steady state operational time

data: the list returned from readSERTTA function which can be seen in description

current_u_b: current systematic uncertainty

voltage_u_b: voltage systematic uncertainty

#:return: float of average power value and the experimental uncertainty

```

def steadyPower(ss_time, data, current_u_b, voltage_u_b):

```

```

    data_time = data[0]

```

```

    power = data[1]

```

```

    current = data[2]

```

```

    voltage = data[3]

```

```

power_list = []
current_list = []
voltage_list = []
for i in range(len(data_time)):
    for q in range(len(ss_time)):
        if data_time[i] >= ss_time[q][0] and data_time[i] <= ss_time[q][1]:
            power_list.append(power[i])
            current_list.append(current[i])
            voltage_list.append(voltage[i])

power_avg = numpy.mean(power_list)
current_avg = numpy.mean(current_list)
current_std = numpy.std(current_list)
voltage_avg = numpy.mean(voltage_list)
voltage_std = numpy.std(voltage_list)

s_squared = (current_avg * voltage_std)**2 + (voltage_avg * current_std)**2
b_squared = (current_avg * voltage_u_b)**2 + (voltage_avg * current_u_b)**2

u_d = numpy.sqrt(s_squared + b_squared)

return power_avg, u_d

#Takes in x and y data set and gets linear regression intercept and slope
#param: x: the x data
#      y: the y data
#return: y-intercept and the line slope
def linearRegression(x,y):
    n = len(x)
    sum_y = 0
    sum_x2 = 0

```

```

sum_x = 0
sum_xy = 0
for i in range(len(x)):
    sum_y += y[i]
    sum_x2 += x[i]**2
    sum_x += x[i]
    sum_xy += x[i]*y[i]

```

```

a_int = (sum_y * sum_x2 - sum_x * sum_xy) / (n*sum_x2 - sum_x**2)
b_slope = (n*sum_xy - sum_x*sum_y) / (n*sum_x2 - sum_x**2)
return a_int, b_slope

```

```

#Takes in thermocouple measurements and associated uncertainties to return
#the experimental uncertainty values in a list
#:param: list_thermocouples: the thermocouple temperature measurements
#    uncertainty_s: systematic uncertainty
#    uncertainty_b: random uncertainty
#:return: list of experimental uncertainty
def thermocoupleUncertainty(list_thermocouples, uncertainty_s, uncertainty_b):
    uncertainty_d_list = []
    for q in range(len(list_thermocouples)):
        u_d = numpy.sqrt((uncertainty_b**2) + (uncertainty_s[q]**2))
        uncertainty_d_list.append(u_d)

    return uncertainty_d_list

```

```

#This function takes in the LUNA data and returns the tared raw data, time, specific
locations vs time, axial location
#:param: tsv_file: the opened LUNA data file
#    name: the name for which to be used if plotting functions separately
#    plot: conditional operator for whether or not to print plots

```

```

#    plenum_location: location of gas plenum for riser or downcomer luna
#:return: data: just the strain measurements where time is first index and second index is
location
#    time: time in seconds
#    location_plot: five different locations
#    dy_axis: the axial locations corresponding to second index in data
def readTSV(tsv_file, name, plot, plenum_location):

    file = csv.reader(tsv_file, delimiter='\t')
    data = []
    for row in file:
        data.append(row)

    data_collect_row = 0
    for y in range(len(data)):
        try:
            if data[y-1][0] == 'x-axis':
                data_collect_row = y
        except:
            pass

    tare_strain = data[data_collect_row - 2]

    dy_axis = data[data_collect_row - 1]

    #these locations are taken as parallel to the thermocouples and the end
    for j in range(len(dy_axis)):
        try:
            if float(dy_axis[j]) == 0.4622:
                index1 = j
            elif float(dy_axis[j]) == 1.055:

