Analytical and Numerical Modeling of Solid-Liquid Phase Change Driven by Internal Heat Generation in Cylindrical Coordinates

A Thesis

Presented in Partial Fulfillment of the Requirements for the

Degree of Master of Science

with a

Major in Mechanical Engineering

in the

College of Graduate Studies

University of Idaho

by

Patrick C. Paulus

Major Professor: John Crepeau, Ph.D.

Committee Members: Lyudmyla Barannyk, Ph.D.; Matthew Swenson, Ph.D.

Department Administrator: Gabriel Potirniche, Ph.D.

December 2021

Authorization to Submit Thesis

This thesis of Patrick C. Paulus, submitted for the degree of Master of Science with a Major in Mechanical Engineering and titled "Analytical and Numerical Modeling of Solid-Liquid Phase Change Driven by Internal Heat Generation in Cylindrical Coordinates," has been reviewed in final form. Permission, as indicated by the signatures and dates below is now granted to submit final copies for the College of Graduate Studies for approval.

Advisor:	John Crepeau, Ph.D.	Date
Committee Members:	Lyudmyla Barannyk, Ph.D.	Date
	Matthew Swenson, Ph.D.	Date
Department Chair:	Gabriel Potirniche, Ph.D.	Date

Abstract

The study of liquid-solid phase change has value in a range of applications, including in nuclear power. Nuclear fuel rods are subject to internal heat generation that, during extreme conditions, can result in the fuel becoming partially molten. Understanding of the melting process is critical to the safe design and operation of nuclear power plants. However, analytical work in this area is still limited.

Two scenarios of liquid-solid phase change driven by internal heat generation are presented for a cylindrical domain: a case with Constant Surface Temperature (CST), and a case with Constant Surface Heat Flux (CSHF). We conducted an analysis of both scenarios in the form of the Stefan problem, a free boundary problem where the position of the interface between liquid and solid phases can change in time. By assuming constant thermal properties, pure conduction, and a sharp interface, we were able to use the superposition principle to derive closed-form, first-order ordinary differential equations with infinite series describing interface motion for one-dimensional, isothermal phase change.

We compared our analytical models against numerical solutions generated through the commercial software Ansys Fluent. This software uses the enthalpy method, which allows the formation of a mushy zone, to solve for temperature, enthalpy, and liquid fraction in the problem domain. Using this model, we were able to check our solutions for mathematical soundness and evaluate the implications of assuming a sharp liquid-solid interface. We performed comparisons between the analytical and numerical models during both melting and solidification scenarios for several values of Stefan number for the CST case and several values of heat flux for the CSHF case.

The CST case saw strong agreement in interface position in time for the slower phase change cases. However, during the melting scenarios for these slower speeds, we saw a divergence in temperature profiles characterized by a nonphysical overheating phenomenon at smaller time steps due to the formation of a mushy zone in the numerical model. This issue lessened with higher interface speeds and did not present itself during the solidification cases. Agreement in the CSHF cases was weaker than for the CST cases, with disagreement being most significant when the material was mostly solid. This disagreement was attributed largely to inaccuracy of the theoretical model. At faster phase change speeds, we saw more mushy zone development in the melting cases, with the solidification cases once again showing no mushy zone effects.

Acknowledgments

I owe many thanks to my advisor, Dr. John Crepeau, for his constant patience, support, and understanding; to Nick Swan and Daicheng Fu with SEL for providing me with not only the financial stability to complete this degree but also for challenging me to improve myself as a future engineer; and to Dr. Lyudmyla Barannyk, Dr. Alexey Sahknov, and Sidney Williams for all of their tireless efforts in phase change research.

Dedication

It takes a village.

Thank you to my family, both born and found.

To my friends, here and there.

To Jess, for advice and accountability.

And to Amanda, for helping me remember what it is to be passionate.

This one's for you.

Table of Contents

List of Figures

List of Acronyms

AWA Area Weighted Average

CFD Computational Fluid Dynamics

CSHF Constant Surface Heat Flux

CST Constant Surface Temperature

FDM Finite Difference Method

FEA Finite Element Analysis

FVM Finite Volume Method

GUI Graphical User Interface

ODE Ordinary Differential Equation

Chapter 1: Introduction

Solid-liquid phase change is a phenomenon common in many engineering processes, including welding, brazing, and soldering; metal casting and additive manufacturing; geological and ice shelf modeling; energy storage; and thermal safety design. Several forms of internal heat generation exist in real-world applications as well, including joule heating in electronics, body heat generation in metabolism, and heating via chemical and nuclear reactions. Knowledge of both phenomena and their interactions are critical in manufacturing and safety design.

Unfortunately, the marriage of phase change and internal heat generation poses significant complications. As materials transition from liquid to solid or vice-versa, they create a front where the material changes phase. This front is influenced by heat transfer, and the inclusion of internal heat generation affects it in unpredictable ways, including increasing the size of the front. Mathematically, internal heat generation adds nonhomogeneous terms to governing equations and adds complexity to the already arduous task of modeling the free boundary that is a phase change front. The result is systems that are often difficult if not impossible to generate closed-form solutions for without fairly restrictive assumptions and idealization. Experimental validation is challenged by the need to measure non-uniform temperature distributions in a partially molten medium.

The development of solutions to problems involving solid-liquid phase change and internal heat generation is paramount in safe nuclear power design. Nuclear fuel rods are subject to internal heat generation from irradiation. In crisis conditions, this heat generation can outpace cooling efforts, leading to partial or total melting of the fuel rod. Being able to predict the response of the fuel rod during such conditions is valuable to mitigating risk.

This thesis explores two cases of heat transfer in the realm of solid-liquid phase change, with both cases considering an infinitely long cylinder with a constant radius. In the first of the cases, the wall of the cylinder is held at a constant temperature, and an internal heat generation source is applied. As heat increases, the interior of the cylinder becomes molten, and an interface between regions of the material that are liquid and solid forms. The position of this interface exists as a function of time and tends towards some steady-state position.

The second case explores the same geometry, but instead of a constant temperature at the wall, a constant amount of heat leaves. If the total heat flux out is less than the total internal heat generated, the cylinder tends towards a fully liquid phase, but as the heat flux increases, the cylinder begins to lose temperature and becomes solid. In both cases, the heat transfer is modeled by conduction, a realm of heat transfer rich in analytical modeling.

The thesis is organized as follows: Chapter 2 analyzes historical and current literature related to liquid-solid phase change and Stefan problems. Gaps in knowledge are identified to solidify the place of our work in the overall framework of understanding of phase change phenomena. Chapter 3 derives the governing equations for our two cases, discusses the solution methodology to be used, and implements the methodology. Chapter 4 proposes several numerical models to verify the analytical solutions generated in Chapter 3, identifies one to be implemented, and discusses the case setups. Model results for both the analytical and numerical models are presented in Chapter 5, with causes of disagreement and key system characteristics discussed. Finally, Chapter 6 summarizes the work and proposes areas of future development and improvement.

Chapter 2: Literature Review

Study of the solid-liquid phase change phenomena was first documented in 1831 by Lamé and Clapeyron's research into the formation of the Earth's crust [1]. Their work studied the solidification of a homogeneous fluid subject to constant temperature along its boundary and found the solid phase grew in proportion to the square root of its time scale. Research in the field lay relatively stagnant until 1889, when the phase change phenomena was expanded into the general class of free boundary problems in four seminal papers published by the now eponymous Stefan [2], whose work focused on melting and expansion of the polar ice cap. Stefan's work introduced the concept of a time-dependent, or moving, boundary problem, where the position of said boundary is a function of time and space. This boundary was expressed through a conservation of energy accounting for latent heat storage.

Danilyuk [3] notes that research into phase change and the Stefan problem saw another decades-long lull in activity until the early 1930's, when interest into practical applications of phase change gained traction. This period saw a rise in similarity, or Neumann, solutions to one-dimensional problems, wherein the solution is self-similar in form to the initial governing equations, absent a characteristic time scale. This time period also saw an increase in focus on problems in cylindrical and spherical coordinate systems. Rubinstein [4] compiled many of the solutions from this time period into the late 1950's. While some general characteristics of free boundary problems had become understood, few exact solutions had been found, and most problems were tackled by numerical methods, greatly limited by the computing power of the time. The 1960's through mid-1980's saw significant expansion into numerical methods, including implementations based on an enthalpy function. These developments, as well as advancements in analytical methods of the time period, are recorded comprehensively by Crank [5]. Continuing into the end of the 20th century, a review of methods extending from this timeframe was conducted by Hu and Argyropoulos [6].

Research into the Stefan problem, free boundary problems, and phase change has a broad range of applications, both in and out of engineering. Within the realm of earth sciences, many of the mechanisms involved in the early formation of the Earth's core are captured by melting and solidification phenomena. In their work, Ulvrova et. al [7] numerically modeled melting and solidification in basal magma and its interactions with convection. Looking to the mechanics of ice formation, Dragomirescu et. al [8] derived semi-analytical solutions for a radially symmetric freezing process through perturbation techniques applied followed by an iterative process. Their work focused on ice formation from supercooled liquids, where small disturbances lead to rapid and unstable solidification. In the world of biology, understanding of thermal responses of tissue undergoing freezing is critical to successful cryosurgery. Singh and Kumar

[9] applied finite difference solidification models to study biological tissue freezing including various forms of bio-heating. Their model employed an enthalpy model and a modification of the standard heat equation to include phase lag. The Stefan problem can be applied outside of phase change as well; in chemistry, it can model growing reaction boundaries. Fedorov and Shul'gin [10] used such a model to describe ignition and combustion of magnesium. Stefan boundary conditions, extended into two dimensions, can also be used to model the formation of sedimentary ocean deltas, as demonstrated by Lorenzo-Trueba and Voller [11].

Due to the complex nature of phase change, even in the modern day, most solutions are limited to approximations in one dimension, typically with infinite or semi-infinite bounds. One such example is the solution generated by Regis et. al [12]. Their solution used a two-point Hermite approximation to obtain heat transfer coefficients within a transient lumped parameter model under the assumption of circumferential symmetry. To accommodate multiple dimensions and finite geometry, numerical methods are regularly employed. One of the most commonly used is the enthalpy method, which enjoyed broad development and implementation by Voller [13]. This method utilizes fixed domains and a natural enforcement of heat fluxes and is a common framework for many numerical schemes. Voller and Swaminathan [14] developed a source-based formulation of the enthalpy method. Its robustness led to widespread adoption by commercial Computational Fluid Dynamics (CFD) packages. A variation of the method was developed by Caldwell and Chan [15] for spherical systems.

While phase change alone is a difficult phenomenon to model, the subset of phase change with internal heat generation is even more complex, and is less well-studied. Understanding of this specific subset of problems is nonetheless valuable in several applications. Furlong and Chapman [16] applied phase change driven by internal generation to geological processes to explain structural characteristics of the lithosphere through heat generation and melting zones. Weinstein [17] theorized catastrophic episodes of rapid thermal convection in the Earth's mantle due to effects of phase change and strong radiogenic heat production. In modern industrial processes, Jiang et. al [18] applied phase change models to study solidification of immiscible alloys subject to electric current pulses. They characterized the nucleation, microstructural development, and electrical conductivity of precipitated phase droplets, increasing understanding of phase deposition processes.

The primary application of concern to us is in modeling of nuclear fuel rods. Nuclear fuel rods are subject to internal, radiogenic heat generation. Under crisis conditions such as over-reactivity or loss of coolant, this heat generation can lead to fuel rod meltdown, a major safety concern. Understanding of the meltdown process is thus critical to the safe design and application of nuclear power plants. Kumar et. al [19] studied the interaction of coolant and molten fuel droplets in fast reactors. They used commercial

computational fluid dynamics software to model the effect of decay heat on the solidification process. Tang et. al [20] studied fuel rod meltdown specific to light water reactors with three-layer fuel rods. Using an enthalpy-based method on a half boundary, they simulated meltdown processes for both excess reactivity and loss of coolant. The model incorporated temperature-dependent thermophysical properties as well as convection and radiation heat transfer in one dimension. Chen et. al [21] developed explicit finite difference models using an enthalpy formulation to model the melting of nuclear fuel rods under accident conditions. This model was used to address shortcomings in the lumped capacitance model the group developed in [22].

Experimental work in this field is very limited. One of the few noted works with experimental data for internal heat generation-driven phase change was conducted by Shrivastava et. al [23]. This experiment used Joule heating to represent the heating mechanism. A CFD numerical investigation, similar to that used in this thesis, showed good agreement within the bounds of measurement uncertainty.

Several approximate techniques have been applied to the subset of phase change problems with heat generation, often with similar limitations to solutions to the classical Stefan problem — geometries are typically restricted to infinite or semi-infinite planes. Yu et. al [24] applied perturbation techniques to a variation of the Stefan problem in one dimension involving a source term. They were able to validate a closed-form solution for the case of a semi-infinite plane. Lumped parameter models were used by An and Su [22] to approximate one-dimensional melting with volumetric heat generation under several impositions of boundary condition. Their model used two-point Hermite approximations for integrals, and was tested across a range of Stefan and Biot numbers. Their work was a direct improvement on that done by Regis et. al. Slota [25] used a variational iteration method to solve direct and inverse one-phase Stefan problems. This method created continuous approximation functions within the solution region, as opposed to the discrete solutions of finite difference and finite element methods.

A common simplification to yield approximate results is a quasi-static assumption, wherein the motion of the temperature profiles is sufficiently slow relative to the motion of the interface such that the transient component of the profile can be neglected. Solutions under such a quasi-steady approximation were derived by Jiji and Gaye [26]. In their model, phase change with uniform heat generation was driven by a sudden surface temperature change. They noted a difference between melting and solidification mechanics in that melting was characterized by a mixture of solid and liquid material at the fusion temperature. Similarly, Crepeau and Siahpush [27] employed a quasi-static assumption to approximate melting and solidification of a material subject to heat generation in Cartesian, cylindrical, and spherical coordinates. They found that interface motion is dependent on the first, square, and cube of time respective to each coordinate system.

Fully analytical solutions are more limited. Bechiri [28] investigated melting and solidification influenced by internal heat generation in cylindrical coordinates, subject to convective heat transfer on the surface. They derived analytical solutions to their problem through separation of variables and an exponential integral function. McCord et. al [29] expanded on the quasi-steady work of Crepeau and Siahpush, deriving analytical solutions to the Stefan problem with internal heat generation in Cartesian coordinates for a constant surface temperature. Their solution adopted a first-order, ordinary differential equation modeling the interface motion through methods of superposition and separation of variables. Similar solutions, but in cylindrical coordinates, were developed by Barannyk et. al [30]. A solution for the constant surface heat flux analog was presented by Barannyk et. al [31]; these last two papers are the basis for the analytical solutions presented in this thesis.

Chapter 3: The Stefan Problem

The following chapter adapts the work done in both [30] and [31], expanding the steps taken and formulating a general strategy for this form of systems of partial differential equations. We look first at formulating the mathematical problem to be solved, then solve first for the Constant Surface Temperature (CST) case then for the Constant Surface Heat Flux (CSHF) case. Finally, a brief discussion of the solution implementation is given.

3.1 Problem Formulation

The situation that we are considering is the process of liquid-solid phase change in cylindrical coordinates during an internal heat generation event, formulated as a Stefan problem. This is similar to an electric wire receiving ohmic heat generation during a high power event or a control rod being heated by irradiation during nuclear power plant operation. A benefit of the Stefan problem formulation is that it uses the heat conduction equation, which has been widely studied and has well-known analytical solutions for many cases.

We begin with an infinitely long cylinder, visually depicted in Figure 3.1, with radius r_0 and some initial temperature profile. An internal heat generation source, $\dot{q}$, is applied to the whole of the cylinder interior, resulting in the temperature of the interior of the cylinder to rise above the melting temperature, T_m, and become molten. The wall is held at either a prescribed temperature, T_0, for the CST case, or a prescribed heat flux, q_0'', for the CSHF case, which allows a portion of the outer region of the cylinder to remain solid. An interface forms at s that divides the cylinder into a liquid region and a solid region.

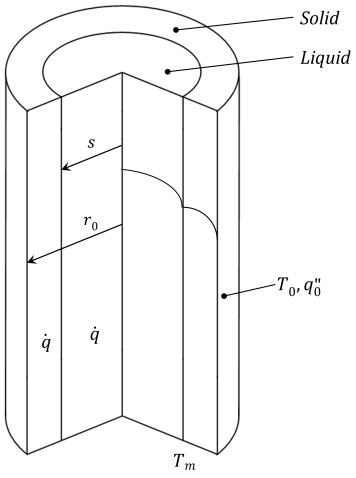

Figure 3.1: Visual depiction of the problem domain, with key parameters labeled.

A number of simplifying assumptions are required to generate a system of equations with a closed-form analytical solution. For our given problem, the following assumptions and their consequences are presented here:

1. All thermophysical properties of the domain material are constant with respect to temperature and phase. This allows for all differential terms to be with respect to temperature, T, only, simplifying the overall system. The thermophysical properties of real materials are not truly constant but rather vary over large temperature ranges; thus, this assumption requires a relatively small temperature range.

2. The internal heat generation is constant and uniform within the problem domain. Similar to the previous assumption, this simplifies differential terms to include only temperature. This restricts the solution to simplified scenarios; however, radiation, ohmic, and chemical heat generation phenomena can all be well-estimated with this assumption.

3. The domain material is a pure material and changes phase from solid to liquid and vice-versa at a singular temperature, T_m. Alloys and materials with a different melting and solidification temperature exhibit complex and pseudo-random phenomena, such as dendritic growth, that cannot be well-described analytically. This assumption avoids such complexity. However, it restricts the domain material to either pure materials or eutectic alloys without complex or geometric phase phenomena.

4. There is no velocity field within the liquid region. This effectively neglects heat transfer by convection, meaning that we reduce the heat transfer to purely conduction. Natural convection is driven by large temperature variations and subsequent changes in density; our previous stipulation of small temperature changes and constant thermophysical properties is sufficient to neglect this form of heat transfer. Furthermore, convective cells require time to form; a further assumption of a relatively short time period allows us to neglect velocity from a temporal standpoint.

5. Temperature variation is independent of axial and circumferential position. This allows for a spatially one-dimensional domain, which, while requiring idealized circumstances, is substantially simpler to solve.

6. Phase change occurs along a sharp interface, such that the space between liquid and solid phases is infinitesimal and there is no mushy zone. In reality, energy is stored in a finite region of material during phase change. The consequence of this assumption is that at early stages of melting and solidification, energy distribution within and nearby the phase change front will be inaccurate.

7. The time scale of front motion is sufficiently small relative to the time scale of temperature change. This assumption is necessary to allow us to separate the governing equation into transient and steady-state components while satisfying the free boundary.

Implementing these assumptions yields our governing equation, the heat conduction equation with internal heat generation for cylindrical coordinates [32]:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right)+\frac{\dot{q}}{k}=\frac{1}{\alpha}\frac{\partial T}{\partial t} \tag{3.1}$$

This equation holds for the domain $T = T(r,t),\ 0 \leq s \leq r_0,\ 0 \leq t < \infty$. Note that the governing equation functionally exists twice: once for the liquid phase, and once for the solid phase. To track the interface, we conduct an energy balance consisting of energy entering from the liquid region, energy transferred to the solid region, and energy stored during phase change. This energy balance, described mathematically, is [33]:

$$k_{liq}\left.\frac{\partial T_{liq}(r,t)}{\partial r}\right|_{r=s(t)}+\rho_{sol}\Delta h_f\frac{ds(t)}{dt}=k_{sol}\left.\frac{\partial T_{sol}(r,t)}{\partial r}\right|_{r=s(t)} \tag{3.2}$$

We now have our full governing system of equations, but to obtain the full strong form, we must include boundary and initial conditions. As our governing system consists of two second-order, time-dependent partial differential equations and one Ordinary Differential Equation (ODE) in time, we will have a total of four boundary conditions and three initial conditions.