```



```

        index2 = j
    elif float(dy_axis[j]) == 1.6166:
        index3 = j
    elif float(dy_axis[j]) == 2.1756:
        index4 = j
    elif float(dy_axis[j]) == 2.2536:
        index5 = j
    except:
        pass

date_initial = data[data_collect_row][0].split()
time_initial = date_initial[1].split(':')
seconds_inital = float(time_initial[0]) * 3600 + float(time_initial[1]) * 60 +
float(time_initial[2])
time_plot = []
location_plot = [], [], [], [], []
for w in range(len(data) - data_collect_row):
    w_ind = w + data_collect_row
    date = data[w_ind][0].split()
    time = date[1].split(':')
    seconds = float(time[0]) * 3600 + float(time[1]) * 60 + float(time[2])
    time_plot.append(seconds - seconds_inital)
    location_plot[0].append(float(data[w_ind][index1]) - float(tare_strain[index1]))
    location_plot[1].append(float(data[w_ind][index2]) - float(tare_strain[index2]))
    location_plot[2].append(float(data[w_ind][index3]) - float(tare_strain[index3]))
    location_plot[3].append(float(data[w_ind][index4]) - float(tare_strain[index4]))
    location_plot[4].append(float(data[w_ind][index5]) - float(tare_strain[index5]))

del dy_axis[0:3]
del tare_strain[0:3]

```

```

for t in range(len(dy_axis)):
    dy_axis[t] = float(dy_axis[t])

del data[0:data_collect_row]
for r in range(len(data)):
    del data[r][0:3]
    for q in range(len(data[r])):
        data[r][q] = float(data[r][q]) - float(tare_strain[q])

if plot == True:
    fig = matplotlib.pyplot.gcf()
    matplotlib.pyplot.plot(time_plot, location_plot[0], color='lime', label='dy = 0 m')
    matplotlib.pyplot.plot(time_plot, location_plot[1], color='cyan',
        label='dy = ' + str((1.055-plenum_location).__round__(3)) + ' m')
    matplotlib.pyplot.plot(time_plot, location_plot[2], color='cornflowerblue',
        label='dy = ' + str((1.6166-plenum_location).__round__(3)) + ' m')
    matplotlib.pyplot.plot(time_plot, location_plot[3], color='b',
        label='dy = ' + str((2.1756-plenum_location).__round__(3)) + ' m')
    matplotlib.pyplot.plot(time_plot, location_plot[4], color='midnightblue',
        label='dy = ' + str((2.2536-plenum_location).__round__(3)) + ' m')
    matplotlib.pyplot.xlabel('time (s)', fontsize=12)
    matplotlib.pyplot.ylabel("r'$\mu$-strain", fontsize=12)
    matplotlib.pyplot.minorticks_on()
    matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
    matplotlib.pyplot.xticks(fontsize=12)
    matplotlib.pyplot.yticks(fontsize=12)
    matplotlib.pyplot.legend(fontsize=12)
    fig.set_size_inches(8, 4.5)
    matplotlib.pyplot.savefig('images\\LUNA\\' + str(name) + '.png', dpi=200)
    matplotlib.pyplot.close()

```

```

return data, time_plot, location_plot, dy_axis

#This function takes in the two LUNA data and time with hard coded SS windows to get
avg LUNA temps and uncertainty
#:param: data_heat: heat up data after calibration
#      data_pulse: pulse data after calibration
#      respective time plots
#:return: avg_temps: list of LUNA avg temps
#      luna_u_d: experimental uncertainty
def lunaSteadyVals(data_heat, data_pulse, time_heat, time_pulse, luna_dy, luna_u_b,
cal_u_s, intercpt, slope):
    avg_temps = []
    std_temps = []
    for q in range(len(luna_dy)):
        dy_q_strain = []
        for j in range(len(data_heat)):
            if time_heat[j] >= 1850 and time_heat[j] <= 2250:
                dy_q_strain.append(data_heat[j][q])
            elif time_heat[j] >= 3400 and time_heat[j] <= 3580:
                dy_q_strain.append(data_heat[j][q])
        for g in range(len(data_pulse)):
            if time_pulse[g] >= 1150 and time_pulse[g] <= 1450:
                dy_q_strain.append(data_pulse[g][q])

        avg_temps.append(numpy.mean(dy_q_strain))
        std_temps.append(numpy.std(dy_q_strain))

    luna_u_d = []
    for p in range(len(std_temps)):
        u_s_sqr = (slope * std_temps[p])**2 + (avg_temps[p] * cal_u_s)**2
        u_b_sqr = (slope * avg_temps[p] * luna_u_b / 100)**2