With the system initially in solid phase, melting occurs at the center and is driven outward by the internal heat generation. Thus, the liquid region consists of material from the centerline of our cylinder out to the interface location. At the centerline, we stipulate that our solution must be bounded, continuous, and symmetric, leading to the prescription of a zero gradient boundary condition. At the interface location, the material must be at melting temperature. Additionally, we prescribe an arbitrary initial temperature profile, $\phi(r)$, as a function of position only. These two boundary conditions and one initial condition defining the liquid region are mathematically described by:

$$\left.\frac{\partial T_{liq}(r,t)}{\partial r}\right|_{r=0}=0 \tag{3.3}$$

$$T_{liq}(s(t),t)=T_m \tag{3.4}$$

$$T_{liq}(r,0)=\phi_{liq}(r) \tag{3.5}$$

The solid region consists of the remaining material from the interface location to the wall of the

cylinder. Similar to the liquid region, the material must be at melting temperature at the interface location, and an arbitrary initial temperature profile is prescribed.

The boundary condition at the wall is contingent upon the analysis conducted. For the case of constant surface temperature, a prescribed wall temperature is used, while for constant surface heat flux, a prescribed outward heat flux must be used. The CST case yields the following conditions for the solid region:

$$T_{sol}(s(t),t) = T_m \tag{3.6}$$

$$T_{sol}(r_0,t) = T_0 \tag{3.7}$$

$$T_{sol}(r,0) = \phi_{sol}(r) \tag{3.8}$$

And for the CSHF case:

$$T_{sol}(s(t),t) = T_m \tag{3.9}$$

$$-k_{sol}\frac{\partial T}{\partial r}\bigg|_{r=r_0} = q_0'' \tag{3.10}$$

$$T_{sol}(r,0) = \phi_{sol}(r) \tag{3.11}$$

Finally, we prescribe an initial interface location, $s(t=0) = s_0$. Combining this last condition with the governing system of equations and other boundary and initial conditions, we have the strong form of the Stefan problem with internal heat generation in cylindrical coordinates. For the CST case, the strong form is:

$$\begin{aligned}
&\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T_{liq}}{\partial r}\right) + \frac{\dot{q}}{k_{liq}} = \frac{1}{\alpha_{liq}}\frac{\partial T_{liq}}{\partial t}, && \text{for } 0 \le r \le s(t);\\
&\frac{\partial T_{liq}(r,t)}{\partial r}\bigg|_{r=0} = 0,\ T_{liq}(s(t),t) = T_m,\ T_{liq}(r,0) = \phi_{liq}(r);\\
&\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T_{sol}}{\partial r}\right) + \frac{\dot{q}}{k_{sol}} = \frac{1}{\alpha_{sol}}\frac{\partial T_{sol}}{\partial t}, && \text{for } s(t) \le r \le r_0;\\
&T_{sol}(s(t),t) = T_m,\ T_{sol}(r_0,t) = T_0,\ T_{sol}(r,0) = \phi_{sol}(r);\\
&k_{liq}\left.\frac{\partial T_{liq}(r,t)}{\partial r}\right|_{r=s(t)} + \rho_{sol}\Delta h_f\frac{ds(t)}{dt} = k_{sol}\left.\frac{\partial T_{sol}(r,t)}{\partial r}\right|_{r=s(t)}, && \text{for } 0 \le t < \infty;\\
&s(t=0) = s_0
\end{aligned} \tag{3.12}$$

And for the CSHF case:

$$
\begin{aligned}
&\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T_{liq}}{\partial r}\right)+\frac{\dot{q}}{k_{liq}}=\frac{1}{\alpha_{liq}}\frac{\partial T_{liq}}{\partial t}, && \text{for } 0\le r\le s(t);\\
&\left.\frac{\partial T_{liq}(r,t)}{\partial r}\right|_{r=0}=0,\ T_{liq}(s(t),t)=T_m,\ T_{liq}(r,0)=\phi_{liq}(r);\\
&\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T_{sol}}{\partial r}\right)+\frac{\dot{q}}{k_{sol}}=\frac{1}{\alpha_{sol}}\frac{\partial T_{sol}}{\partial t}, && \text{for } s(t)\le r\le r_0;\\
&T_{sol}(s(t),t)=T_m,\ -k_{sol}\left.\frac{\partial T}{\partial r}\right|_{r=r_0}=q_0'',\ T_{sol}(r,0)=\phi_{sol}(r);\\
&k_{liq}\left.\frac{\partial T_{liq}(r,t)}{\partial r}\right|_{r=s(t)}+\rho_{sol}\Delta h_f\frac{ds(t)}{dt}=k_{sol}\left.\frac{\partial T_{sol}(r,t)}{\partial r}\right|_{r=s(t)}, && \text{for } 0\le t<\infty;\\
&s(t=0)=s_0
\end{aligned}
\tag{3.13}
$$

3.2 Solution: Constant Surface Temperature

This solution is based on the work derived and presented in [30]. The following general procedure is used in developing a closed-form solution. This methodology will be used for both the CST and CSHF cases:

1. Derive the strong form system of equations.
2. Establish the requirements for phase change to occur.
3. Perform dimensional analysis and reduce the system to a nondimensional form.
4. Using the principle of superposition, separate the governing equation of the liquid region into transient and steady-state components.
5. Solve the steady-state components using standard integration techniques.
6. Solve the transient components using separation of variables.
7. Combine the transient and steady-state solution components to obtain overall temperature profiles.
8. Repeat steps 4 through 7 for the solid region.
9. Differentiate the temperature profiles and apply them to the interface equation.
10. Establish suitable initial conditions for physical interface curves.

Having already derived the strong form system of equations for our cases, we move to establish the requirements for phase change. In order for the phenomena to occur in the CST case, the overall heat

generated within the problem domain must be greater than the heat rejected through the boundary. To quantify the level of heat generation required, we look at the steady state case:

$$\begin{aligned}
&\frac{1}{r}\frac{d}{dr}\left(r\frac{dT_{liq,ss}}{dr}\right)+\frac{\dot{q}}{k_{liq}}=0, && \text{for } 0\le r\le s_{ss};\\
&\left.\frac{dT_{liq,ss}(r)}{dr}\right|_{r=0}=0,\ T_{liq,ss}(s_{ss})=T_m,;\\
&\frac{1}{r}\frac{d}{dr}\left(r\frac{dT_{sol,ss}}{dr}\right)+\frac{\dot{q}}{k_{sol}}=0, && \text{for } s_{ss}\le r\le r_0;\\
&T_{sol,ss}(s_{ss})=T_m,\ T_{sol,ss}(r_0)=T_0;\\
&k_{liq}\left.\frac{dT_{liq,ss}(r)}{dr}\right|_{r=s_{ss}}=k_{sol}\left.\frac{dT_{sol,ss}(r)}{dr}\right|_{r=s_{ss}}
\end{aligned} \tag{3.14}$$

Both of the steady-state conduction equations can be solved by integrating twice and applying boundary conditions. For the liquid region, this yields:

$$T_{liq,ss}(r)=\frac{\dot{q}s^2}{4k_{liq}}\left(1-\frac{r^2}{s^2}\right)+T_m \tag{3.15}$$

Evaluating the gradient at the interface yields:

$$\left.\frac{dT_{liq,ss}}{dr}\right|_{r=s_{ss}}=-\frac{\dot{q}s_{ss}}{2k_{liq}} \tag{3.16}$$

Solving for the solid region yields:

$$T_{sol,ss}(r)=\frac{\dot{q}r^2}{4k_{sol}}-\frac{\left(s^2\ln r_0-r_0^2\ln s+\left(s^2-r_0^2\right)\ln r\right)\dot{q}+4k_{sol}\left(T_m\ln r_0-T_0\ln s+(T_m-T_0)\ln r\right)}{4k_{sol}\left(\ln s-\ln r_0\right)} \tag{3.17}$$

With gradient evaluated at the interface:

$$\left.\frac{dT_{sol,ss}}{dr}\right|_{r=s_{ss}}=-\frac{\dot{q}s_{ss}}{2k_{sol}}+\frac{\left(s_{ss}^2-r_0^2\right)\dot{q}+4k_{sol}\left(T_m-T_0\right)}{4k_{sol}s_{ss}\left(\ln s_{ss}-\ln r_0\right)} \tag{3.18}$$

We can now substitute the gradients into our steady-state interface equation. Acknowledging $k_{liq}=k_{sol}$, simplifying, and solving for $\frac{s_{ss}}{r_0}$, we get:

$$\frac{s_{ss}}{r_0}=\sqrt{1-\frac{4k\left(T_m-T_0\right)}{r_0^2\dot{q}}},\ 0<\frac{s_{ss}}{r_0}<1 \tag{3.19}$$

We expect our steady state position to be within the problem domain and of a non-complex value, which requires the value inside the square root to be positive We can thus demonstrate for the required

heat generation for melting is:

$$\dot{q} > \frac{4k\left(T_m - T_0\right)}{r_0^2} \tag{3.20}$$

We now want to use dimensional analysis to substitute our current parameters with nondimensional counterparts. The benefits of this nondimensionalization are twofold: our solution is easily generalized to any material or domain that fits the previously stated assumptions of our problem, and we obtain variable groupings that can provide valuable insight as to the general behavior of the system.

To start, we create dimensionless parameters for position and time:

$$\eta = \frac{r}{r_0} \tag{3.21}$$

$$\zeta = \frac{s}{r_0} \tag{3.22}$$

$$\tau = Fo = \frac{\alpha t}{r_0^2} \tag{3.23}$$

These parameters map position onto a range of $[0, 1]$, where the cylinder center is 0 and the cylinder wall is 1. Our dimensionless time, which is equal to the Fourier number Fo, is effectively a ratio of the rate of heat conduction to the overall heat storage. Next, we introduce a dimensionless temperature:

$$\theta = \frac{T - T_0}{T_m - T_0} \tag{3.24}$$

Our dimensionless temperature remaps wall temperature to 0 and melting temperature to 1. Finally, we have two dimensionless heat transfer parameters:

$$\dot{Q} = \frac{\dot{q} r_0^2}{k(T_m - T_0)} \tag{3.25}$$

$$St = \frac{c_P(T_m - T_0)}{\Delta h_f} \tag{3.26}$$

These two parameters consist of the domain inputs and thermophysical properties that dictate the general behavior of the system. The dimensionless heat generation term, $\dot{Q}$, is a ratio of the internal heat generation in the domain material to that material's ability to conduct heat. Higher values of $\dot{Q}$ correspond to higher levels of heat source relative to sink. The Stefan number St is a ratio of the sensible heat storage within a system to its latent heat storage. For Stefan problems, it correlates with the relative

speed a phase change front can move.

As previously discussed, a critical level of internal heat generation is required for melting phenomena to occur. Substituting our nondimensional parameters into Eq. (3.19) yields:

$$\zeta = \sqrt{1 - \frac{4}{\dot{Q}}} \tag{3.27}$$

Phase change will only occur if Equation (3.27) yields real values, resulting in the following requirement:

$$\dot{Q} > 4 \tag{3.28}$$

With the dimensionless groupings we have created, we can now redefine our governing system in a nondimensional form. Substituting values and simplifying yields the following system:

$$\begin{aligned}
&\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{liq}(\eta,\tau)}{\partial\eta}\right) + \dot{Q} = \frac{\partial\theta_{liq}(\eta,\tau)}{\partial\tau} && \text{for } 0 \le \eta \le \zeta(\tau);\\
&\left.\frac{\theta_{liq}(\eta,\tau)}{\partial\eta}\right|_{\eta=0} = 0,\ \theta_{liq}(\zeta(\tau),\tau) = 1,\ \theta_{liq}(\eta,0) = \Phi_{liq}(\eta)\\
&\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{sol}(\eta,\tau)}{\partial\eta}\right) + \dot{Q} = \frac{\partial\theta_{sol}(\eta,\tau)}{\partial\tau} && \text{for } \zeta(\tau) \le \eta \le 1;\\
&\theta_{sol}(\zeta(\tau),\tau) = 1,\ \theta(1,\tau) = 0,\ \theta_{sol}(\eta,0) = \Phi_{sol}(\eta)\\
&\left.\frac{\partial\theta_{liq}(\eta,\tau)}{\partial\eta}\right|_{\eta=\zeta(\tau)} + \frac{1}{St}\frac{d\zeta(\tau)}{d\tau} = \left.\frac{\partial\theta_{sol}(\eta,\tau)}{\partial\eta}\right|_{\eta=\zeta(\tau)} && \text{for } 0 \le \tau < \infty;\\
&\zeta(\tau=0) = \zeta_0
\end{aligned} \tag{3.29}$$

Liquid Region Solution

The principle of superposition allows the components of a linear system to be separated into individual terms within a sum. In our system, it allows us to solve the conduction equations as the sums of their transient and steady components. For the liquid region, this yields:

$$\theta_{liq}(\eta,\tau) = \theta_{liq,tr}(\eta,\tau) + \theta_{liq,ss}(\eta) \quad \text{for } 0 \le \eta \le \zeta(\tau) \tag{3.30}$$

Substituting the separated temperature profiles into the liquid conduction equation yields:

$$\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{liq,tr}(\eta,\tau)}{\partial\eta}\right) + \frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{liq,ss}(\eta)}{\partial\eta}\right) + \dot{Q} = \frac{\partial\theta_{liq,tr}(\eta,\tau)}{\partial\tau} \tag{3.31}$$

With associated boundary and initial conditions:

$$\begin{aligned} &\frac{\partial \theta_{liq,tr}(0,\tau)}{\partial \eta} + \frac{\partial \theta_{liq,ss}(0)}{\partial \eta} = 0, \\ &\theta_{liq,tr}(\zeta(\tau),\tau) + \theta_{liq,ss}(\zeta(\tau)) = 1, \\ &\theta_{liq,tr}(\eta,0) + \theta_{liq,ss}(\eta) = \Phi_{liq}(\eta) \end{aligned} \tag{3.32}$$

The modified liquid system of equations can be further separated into distinct systems for the transient and steady-state components. Starting with the steady-state system, we have:

$$\begin{aligned} &\frac{1}{\eta}\frac{d}{d\eta}\left(\eta \frac{d\theta_{liq,ss}(\eta)}{d\eta}\right) = -\dot{Q}, \\ &\frac{d\theta_{liq,ss}(0)}{d\eta} = 0, \\ &\theta_{liq,ss}(\zeta(\tau)) = 1 \end{aligned} \tag{3.33}$$

This system can be easily solved by integrating twice and applying boundary conditions, yielding the resultant profile:

$$\theta_{liq,ss}(\eta) = \frac{1}{4}\dot{Q}\left(\zeta^2(\tau) - \eta^2\right) + 1 \tag{3.34}$$

Turning our attention to the transient system, we have:

$$\begin{aligned} &\frac{1}{\eta}\frac{\partial}{\partial \eta}\left(\eta \frac{\partial \theta_{liq,tr}(\eta,\tau)}{\partial \eta}\right) = \frac{\partial \theta_{liq,tr}(\eta,\tau)}{\partial \tau}, \\ &\frac{\partial \theta_{liq,tr}(0,\tau)}{\partial \eta} = 0, \\ &\theta_{liq,tr}(\zeta(\tau),\tau) = 0, \\ &\theta_{liq,tr}(\eta,0) = \Phi_{liq}(\eta) - \theta_{liq,ss}(\eta) \end{aligned} \tag{3.35}$$

This system is now homogeneous and can be solved using the method of separation of variables. We assume a solution exists that is the product of two arbitrary functions which are dependent only on η and τ, respectively:

$$\theta_{liq,tr}(\eta,\tau) = f(\eta)\cdot g(\tau) \tag{3.36}$$

This separation is valid so long as the interface moves sufficiently slowly in time relative to the change in temperature profiles in time. Dropping subscripts and functional inclusions temporarily, it can be readily seen that $\theta' = f' \cdot g$ and $\dot{\theta} = f \cdot \dot{g}$. Substituting these total derivatives and rearranging Eq. (3.35),

we are able to completely separate our governing equation into two equal univariate functions, which must be equal to some constant value:

$$\frac{1}{\eta}\frac{f'}{f} + \frac{f''}{f} = \frac{\dot{g}}{g} = const. \tag{3.37}$$

Both of these now ODEs have common general solutions associated with them:

$$\begin{aligned} f(\eta) &= c_1 J_0(\lambda\eta) + c_2 Y_0(\lambda\eta), \\ g(\tau) &= e^{-\lambda^2\tau} \end{aligned} \tag{3.38}$$

Here, J_0 and Y_0 denote Bessel functions of zero order of the first and second kind, respectively, and λ is an eigenvalue that will allow for a nontrivial solution. To define our integration constants, we turn to boundary conditions. We know that $f(\eta)$ must have a bounded, finite value at $\eta = 0$. As $\lim_{\eta\to 0^+}(Y_0(\lambda\eta)) = \infty$, this necessitates that $c_2 = 0$. For our second boundary condition,

$$f(\eta = \zeta(\tau)) = c_1 J_0(\lambda\zeta(\tau)) = 0 \tag{3.39}$$

The condition $c_1 = 0$ would result in a trivial solution. To prevent this, we stipulate that

$$J_0\left(\lambda_n\zeta(\tau)\right) = 0 \tag{3.40}$$

This transcendental equation defines our eigenvalues as

$$\lambda_n = \frac{z_{0,n}}{\zeta(\tau)} \tag{3.41}$$

where $z_{0,n}$ denotes zeroes of the Bessel function $J_0(\lambda_n\zeta(\tau))$, the eigenfunction for this system. Combining the results of Eq. (3.36) through Eq. (3.41) and summing for all values of n yields the transient temperature for the liquid region as an infinite series:

$$\theta_{liq,tr}(\eta,\tau) = \sum_{n=1}^{\infty} A_n e^{-\lambda_n^2\tau} J_0(\lambda_n\eta) \tag{3.42}$$

Solving for the coefficients A_n requires applying the initial condition in Eq. (3.35):

$$\theta_{liq,tr}(\eta,0) = \Phi_{liq}(\eta) - \theta_{liq,ss}(\eta) = \sum_{n=1}^{\infty} A_n J_0(\lambda_n\eta) \tag{3.43}$$

To isolate A_n, we take advantage of the orthogonality of the eigenfunctions with respect to their

weight function η. By multiplying both sides by $J_0(\lambda_m\eta)\eta$, integrating across the liquid domain, applying orthogonality principles, and rearranging we get:

$$A_n = \frac{\int_0^{\zeta(\tau)} \left(\Phi_{liq}(\eta) - \theta_{liq,ss}(\eta)\right) J_0(\lambda_n\eta)\eta d\eta}{\int_0^{\zeta(\tau)} J_0^2(\lambda_n\eta)\eta d\eta} \tag{3.44}$$

Substituting Eq. (3.34) and Eq. (3.42) into Eq. (3.30) and referencing Eq. (3.41) and Eq. (3.44) for the associated eigenvalues and Fourier coefficients yields the complete liquid temperature profile:

$$\begin{aligned}
\theta_{liq}(\eta,\tau) &= \sum_{n=1}^{\infty}\left[A_n e^{-\lambda_n^2\tau} J_0(\lambda_n\eta)\right] + \frac{\dot{Q}}{4}\left(\zeta^2(\tau) - \eta^2\right) + 1, \\
\lambda_n &= \frac{z_{0,n}}{\zeta(\tau)}, \\
A_n &= \frac{\int_0^{\zeta(\tau)} \left(\Phi_{liq}(\eta) - \theta_{liq,ss}(\eta)\right) J_0(\lambda_n\eta)\eta d\eta}{\int_0^{\zeta(\tau)} J_0^2(\lambda_n\eta)\eta d\eta}
\end{aligned} \tag{3.45}$$

Solid Region Solution

We can now repeat our solution process for the solid region. We start with superposition of the transient and steady-state profiles:

$$\theta_{sol}(\eta,\tau) = \theta_{sol,tr}(\eta,\tau) + \theta_{sol,ss}(\eta) \quad \text{for } \zeta(\tau) \le \eta \le 1 \tag{3.46}$$