```

```

u_d = numpy.sqrt(u_s_sqr + u_b_sqr)
luna_u_d.append(u_d)

for r in range(len(avg_temps)):
    avg_temps[r] = avg_temps[r] * slope + intercpt

return avg_temps, luna_u_d

#This function takes in the two LUNA data and time with corresponding SERTTA data to
get calibration from TC-103
#:param: luna_strain: luna data strain in downcomer
#    luna_time: time list of LUNA data
#    sertta_temp: temp reading from TC-103
#    sertta_time: time of TC-103
#:return: intercept: calibration y-intercept
#    slope: calibration slope
#    cal_std_dev: calibration std deviation for uncertainty
def calibrateLuna(luna_strain, luna_time, sertta_temp, sertta_time):
    # time clipping to merge the sertta data timeline onto the LUNA fiber timeline
    sertta_time_clip = []
    sertta_temp_clip = []
    for i in range(len(sertta_time)):
        if sertta_time[i] <= 3590 + 233 and sertta_time[i] >= 233:
            sertta_time_clip.append(sertta_time[i] - 233)
            sertta_temp_clip.append(sertta_temp[i])

    # converging the time sampling rates because SERTTA = 2Hz and LUNA = 2.25Hz
    luna_new_time = []
    luna_new_strain = []
    for i in range(len(sertta_time_clip)):
        smallest_difference = 10

```

```

index = 0
for j in range(len(luna_time)):
    if abs(luna_time[j] - sertta_time_clip[i]) <= smallest_difference:
        smallest_difference = abs(luna_time[j] - sertta_time_clip[i])
        index = j
luna_new_time.append(luna_time[index])
luna_new_strain.append(luna_strain[index])

# The intercept and slope for calculating temperature from strain
intercept, slope = linearRegression(luna_new_strain, sertta_temp_clip)
x = []
y = []
cal_diff = []
for i in range(len(luna_new_strain)):
    x_val = luna_new_strain[i]
    x.append(x_val)
    y.append(intercept + slope * x_val)
    cal_diff.append(abs((intercept + slope * x_val)/sertta_temp_clip[i]))

cal_std_dev = abs(numpy.mean(cal_diff) - 1)

luna_temp_plot = []
for l in range(len(luna_time)):
    luna_temp_plot.append(luna_strain[l]*slope + intercept)

fig = matplotlib.pyplot.gcf()
matplotlib.pyplot.plot(luna_new_strain, sertta_temp_clip, label='Raw Data')
matplotlib.pyplot.plot(x, y, label='Linear Regression')
matplotlib.pyplot.xlabel('LUNA 'r'$\mu$-strain', fontsize=12)
matplotlib.pyplot.ylabel('TF-103 (F)', fontsize=12)
matplotlib.pyplot.minorticks_on()

```

```

matplotlib.pyplot.grid(b=True, which='both', axis='both', linestyle=':')
matplotlib.pyplot.xticks(fontsize=12)
matplotlib.pyplot.yticks(fontsize=12)
matplotlib.pyplot.legend(fontsize=12)
fig.set_size_inches(8, 4.5)
matplotlib.pyplot.savefig('images\\LUNA\\luna calibration.png', dpi=200)
matplotlib.pyplot.close()

fig, axs = matplotlib.pyplot.subplots(1, 2, sharey=True)
axs[0].plot(luna_new_strain, sertta_temp_clip, label='TF-103')
axs[0].plot(x, y, label='Linear Regression')
axs[0].minorticks_on()
axs[0].grid(b=True, which='both', axis='both', linestyle=':')
axs[0].legend(fontsize=14)
axs[0].set_xlabel('LUNA 'r'$\mu$-strain', fontsize=14)
axs[0].tick_params(axis='x', labels=14)
axs[0].set_ylabel('Temp (F)', fontsize=14)
axs[0].tick_params(axis='y', labels=14)
axs[1].plot(sertta_time_clip, sertta_temp_clip, label='TF-103')
axs[1].plot(luna_time, luna_temp_plot, label='LUNA TF-103 Location',
color='slategray', linestyle='--')
axs[1].minorticks_on()
axs[1].grid(b=True, which='both', axis='both', linestyle=':')
axs[1].legend(fontsize=14)
axs[1].set_xlabel('Time (s)', fontsize=14)
axs[1].tick_params(axis='x', labels=14)
matplotlib.pyplot.subplots_adjust(wspace=0, hspace=0)
fig.set_size_inches(13, 5)
matplotlib.pyplot.savefig('images\\LUNA\\luna strain correction.png', dpi=200)
matplotlib.pyplot.close()

```

```
return intercept, slope, cal_std_dev
```

```
#This function takes CFD temp data and experimental temp data to calculate the error
```

```
#:param: dy_cfd: axial plot of cfd temps
```

```
#    temp_cfd: cfd temps
```

```
#    dy_exp: experimental axial location
```

```
#    dy_exp_ud: experimental axial location uncertainty
```

```
#:return: cfd_epsilon: the list of error
```

```
def epsilonTC(dy_cfd, temp_cfd, dy_exp, dy_exp_ud):
```

```
    cfd_epsilon = []
```

```
    for j in range(len(dy_exp)):
```

```
        sum_temps = []
```

```
        for q in range(len(dy_cfd)):
```

```
            if dy_cfd[q] <= dy_exp[j] + dy_exp_ud[j] and dy_cfd[q] >= dy_exp[j] - dy_exp_ud[j]:
```

```
                sum_temps.append(temp_cfd[q])
```

```
        cfd_epsilon.append(numpy.mean(sum_temps))
```

```
    return cfd_epsilon
```

```
#This function takes CFD temp data and experimental LUNA data to calculate the error
```

```
#:param: dy_cfd: axial plot of cfd temps
```

```
#    temp_cfd: cfd temps
```

```
#    dy_exp: experimental luna axial location
```

```
#:return: cfd_epsilon: the list of error
```

```
def epsilonLuna(dy_cfd, temp_cfd, dy_exp):
```

```
    cfd_epsilon = []
```

```
    for k in range(len(dy_exp)):
```

```
        sum_temps = []
```

```
        for j in range(len(dy_cfd)):
```

```
            if k == 0:
```

```
                dy = (dy_exp[k + 1] - dy_exp[k])
```

```
                if dy_cfd[j] <= dy_exp[k] + dy:
```

```

        sum_temps.append(temp_cfd[j])
    elif k == len(dy_exp) - 1:
        dy = (dy_exp[k] - dy_exp[k - 1]) / 2
        if dy_cfd[j] <= dy_exp[k] + dy and dy_cfd[j] >= dy_exp[k] - dy:
            sum_temps.append(temp_cfd[j])
        else:
            dy = ((dy_exp[k + 1] - dy_exp[k]) / 2 + (dy_exp[k] - dy_exp[k - 1]) / 2) / 2
            if dy_cfd[j] <= dy_exp[k] + dy and dy_cfd[j] >= dy_exp[k] - dy:
                sum_temps.append(temp_cfd[j])
    cfd_epsilon.append(numpy.mean(sum_temps))
return cfd_epsilon

def fahrToKelvin(temp):
    kelvin_temp = (temp - 32) * 5 / 9 + 273.15
    return kelvin_temp

#This function takes in two lists and converges them based on the criteria given
#param: dy: axial plot
#    temp: temp plot
#    delta_y: window to converge to axially
#return: new_dy: new axial plot
#    new_temp: new temp plot
def convergeList(dy, temp, delta_y):
    bounder = delta_y/2
    yval = dy[0]
    list_delta = []
    new_dy = []
    new_temp = []
    for q in range(len(dy)):
        if dy[q] < yval + bounder and dy[q] >= yval - bounder:
            list_delta.append(temp[q])