Substituting into the solid conduction equation gives:

$$\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\eta}\right) + \frac{1}{\eta}\frac{d}{d\eta}\left(\eta\frac{d\theta_{sol,ss}(\eta)}{d\eta}\right) + \dot{Q} = \frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\tau} \tag{3.47}$$

With associated separated boundary and initial conditions:

$$\begin{aligned}
&\theta_{sol,tr}(\zeta(\tau),\tau) + \theta_{sol,ss}(\zeta(\tau)) = 1, \\
&\theta_{sol,tr}(1,\tau) + \theta_{sol,ss}(1) = 0, \\
&\theta_{sol,tr}(\eta,0) + \theta_{sol,ss}(\eta) = \Phi_{liq}(\eta)
\end{aligned} \tag{3.48}$$

We now gather the steady-state and transient components into their own systems, starting with the steady-state:

$$\begin{aligned}
&\frac{1}{\eta}\frac{d}{d\eta}\left(\eta\frac{d\theta_{sol,ss}(\eta)}{d\eta}\right) = -\dot{Q},\\
&\theta_{sol,ss}(\zeta(\tau)) = 1,\\
&\theta_{sol,ss}(1) = 0
\end{aligned} \tag{3.49}$$

Integrating twice and applying boundary conditions yields:

$$\theta_{sol,ss}(\eta) = \frac{\dot{Q}}{4}(1-\eta^2) + \left(\frac{\dot{Q}}{4}\left(\zeta^2(\tau)-1\right)+1\right)\frac{\ln\eta}{\ln\zeta(\tau)} \tag{3.50}$$

Taking the remaining components we form the transient system:

$$\begin{aligned}
&\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\eta}\right) = \frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\tau},\\
&\theta_{sol,tr}(\zeta(\tau),\tau) = 0,\\
&\theta_{sol,tr}(1,\tau) = 0,\\
&\theta_{sol,tr}(\eta,0) = \Phi_{sol}(\eta) - \theta_{sol,ss}(\eta)
\end{aligned} \tag{3.51}$$

Once more, we use the separation of variables, denoting the univariate functions of the solid region as:

$$\theta_{sol,tr}(\eta,\tau) = \tilde{f}(\eta)\cdot\tilde{g}(\tau) \tag{3.52}$$

These functions have the same general solutions as in the liquid region:

$$\begin{aligned}
&\tilde{f}(\eta) = c_3 J_0(\lambda\zeta(\tau)) + c_4 Y_0(\lambda\zeta(\tau))\\
&\tilde{g}(\tau) = e^{-\lambda^2\tau}
\end{aligned} \tag{3.53}$$

Applying boundary conditions to $\tilde{f}(\eta)$, we have:

$$\begin{aligned}
&\tilde{f}(\zeta(\tau)) = 0 = c_3 J_0(\lambda\zeta(\tau)) + c_4 Y_0(\lambda\zeta(\tau))\\
&\tilde{f}(1) = 0 = c_3 J_0(\lambda) + c_4 Y_0(\lambda)
\end{aligned} \tag{3.54}$$

Considerations to prevent the trivial solution arising from $c_3 = c_4 = 0$ must again be taken. For this case, we enforce the following determinant:

$$\begin{vmatrix} J_0(\lambda\zeta(\tau)) & Y_0(\lambda\zeta(\tau)) \\ J_0(\lambda) & Y_0(\lambda) \end{vmatrix} = 0 \tag{3.55}$$

Rearranged as a transcendental equation, this determinant provides the eigenvalues, $\tilde{\lambda}_n$, for the solid region as roots of the equation:

$$J_0(\tilde{\lambda}_n\zeta(\tau))Y_0(\tilde{\lambda}_n) - Y_0(\tilde{\lambda}_n\zeta(\tau))J_0(\tilde{\lambda}_n) = 0 \tag{3.56}$$

These values can be found using a suitable numerical root finding method. To obtain the eigenfunction for the solid region, we reapply the boundary condition $\tilde{f}(1) = 0$ to determine that:

$$c_3 = -c_4\frac{Y_0(\tilde{\lambda}_n)}{J_0(\tilde{\lambda}_n)} \tag{3.57}$$

Substituting this relation into $\tilde{f}_n$ and acknowledging that c_4 can be wrapped into the solution's Fourier coefficient, we obtain the eigenfunctions:

$$\tilde{f}_n(\eta) = Y_0(\tilde{\lambda}_n\eta) - \frac{Y_0(\tilde{\lambda}_n)}{J_0(\tilde{\lambda}_n)}J_0(\tilde{\lambda}_n\eta) \tag{3.58}$$

Combining the results of Eq. (3.52) through Eq. (3.58) and summing for all n yields the transient solid region temperature profile:

$$\theta_{sol,tr}(\eta,\tau) = \sum_{n=1}^{\infty} B_n e^{-\tilde{\lambda}_n^2\tau}\tilde{f}_n(\eta) \tag{3.59}$$

The Fourier coefficient B_n can be solved for in the same manner as A_n. We first apply the initial condition:

$$\theta_{sol,tr}(\eta,0) = \Phi_{sol}(\eta) - \theta_{sol,ss}(\eta) = \sum_{n=1}^{\infty} B_n\tilde{f}_n(\eta) \tag{3.60}$$

We then multiply both sides by $\tilde{f}_m(\eta)\eta$, integrate over the solid region, apply orthogonality principles, and rearrange to obtain our coefficient:

$$B_n = \frac{\displaystyle\int_{\zeta(\tau)}^{1} \left(\Phi_{sol}(\eta) - \theta_{sol,ss}(\eta)\right)\tilde{f}_n(\eta)\eta d\eta}{\displaystyle\int_{\zeta(\tau)}^{1} \tilde{f}_n^2(\eta)\eta d\eta} \tag{3.61}$$

To obtain the overall solid region temperature profile, we substitute Eq. (3.50) and Eq. (3.59) into Eq. (3.46), referencing Eq. (3.56), Eq. (3.58), and Eq. (3.61) for the associated eigenvalues, eigenfunctions, and Fourier coefficients respectively:

$$
\begin{aligned}
&\theta_{sol}(\eta,\tau) = \left[\sum_{n=1}^{\infty} B_n e^{-\tilde{\lambda}_n^2\tau}\tilde{f}_n(\tilde{\lambda}_n\eta)\right] + \frac{\dot{Q}}{4}\left(1-\eta^2\right) + \left(\frac{\dot{Q}}{4}\left(\zeta^2(\tau)-1\right)+1\right)\frac{\ln\eta}{\ln\zeta(\tau)},\\
&J_0(\tilde{\lambda}_n\zeta(\tau))Y_0(\tilde{\lambda}_n) - Y_0(\tilde{\lambda}_n\zeta(\tau))J_0(\tilde{\lambda}_n) = 0,\\
&\tilde{f}_n(\eta) = Y_0(\tilde{\lambda}_n\eta) - \frac{Y_0(\tilde{\lambda}_n)}{J_0(\tilde{\lambda}_n)}J_0(\tilde{\lambda}_n\eta),\\
&B_n = \frac{\displaystyle\int_{\zeta(\tau)}^{1}\left(\Phi_{sol}(\eta)-\theta_{sol,ss}(\eta)\right)\tilde{f}_n(\eta)\eta d\eta}{\displaystyle\int_{\zeta(\tau)}^{1}\tilde{f}_n^2(\eta)\eta d\eta}
\end{aligned}
\tag{3.62}
$$

The Interface Solution

Eq. (3.45) and Eq. (3.62) describe the overall temperature profile for our system for any given time step τ and associated interface position $\zeta(\tau)$. We can determine this interface position by differentiating both profiles with respect to location:

$$
\begin{aligned}
\frac{\partial\theta_{liq}(\eta,\tau)}{\partial\eta} &= -\sum_{n=1}^{\infty}\left[A_n e^{-\lambda_n^2\tau}\lambda_n J_1(\lambda_n\eta)\right] - \frac{\dot{Q}}{2}\eta\\
\frac{\partial\theta_{sol}(\eta,\tau)}{\partial\eta} &= \sum_{n=1}^{\infty}\left[B_n e^{-\tilde{\lambda}_n^2\tau}\tilde{f}_n'(\tilde{\lambda}_n\eta)\right] - \frac{\dot{Q}}{2}\eta + \left(\frac{\dot{Q}}{4}\left(\zeta^2(\tau)-1\right)+1\right)\frac{1}{\eta\ln\zeta(\tau)}
\end{aligned}
\tag{3.63}
$$

where

$$
\tilde{f}_n' = \frac{d\tilde{f}_n}{d\eta} = -\tilde{\lambda}_n\left[Y_1(\tilde{\lambda}_n\eta) - \frac{Y_0(\tilde{\lambda}_n)}{J_0(\tilde{\lambda}_n)}J_1(\tilde{\lambda}_n\eta)\right] \tag{3.64}
$$

Evaluating Eq. (3.63) at the interface and substituting the resultant relations into the interface equation of our strong form, Eq. (3.12), yields a closed-form ODE for interface location:

$$
\begin{aligned}
\frac{1}{St}\frac{d\zeta(\tau)}{d\tau} &= \sum_{n=1}^{\infty}\left[A_n e^{-\lambda_n^2\tau}\lambda_n J_1(\lambda_n\zeta(\tau))\right]\\
&+ \sum_{n=1}^{\infty}\left[B_n e^{-\tilde{\lambda}_n^2\tau}\tilde{f}_n'(\zeta(\tau))\right] + \left(\frac{\dot{Q}}{4}\left(\zeta^2(\tau)-1\right)+1\right)\frac{1}{\zeta(\tau)\ln\zeta(\tau)}
\end{aligned}
\tag{3.65}
$$

Setting Initial Conditions

The crafted interface solution is valid for arbitrary initial conditions so long as the temperatures in each phase are valid, i.e., $\Phi_{liq} \geq 1$ and $\Phi_{sol} \leq 1$. To ensure physical solutions, we want a smooth initial profile that is at the melting temperature at the interface location. We can use a steady-state conduction

equation to establish this profile:

$$\frac{1}{\eta}\frac{d}{d\eta}\left(\eta\frac{d\Phi}{d\eta}\right)+\dot{Q}=0 \tag{3.66}$$