```



```

else:
    list_delta.append(temp[q])

    new_dy.append(yval)
    new_temp.append(numpy.mean(list_delta))

    yval += delta_y
    list_delta.clear()

return new_dy, new_temp

#This function takes in gci data to get numerical uncertainty based on gci study
#:param: xaxis: the x-value corresponding to temp converged for each temp list
#    temp(3,2,1): temp plot of course to fine mesh values
#    r_(21, 32): mesh ratios
#:return: u_num: numerical uncertainty over xaxis in Fahrenheit
def numUncertain(xaxis, temp3, temp2, temp1, r_21, r_32):
    u_num = []
    u_num_ret = []
    extrapolated = []
    for q in range(len(xaxis)):
        phi3 = temp3[q]
        phi2 = temp2[q]
        phi1 = temp1[q]
        e_32 = phi3 - phi2
        e_21 = phi2 - phi1
        ea_32 = abs(e_32)
        ea_21 = abs(e_21)

        sign_val = (e_32/e_21) / abs(e_32/e_21)
        converged = False

```

```

conv_criteria = 1*10**-10
p_init = [1, 1]
while converged == False:
    q_p = numpy.log(((r_21**p_init[0]) - sign_val)/((r_32**p_init[0]) - sign_val))
    p_init[1] = (1/numpy.log(r_21)) * (numpy.log(abs(e_32/e_21)) + q_p)
    if abs(p_init[1] - p_init[0]) <= conv_criteria:
        converged = True
        p_val = p_init[1]

    p_init[0] = p_init[1]

# if p_val <= 1.0:
#   p_val = abs(p_val)

# print(p_val)
u_num.append(abs(p_val))

# print('q(p)= ' + str(q_p))
# print('s_val= ' + str(sign_val))
p_fin = abs(numpy.mean(u_num))
print('p_final= ' + str(p_fin))

for q in range(len(xaxis)):
    phi3 = temp3[q]
    phi2 = temp2[q]
    phi1 = temp1[q]
    e_32 = phi3 - phi2
    e_21 = phi2 - phi1
    ea_32 = abs(e_32)

```

```

ea_21 = abs(e_21)
# if xaxis[q] <= 1.94:
#   p_fin_use = p_fin
# else:
#   p_fin_use = p_fin
#gci_21 = abs(1.25 * ea_21 / (r_21**p_fin - 1))
gci_21 = abs(ea_21 / (r_21 ** p_fin - 1))
#gci_32 = abs(ea_32 / (r_32**p_fin - 1))
#u_num_ret.append((gci_21 + gci_32)/2)
u_num_ret.append(gci_21)

extrapolated.append(((r_21 ** p_fin) * phi1 - phi2) / (r_21 ** p_fin - 1))
# print('ea21= ' +str(ea_21))
# print('ea32= ' +str(ea_32))
# print('extrap sol= ' + str(extrapolated[len(extrapolated)-1]))
# print('gci21= ' +str(gci_21))
# print('gci32= ' +str(gci_32))

print('max num_unc (theta): ' + str(max(u_num_ret)))
return u_num_ret, extrapolated

#This function takes in two lists and expands them to the x_axis vals
#:param: avg_val: the average list of temp, or other value
#   avg_xaxis: the average x-axis list
#   list_xaxis: axis to be expanded to
#:return: new_xaxis: new axis expanded to list_xaxis
#   new_val: expanded temps or other vals
def expandAvg(avg_val, avg_xaxis, list_xaxis):
    new_val = []
    new_xaxis = []
    for q in range(len(list_xaxis)):