For the melting scenario, we stipulate the following initial conditions:

$$\begin{aligned}&\dot{Q}=4,\\&\left.\frac{d\Phi}{d\eta}\right|_{\eta=0}=0,\\&\Phi(1)=0\end{aligned} \tag{3.67}$$

Integrating twice, substituting $\dot{Q}$, and applying boundary conditions yields:

$$\Phi(\eta)=1-\eta^2 \tag{3.68}$$

For the melting case, we wish to have an initial interface position of $\zeta_0=0$. As ζ must be nonzero for two phases to exist, we instead prescribe an arbitrarily small initial interface position. Separating Eq. (3.68) for the two phases yields our set of initial conditions for melting:

$$\begin{aligned}&\Phi_{liq}(\eta)=1,\\&\Phi_{sol}(\eta)=1-\eta^2,\\&\zeta_0=\epsilon,\\&\tau=0\end{aligned} \tag{3.69}$$

For the solidification case, we once again use Eq. (3.66), but instead stipulate

$$\begin{aligned}&\left.\frac{d\Phi}{d\eta}\right|_{\eta=0}=0,\\&\Phi(1)=1\end{aligned} \tag{3.70}$$

Integrating Eq. (3.66) twice and applying the boundary conditions of Eq. (3.70) yields:

$$\Phi(\eta)=\frac{\dot{Q}}{4}\left(1-\eta^2\right)+1 \tag{3.71}$$

As before, our desired initial interface position of $\zeta_0=1$ would result in a single phase. We thus subtract an arbitrarily small value ϵ. Separating Eq. (3.71) for the two phases yields the set of initial conditions for solidification:

$$\begin{aligned}
\Phi_{liq}(\eta) &= \frac{\dot{Q}}{4}\left(1-\eta^2\right)+1, \\
\Phi_{sol}(\eta) &= 1, \\
\zeta_0 &= 1-\epsilon, \\
\tau &= 0
\end{aligned} \tag{3.72}$$

3.3 Solution: Constant Surface Heat Flux

We turn our attention now to the case of constant surface heat flux. This work is adapted from [31]. We follow the same solution methodology as before, reiterated here for convenience:

1. Derive the strong form.
2. Establish critical parameters for phase change.
3. Nondimensionalize the system.
4. Use superposition to separate the liquid system into transient and steady-state components.
5. Solve the steady-state system.
6. Solve the transient system.
7. Combine the steady-state and transient components.
8. Repeat steps 4 through 7 for the solid system.
9. Differentiate the solutions and apply to the interface equation.
10. Establish initial conditions.

The strong form has already been defined in Eq. (3.13). For the CSHF case, the critical conditions for phase change can be determined by a simple energy balance. If the overall heat flux out through the walls is less than the overall heat generated internally, the system will tend towards melting, and vice-versa. Mathematically, this is described by the balance of $q_0'' A_s$ and $\dot{q}V$, where A_s and V are the surface area and volume of our problem domain, respectively. We can conclude from the balance:

$$\frac{r_0}{2}\dot{q}\begin{cases} > q_0'' \rightarrow \text{Melting} \\ = q_0'' \rightarrow \text{Equilibrium} \\ < q_0'' \rightarrow \text{Solidification} \end{cases} \tag{3.73}$$

Next, we proceed to nondimensionalize our system. The same dimensionless parameters for position and time will be used as before:

$$\eta = \frac{r}{r_0} \tag{3.74}$$

$$\zeta = \frac{s}{r_0} \tag{3.75}$$

$$\tau = \frac{\alpha t}{r_0^2} \tag{3.76}$$

As wall temperature is no longer a constant parameter, we introduce a characteristic temperature:

$$T^* = \frac{\Delta h_f}{c_P} \tag{3.77}$$

Using the characteristic temperature, we define our dimensionless temperature, heat generation, and heat flux:

$$\theta = \frac{T - T_m}{T^*} \tag{3.78}$$

$$\dot{Q} = \frac{\dot{q} r_0^2}{k T^*} \tag{3.79}$$

$$Q'' = \frac{q_0'' r_0}{k T^*} \tag{3.80}$$

It is worth noting that the ratio of latent heat to sensible heat is now present in all three of these terms, a contrast from the CST case, where it is only present in the Stefan number. Substituting our dimensionless parameters into Eq. (3.73), our phase change direction can now be governed by:

$$\frac{\dot{Q}}{2} \begin{cases} > Q'' \rightarrow \text{Melting} \\ = Q'' \rightarrow \text{Equilibrium} \\ < Q'' \rightarrow \text{Solidification} \end{cases} \tag{3.81}$$

Substituting dimensionless parameters into Eq. (3.13) and simplifying yields the dimensionless strong form for CSHF:

$$
\begin{aligned}
&\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{liq}(\eta,\tau)}{\partial\eta}\right)+\dot{Q}=\frac{\partial\theta_{liq}(\eta,\tau)}{\partial\tau}, && \text{for } 0\le\eta\le\zeta(\tau);\\
&\left.\frac{\partial\theta_{liq}(\eta,\tau)}{\partial\eta}\right|_{\eta=0}=0,\ \theta_{liq}(\zeta(\tau),\tau)=0,\ \theta_{liq}(\eta,0)=\Phi_{liq}(\eta);\\
&\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{sol}(\eta,\tau)}{\partial\eta}\right)+\dot{Q}=\frac{\partial\theta_{sol}(\eta,\tau)}{\partial\tau}, && \text{for } \zeta(\tau)\le\eta\le 1;\\
&\theta_{sol}(\zeta(\tau),\tau)=0,\ \left.\frac{\partial\theta_{sol}(\eta,\tau)}{\partial\eta}\right|_{\eta=1}=-Q'',\ \theta_{sol}(\eta,0)=\Phi_{sol}(\eta);\\
&\left.\frac{\partial\theta_{liq}(\eta,\tau)}{\eta}\right|_{\eta=\zeta(\tau)}+\frac{d\zeta(\tau)}{d\tau}=\left.\frac{\partial\theta_{sol}(\eta,\tau)}{\partial\eta}\right|_{\eta=\zeta(\tau)}, && \text{for } \tau\ge 0,\ 0\le\zeta(\tau)\le 1;\\
&\zeta(\tau=0)=\zeta_0
\end{aligned}
\tag{3.82}
$$

Liquid Region Solution

Starting with the liquid region and following our solution methodology, the next step is to use superposition to separate the transient and steady-state components, yielding a system very similar to its CST counterpart:

$$
\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{liq,tr}(\eta,\tau)}{\partial\eta}\right)+\frac{1}{\eta}\frac{d}{d\eta}\left(\eta\frac{d\theta_{liq,ss}(\eta)}{d\eta}\right)+\dot{Q}=\frac{\partial\theta_{liq,tr}(\eta,\tau)}{\partial\tau} \tag{3.83}
$$

$$
\begin{aligned}
&\frac{\partial\theta_{liq,tr}(0,\tau)}{\partial\eta}+\frac{d\theta_{liq,ss}(0)}{d\eta}=0,\\
&\theta_{liq,tr}(\zeta(\tau),\tau)+\theta_{liq,ss}(\zeta(\tau))=0,\\
&\theta_{liq,tr}(\eta,0)+\theta_{liq,ss}(\eta)=\Phi_{liq}(\eta)
\end{aligned}
\tag{3.84}
$$

Further separating into distinct subsystems, our steady-state component becomes:

$$
\begin{aligned}
&\frac{1}{\eta}\frac{d}{d\eta}\left(\eta\frac{d\theta_{liq,ss}(\eta)}{d\eta}\right)=-\dot{Q},\\
&\frac{d\theta_{liq,ss}(0)}{d\eta}=0,\\
&\theta_{liq,ss}(\zeta(\tau))=0
\end{aligned}
\tag{3.85}
$$

Eq. (3.85) can be solved for simply by integrating twice and applying boundary conditions, with the following result:

$$
\theta_{liq,ss}(\eta)=\frac{\dot{Q}}{4}\left(\zeta^2(\tau)-\eta^2\right) \tag{3.86}
$$

We then gather the remaining terms of Eq. (3.83) and Eq. (3.84) to create the transient subsystem:

$$\begin{aligned}
&\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{liq,tr}(\eta,\tau)}{\partial\eta}\right)=\frac{\partial\theta_{liq,tr}(\eta,\tau)}{\partial\tau},\\
&\frac{\partial\theta_{liq,tr}(0,\tau)}{\partial\eta}=0,\\
&\theta_{liq,tr}(\zeta(\tau),\tau)=0,\\
&\theta_{liq,tr}(\eta,0)=\Phi_{liq}(\eta)-\theta_{liq,ss}(\eta)
\end{aligned} \tag{3.87}$$

This can be solved using separation of variables. We use the same function designations as before:

$$\theta_{liq,tr}(\eta,\tau)=f(\eta)\cdot g(\tau) \tag{3.88}$$

Differentiating these new functions, substituting into the governing equation, and rearranging to isolate variables, we get:

$$\frac{1}{\eta}\frac{f'}{f}+\frac{f''}{f}=\frac{\dot{g}}{g}=const. \tag{3.89}$$

Both sets of ODEs have common general solutions:

$$\begin{aligned}
f(\eta)&=c_1J_0(\lambda\eta)+c_2Y_0(\lambda_n\eta)\\
g(\tau)&=Ce^{-\lambda^2\tau}
\end{aligned} \tag{3.90}$$

Turning to boundary conditions, $f(\eta)$ must be bounded at $\eta=0$. As $\lim_{\eta\to0^+}(Y_0(\lambda\eta))=\infty$, c_2 must be 0. Applying our second boundary condition gives:

$$f(\zeta(\tau))=c_1J_0(\lambda\zeta(\tau))=0 \tag{3.91}$$

The condition of $c_1=0$ yields a trivial solution. We instead use the condition to define the transcendental equation:

$$J_0(\lambda_n\zeta(\tau))=0 \tag{3.92}$$

which yields the following eigenvalues:

$$\lambda_n=\frac{z_{0,n}}{\zeta(\tau)} \tag{3.93}$$

where $z_{0,n}$ is once again zeros of the Bessel function $J_0(\lambda_n\zeta(\tau))$. We can now combine the results of