```

```

difference = 10
for r in range(len(avg_xaxis)):
    if abs(list_xaxis[q] - avg_xaxis[r]) <= difference:
        difference = abs(list_xaxis[q] - avg_xaxis[r])
        index = r
new_val.append(avg_val[index])
new_xaxis.append(avg_xaxis[index])

return new_xaxis, new_val

#This function takes in two lists and expands them to the x_axis vals
#:param: num: numerical values
#    num_ud: numerical uncertainty
#    exp: experimental values
#    exp_ud: experimental uncertainty
#:return: error_v: validation error
#    error_ud: validation error uncertainty
def validationMetrics(num, num_ud, exp, exp_ud):
    error_v = []
    error_ud = []
    for q in range(len(num)):
        error_v.append(abs(num[q] - exp[q]))
        error_ud.append(numpy.sqrt((num_ud[q]**2) + (exp_ud[q]**2)))
    return error_v, error_ud

#This function calculates the velocity of pulses based on center of heat method and returns
the
#:param: data: the tared LUNA data
#    time: the LUNA time
#    dy: the axial location
#    window: the window where pulses are occuring

```

```

# [p][0]: beginning of pulse integral
# [p][4]: end of pulse integral
# [p][2]: beginning of converged correction
# [p][3]: end of converged correction
# [p][4]: clip criteria for LUNA near plenum where pulse is not apparent
# [p][5]: the total input power
#:return: power_input: the input power list
# velocity: respective velocity estimates from input power
def centerOfHeat(data, time, dy, window, gas_y):

    power_input = []
    velocity = []
    for p in range(len(window)):
        time_center_pulse1 = []
        dx_center_pulse1 = []
        for i in range(len(dy)):
            if float(dy[i]) >= gas_y:
                temp = []
                time_of = []
                integral = []
                counter = 0
                for j in range(len(time)):
                    if time[j] >= window[p][0] and time[j] <= window[p][1]:
                        if counter == 0:
                            initial = float(data[j][i])
                            counter = 1
                        temp.append(float(data[j][i]) - initial)
                        time_of.append(time[j])
                        integral.append(scipy.integrate.trapz(temp, time_of))

                for l in range(len(integral)):

```

```

    half_integral = integral[len(integral) - 1] / 2
    d_i = abs(half_integral - integral[1])
    min_d_i = 10
    if d_i <= min_d_i:
        min_d_i = d_i
        pulse_index = 1
    dx_center_pulse1.append(float(dy[i]))
    time_center_pulse1.append(time_of[pulse_index])

time_center_pulse = []
dx_center_pulse = []
for q in range(len(time_center_pulse1)):
    if time_center_pulse1[q] >= window[p][2] and time_center_pulse1[q] <=
window[p][3] and dx_center_pulse1[q] > window[p][4]:
        time_center_pulse.append(time_center_pulse1[q])
        dx_center_pulse.append(dx_center_pulse1[q])

linex = []
liney = []
int1, slope1 = linearRegression(time_center_pulse, dx_center_pulse)

power_input.append(window[p][5])
velocity.append(abs(slope1))

for i in range(len(time_center_pulse)):
    value = int1 + slope1 * time_center_pulse[i]
    if value <= float(dy[len(dy) - 1]):
        liney.append(value)
        linex.append(time_center_pulse[i])

time_3d = []