Eq. (3.88) through Eq. (3.93) and summing for all n yields the liquid region's transient temperature profile:

$$\theta_{liq,tr}(\eta,\tau) = \sum_{n=1}^{\infty} A_n e^{-\lambda_n^2 \tau} J_0(\lambda_n \eta) \tag{3.94}$$

To solve for the Fourier coefficient A_n, we apply the initial condition for the transient profile:

$$\theta_{liq,tr}(\eta,0) = \Phi_{liq}(\eta) - \theta_{liq,ss}(\eta) = \sum_{n=1}^{\infty} A_n J_0(\lambda_n \eta) \tag{3.95}$$

Isolating A_n once again requires using the orthogonality of eigenfunctions. We multiply both sides by $J_0(\lambda_m \eta)\eta$, integrate across the liquid domain, apply orthogonality principles, and rearrange to yield the Fourier coefficient:

$$A_n = \frac{\displaystyle\int_0^{\zeta(\tau)} \left(\Phi_{liq}(\eta) - \theta_{liq,ss}(\eta)\right) J_0(\lambda_n \eta)\eta d\eta}{\displaystyle\int_0^{\zeta(\tau)} J_0^2(\lambda_n \eta)\eta d\eta} \tag{3.96}$$

Substituting the transient and steady-state profiles from Eq. (3.86) and Eq. (3.94) respectively and referencing Eq. (3.93) and Eq. (3.96) for the associated eigenvalues and Fourier coefficient yields the complete liquid region temperature profile:

$$\begin{aligned} \theta_{liq}(\eta,\tau) &= \sum_{n=1}^{\infty} A_n e^{-\lambda_n^2 \tau} J_0(\lambda_n \eta) + \frac{\dot{Q}}{4}\left(\zeta(\tau)^2 - \eta^2\right), \\ \lambda_n &= \frac{z_{0,n}}{\zeta(\tau)}, \\ A_n &= \frac{\displaystyle\int_0^{\zeta(\tau)} \left(\Phi_{liq}(\eta) - \theta_{liq,ss}(\eta)\right) J_0(\lambda_n \eta)\eta d\eta}{\displaystyle\int_0^{\zeta(\tau)} J_0^2(\lambda_n \eta)\eta d\eta} \end{aligned} \tag{3.97}$$

Solid Region Solution

We now repeat our solution process for the solid region, starting the superposition of the transient and steady-state profiles:

$$\theta_{sol}(\eta,\tau) = \theta_{sol,tr}(\eta,\tau) + \theta_{sol,ss}(\eta) \quad \text{for } \zeta(\tau) \le \eta \le 1 \tag{3.98}$$

Substituted into the solid conduction equation gives:

$$\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\eta}\right)+\frac{1}{\eta}\frac{d}{d\eta}\left(\eta\frac{d\theta_{liq,ss}(\eta)}{d\eta}\right)+\dot{Q}=\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\tau} \tag{3.99}$$

with associated superposed boundary and initial conditions:

$$\begin{aligned}
&\theta_{sol,tr}(\zeta(\tau),\tau)+\theta_{sol,ss}(\zeta(\tau))=0,\\
&\left.\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\eta}\right|_{\eta=1}+\left.\frac{d\theta_{sol,ss}(\eta)}{d\eta}\right|_{\eta=1}=-Q'',\\
&\theta_{sol,tr}(\eta,0)+\theta_{sol,ss}(\eta)=\Phi_{sol}(\eta)
\end{aligned} \tag{3.100}$$

We can further separate the steady-state and transient components into their own subsystems. We start with the steady-state:

$$\begin{aligned}
&\frac{1}{\eta}\frac{d}{d\eta}\left(\eta\frac{d\theta_{sol,ss}(\eta)}{d\eta}\right)=-\dot{Q},\\
&\theta_{sol,ss}(\zeta(\tau))=0,\\
&\left.\frac{d\theta_{sol,ss}(\eta)}{d\eta}\right|_{\eta=1}=-Q''
\end{aligned} \tag{3.101}$$

As before, this can be solved simply by integrating twice and applying boundary conditions, yielding the solid steady-state temperature profile:

$$\theta_{sol,ss}(\eta)=\frac{\dot{Q}}{4}\left(\zeta(\tau)^2-\eta^2\right)+\left(\frac{\dot{Q}}{2}-Q''\right)\ln\frac{\eta}{\zeta(\tau)} \tag{3.102}$$

We form the transient subsystem with the remaining components:

$$\begin{aligned}
&\frac{1}{\eta}\frac{\partial}{\partial\eta}\left(\eta\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\eta}\right)=\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\tau},\\
&\theta_{sol,tr}(\zeta(\tau),\tau)=0,\\
&\left.\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\eta}\right|_{\eta=1}=0,\\
&\theta_{sol,tr}(\eta,0)=\Phi_{sol}(\eta)-\theta_{sol,ss}(\eta)
\end{aligned} \tag{3.103}$$

We use separation of variables, denoting the univariate functions of the solid region as:

$$\theta_{sol,tr}=\tilde{f}(\eta)\cdot\tilde{g}(\tau) \tag{3.104}$$

With the same general solutions as before:

$$\begin{aligned}\tilde{f}(\eta) &= c_3 J_0(\tilde{\lambda}\eta) + c_4 Y_0(\tilde{\lambda}\eta) \\ \tilde{g}(\tau) &= e^{-\tilde{\lambda}^2\tau}\end{aligned} \tag{3.105}$$

Applying our boundary conditions to $\tilde{f}(\eta)$ gives:

$$\begin{aligned}\tilde{f}(\zeta(\tau)) &= c_3 J_0(\tilde{\lambda}\zeta(\tau)) + c_4 Y_0(\tilde{\lambda}\zeta(\tau)) = 0 \\ \left.\frac{\partial\theta_{sol,tr}(\eta,\tau)}{\partial\eta}\right|_{\eta=1} &= -c_3\tilde{\lambda}J_1(\tilde{\lambda}) - c_4\tilde{\lambda}Y_1(\tilde{\lambda}) = 0\end{aligned} \tag{3.106}$$

which uses the following property of Bessel functions:

$$\begin{aligned}\frac{d}{d\eta}J_0(\eta) &= -J_1(\eta), \\ \frac{d}{d\eta}Y_0(\eta) &= -Y_1(\eta)\end{aligned} \tag{3.107}$$

To prevent the trivial solution, we stipulate the following determinant must hold true:

$$\begin{vmatrix} J_0(\tilde{\lambda}\zeta(\tau)) & Y_0(\tilde{\lambda}\zeta(\tau)) \\ \tilde{\lambda}J_1(\tilde{\lambda}) & \tilde{\lambda}Y_1(\tilde{\lambda}) \end{vmatrix} = 0 \tag{3.108}$$

This can be simplified to yield the transcendental equation:

$$J_0(\tilde{\lambda}_n\zeta(\tau))Y_1(\tilde{\lambda}_n) - Y_0(\tilde{\lambda}_n\zeta(\tau))J_1(\tilde{\lambda}_n) = 0 \tag{3.109}$$

As with the CST case, a numerical root finder must be used to determine these values. Reapplying the boundary condition $\tilde{f}'(1) = 0$, we relate the constants c_3 and c_4 to each other:

$$c_3 = -c_4\frac{J_1(\tilde{\lambda}_n)}{Y_1(\tilde{\lambda}_n)} \tag{3.110}$$

With Eq. (3.110), we simplify $\tilde{f}_n(\eta)$ and move the constant c_3 into the solution's Fourier coefficient to obtain the eigenfunctions:

$$\tilde{f}_n = J_0(\tilde{\lambda}_n\eta) - \frac{J_1(\tilde{\lambda}_n)}{Y_1(\tilde{\lambda}_n)}Y_0(\tilde{\lambda}_n\eta) \tag{3.111}$$

Combining the results of Eq. (3.104) through Eq. (3.111) and summing for all n yields the solid region transient temperature profile:

$$\theta_{sol,tr}(\eta,\tau)=\sum_{n=1}^{\infty}B_n e^{-\tilde{\lambda}_n^2\tau}\tilde{f}_n(\eta) \tag{3.112}$$

B_n is solved for by applying the initial condition:

$$\theta_{sol,tr}(\eta,0)=\Phi_{sol}(\eta)-\theta_{sol,ss}(\eta)=\sum_{n=1}^{\infty}B_n\tilde{f}_n(\eta) \tag{3.113}$$

Again taking advantages of the orthogonality of the eigenfunctions with respect to the weight function η, we multiply both sides by $\tilde{f}_m(\eta)\eta$, integrate over the solid region, simplify, and rearrange to yield our Fourier coefficient:

$$B_n=\frac{\int_{\zeta(\tau)}^{1}\left(\Phi_{sol}(\eta)-\theta_{sol,ss}(\eta)\right)\tilde{f}_n(\eta)\eta d\eta}{\int_{\zeta(\tau)}^{1}\tilde{f}_n^2(\eta)\eta d\eta} \tag{3.114}$$

The overall solid region temperature profile can now be obtained by substituting Eq. (3.102) and Eq. (3.112) into Eq. (3.98) and referencing Eq. (3.109), Eq. (3.111), and Eq. (3.114) for the associated eigenvalues, eigenfunction, and Fourier coefficient, respectively:

$$\begin{aligned}
&\theta_{sol}(\eta,\tau)=\left[\sum_{n=1}^{\infty}B_n e^{-\tilde{\lambda}_n^2\tau}\tilde{f}_n(\tilde{\lambda}_n\eta)\right]+\frac{\dot{Q}}{4}\left(\zeta(\tau)^2-\eta^2\right)+\left(\frac{\dot{Q}}{2}-Q''\right)\ln\frac{\eta}{\zeta(\tau)},\\
&J_0(\tilde{\lambda}_n\zeta(\tau))Y_1(\tilde{\lambda}_n)-Y_0(\tilde{\lambda}_n\zeta(\tau))J_1(\tilde{\lambda}_n),\\
&\tilde{f}_n(\eta)=J_0(\tilde{\lambda}_n\eta)-\frac{J_1(\tilde{\lambda}_n)}{Y_1(\tilde{\lambda}_n)}Y_0(\tilde{\lambda}_n\eta),\\
&B_n=\frac{\int_{\zeta(\tau)}^{1}\left(\Phi_{sol}(\eta)-\theta_{sol,ss}(\eta)\right)\tilde{f}_n(\eta)\eta d\eta}{\int_{\zeta(\tau)}^{1}\tilde{f}_n^2(\eta)\eta d\eta}
\end{aligned} \tag{3.115}$$