```

```

strain_3d = []
x_axis_3d = []
for i in range(len(data)):
    current_time = time[i]
    for z in range(len(data[i])):
        if time[i] >= window[p][0] and time[i] <= window[p][1] and dy[z] >= 0.4:
            time_3d.append(current_time)
            strain_3d.append(data[i][z])
            x_axis_3d.append(dy[z])

fig = matplotlib.pyplot.gcf()
matplotlib.pyplot.scatter(time_3d, x_axis_3d, c=strain_3d)
matplotlib.pyplot.plot(linex, liney, label=str(slope1.__round__(3)), color='black', ls=':')
matplotlib.pyplot.xlabel('time (s)', fontsize=12)
matplotlib.pyplot.ylabel('dy (m)', fontsize=12)
matplotlib.pyplot.colorbar(label='strain (r'$\mu$S)')
fig.set_size_inches(5.5, 4.5)
matplotlib.pyplot.savefig('images\\LUNA\\' + str(abs(slope1.__round__(4))) + '.png')
matplotlib.pyplot.close()

return power_input, velocity

def getRiserT(csv_riserT):
    data = []
    for i in csv_riserT:
        data.append(i)

    time = []
    monitors = []
    dx = []
    counter = 0

```

```

end_temp = []
while counter < 2.3:
    dx.append(counter)
    counter += 0.1
dx.append(2.35-0.062)

for i in range(len(data)):
    store = []
    for j in range(len(data[i])):
        try:
            if j == 0:
                time.append(float(data[i][j])*0.001/10)
            else:
                store.append(float(data[i][j]))
        except:
            pass
    monitors.append(store)
monitors.pop(0)

for i in range(len(monitors)):
    end_index = len(monitors[i]) - 1
    end_temp.append(monitors[i][end_index])

mon_organized = []
complete = False
mon_ind = 0
while complete == False:
    location = []
    for z in range(len(monitors)):
        location.append(monitors[z][mon_ind])

```



```

mon_organized.append(location)
mon_ind += 1
if mon_ind == 23:
    complete = True

return time, mon_organized, end_temp, dx

def getDownT(csv_downT):
    data = []
    for i in csv_downT:
        data.append(i)

    time = []
    monitors = []
    dx = []
    counter = 0
    end_temp = []
    while counter < 2:
        dx.append(counter)
        counter += 0.1

    for i in range(len(data)):
        store = []
        for j in range(len(data[i])):
            try:
                if j == 0:
                    time.append(float(data[i][j]) * 0.001 / 10)
            else:
                store.append(float(data[i][j]))
        except:

```

```

        pass
    monitors.append(store)
monitors.pop(0)

for i in range(len(monitors)):
    end_index = len(monitors[i]) - 1
    end_temp.append(monitors[i][end_index])

return time, monitors, end_temp, dx

def getWallT(csv_wallT):
    data = []
    for i in csv_wallT:
        data.append(i)

    time = []
    monitors = []
    dx = []
    counter = 0
    end_temp = []
    while counter < 2.4:
        dx.append(counter)
        counter += 0.1
    dx.append(2.355)
    dx.append(2.39)

    for i in range(len(data)):
        store = []
        for j in range(len(data[i])):
            try:
                if j == 0:

```

```

        time.append(float(data[i][j])*0.001/10)
    else:
        store.append(float(data[i][j]))
    except:
        pass
    monitors.append(store)
monitors.pop(0)

```

```

for i in range(len(monitors)):
    end_index = len(monitors[i]) - 1
    end_temp.append(monitors[i][end_index])

```

```

mon_organized = []
complete = False
mon_ind = 0
while complete == False:
    location = []
    for z in range(len(monitors)):
        location.append(monitors[z][mon_ind])
    mon_organized.append(location)
    mon_ind += 1
    if mon_ind == 25:
        complete = True

```

```

return time, mon_organized, end_temp, dx

```

```

def velToRe(x_axis, vel, rho, d_h, mu):
    re_num = []
    for q in range(len(x_axis)):
        reyn = vel[q] * rho * d_h / mu
        re_num.append(reyn)

```

```
return x_axis, re_num
```

```
def velToMfr(x_axis, vel, rho, area):
```

```
    re_num = []
```

```
    for q in range(len(x_axis)):
```

```
        mfr = vel[q] * rho * area
```

```
        re_num.append(mfr)
```

```
    return x_axis, re_num
```