The Interface Solution

With Eq. (3.97) and Eq. (3.115), we can describe the overall temperature profile for our system for any given time τ and interface position $\zeta(\tau)$. To determine the interface position, we now solve for the interface equation, starting by differentiating both temperature profiles with respect to position:

$$\begin{aligned}\frac{\partial\theta_{liq}(\eta,\tau)}{\partial\eta} &= -\sum_{n=1}^{\infty}\left[A_n\lambda_n e^{-\lambda_n^2\tau}J_1(\lambda_n\eta)\right] - \frac{\dot{Q}}{2}\eta \\ \frac{\partial\theta_{sol}(\eta,\tau)}{\partial\eta} &= \sum_{n=1}^{\infty}\left[B_n\tilde{f}_n'(\eta)e^{-\tilde{\lambda}_n^2\tau}\right] + \left(\frac{\dot{Q}}{2} - Q''\right)\frac{1}{\eta} - \frac{\dot{Q}}{2}\end{aligned} \tag{3.116}$$

where

$$\tilde{f}_n'(\eta) = -\tilde{\lambda}_n \cdot \left(J_1(\tilde{\lambda}_n\eta) - \frac{J_1(\tilde{\lambda}_n)}{Y_1(\tilde{\lambda}_n)}Y_1(\tilde{\lambda}_n\eta)\right) \tag{3.117}$$

Evaluating Eq. (3.116) at $\zeta(\tau)$ and substituting into the interface equation in our strong form, Eq. (3.13), we obtain a closed form ODE for the interface location over time:

$$\frac{d\zeta(\tau)}{d\tau} = \sum_{n=1}^{\infty}A_n\lambda_n e^{-\lambda_n^2\tau}J_1(\lambda_n\zeta(\tau)) + \sum_{n=1}^{\infty}B_n e^{-\tilde{\lambda}_n^2\tau}\tilde{f}_n'(\zeta(\tau)) + \left(\frac{\dot{Q}}{2} - Q''\right)\frac{1}{\zeta(\tau)} \tag{3.118}$$

Setting Initial Conditions

As with the CST case, though the crafted interface solution is valid for arbitrary initial conditions with suitable temperatures for each phase, we wish to use a smooth initial profile. For the melting case, we simply once again use the steady-state conduction equation, setting our centerline temperature $\Phi(0) = 0$. Splitting Φ into liquid and solid regions, yields the following set of initial conditions for melting:

$$\begin{aligned}&\Phi_{liq}(\eta) = 0, \\ &\Phi_{sol}(\eta) = -\frac{\dot{Q}}{4}\eta^2, \\ &\zeta_0 = \epsilon, \\ &\tau = 0\end{aligned} \tag{3.119}$$

The same profile, shifted up so that $\Phi(1) = 0$, can be used for the solidification case. This results in the following set of initial conditions:

$$\begin{aligned}&\Phi_{liq}(\eta) = \frac{\dot{Q}}{4}\left(1-\eta^2\right), \\ &\Phi_{sol}(\eta) = 0, \\ &\zeta_0 = 1-\epsilon, \\ &\tau = 0\end{aligned} \tag{3.120}$$

3.4 Implementation

The analytical solutions crafted are in the form of first-order, closed-form ODEs. Between the complexity of the solutions and the inclusion of series terms, numerical implementation is necessary. To integrate the solutions in time, we use Python, a multi-paradigm programming language with extensive library support for mathematics applications. Python is chosen because of its ease of use, readily available documentation and community forums, and libraries with direct applicability to the problems at hand, including:

- NumPy: An optimized linear algebra library that includes support for array operations and basic mathematical functions. This library serves as a base for many other common Python libraries.
- SciPy: A scientific computation library containing specialized functions for ODEs, integration, root finding, and the family of Bessel functions.
- Pandas: Designed with large-scale data science in mind, Pandas provides robust file I/O support, allowing for seamless reading and writing of solution data.
- MatPlotLib: This library contains functions for graphics development and includes the PyPlot environment, a package designed for easy, thesis-grade plotting with extensive customization and an interactive GUI.

Python implementations of the analytical solutions are included in the appendices.

Chapter 4: Numerical Verification

This chapter describes the process used in building a numerical model to verify the derived analytical solutions. It explores several models that were considered as well as their benefits and shortcomings, then discusses in more depth the solution method ultimately used. Finally, the general case set-up and workflow used in generating solution curves is presented.

Verifying the soundness of a novel analytical solution is often challenging. Experimental validation, while preferred as it compares solutions against observable reality, is not always feasible or even physically possible. Approximate solutions can be useful for ensuring that analytical solution curves are at least correctly shaped, and can provide insight into the contribution of terms or mechanisms that the approximate solution framework neglects. Barannyk et. al used approximate solutions based on quasi-steady state assumptions to compare the analytical solutions derived in this paper in [30] and [31]. However, by their nature, approximate solutions are typically limited to highly idealized scenarios and do not strongly establish the efficacy of more complex analytical solutions.

Numerical solutions and approximation do not provide nearly the same level of confidence in physical accuracy of solutions as empirical methods do, but they offer high mathematical accuracy at a low cost compared to an experimental setup. With the advancement of modern computing power, numerical solutions in the form of CFD and Finite Element Analysis (FEA) have transformed the engineering design process due to their relative speed, ease of setup, and reasonable accuracy.

4.1 Finite Differencing Methods

Two of the most common forms of numerical approximation are the Finite Difference Method (FDM) and Finite Volume Method (FVM). Finite difference schemes approximate differential equations through manipulation of Taylor Series expansions, which take the form [34]:

$$f(x+\Delta x) = f(x) + \Delta x \left.\frac{df(x)}{dx}\right|_x + \frac{(\Delta x)^2}{2!}\left.\frac{d^2 f(x)}{dx^2}\right|_x + \cdots + \frac{(\Delta x)^n}{n!}\left.\frac{d^n f(x)}{dx^n}\right|_x \tag{4.1}$$

For sufficiently small values of Δx, higher order terms become negligible. The lower order terms can then be rearranged to form approximations of derivatives and differential terms. Using multiple series expansions with various spacings, the approximations can be of arbitrarily high order, though in engineering applications, second- or third-order approximations are typically considered suitable.

The core advantage of the FDM is its simplicity. Differential terms can be directly discretized without excess manipulation. There is a visible relation between the terms in a finite difference approximation

and the terms in the original differential equation. For one-dimensional transient and two-dimensional steady-state problems, the system of discretized solutions is often tridiagonal and can be solved efficiently through the Thomas algorithm [34].

These advantages do come at a cost, however. The FDM is typically limited to simple domains. Explicitly solved discretized systems often have prohibitive stability requirements, while more stable implicit systems introduce additional complexity and error through iteration. Finally, finite difference schemes are not conservative for gradient or flux based boundary conditions.

Nonetheless, FDM models are a valuable tool in generating numerical solutions. Three models based on the FDM were considered for verifying the solutions derived in Chapter 3: a fixed-grid, fixed-time step model; a variable-grid, fixed-time step model; and a fixed-grid, variable-time step model. For the sake of scope, system discretization will only be demonstrated for the CST case.

Fixed-Grid Technique

The first of the FDM models attempted was a fixed-grid, fixed-time step model. This model was originally proposed by Crank [35] as an alternative to boundary immobilization techniques in Cartesian coordinates that were prevalent in diffusion problems.

The fixed-grid technique, as its name implies, solves an approximation of the heat equation at discrete points of a fixed, static grid. At most given points in time, the interface position will exist between two grid points. The finite difference formula of the two encompassing points can be modified to incorporate a reduced distance to the interface using a Lagrange-type interpolation.

A benefit of this technique is that it requires only one domain to describe both phases. Another benefit is that equation homogeneity is not required. Using the chain rule, we expand our nondimensional governing equation to:

$$\frac{\partial \theta}{\partial \tau} = \frac{\partial^2 \theta}{\partial \eta^2} + \frac{1}{\eta}\frac{\partial \theta}{\partial \eta} + \dot{Q} \tag{4.2}$$

Replacing our continuous medium with a discretized mesh of nodes, we can denote spatial discretization by the subscript i and spacing $\Delta\eta$, and temporal by j and $\Delta\tau$. Then, using the Taylor series expansion outlined in Equation 4.1, we replace each differential term in Equation 4.2 with a finite difference approximation, starting with the time-dependent term:

$$\left.\frac{\partial \theta}{\partial \tau}\right|_{\eta_i,\tau_j} = \frac{\theta_{i,j} - \theta_{i,j-1}}{\Delta\tau} + \mathcal{O}((\Delta\tau)^2) \tag{4.3}$$

This offers first-order accuracy in time. Higher orders of accuracy can be generated by referencing

multiple previous time steps. However, this comes at the cost of solution complexity and often harsher stability requirements. For spatial discretization, we aim for second-order accuracy. Using a central differencing scheme, our first-order differential term becomes:

$$\left.\frac{\partial\theta}{\partial\eta}\right|_{\eta_i,\tau_j} = \frac{\theta_{i+1,j} - \theta_{i-1,j}}{2\Delta\eta} + \mathcal{O}((\Delta\eta)^3) \tag{4.4}$$

And the second-order differential term becomes:

$$\left.\frac{\partial^2\theta}{\partial\eta^2}\right|_{\eta_i,\tau_j} = \frac{\theta_{i+1,j} - 2\theta_{i,j} + \theta_{i-1,j}}{(\Delta\eta)^2} + \mathcal{O}((\Delta\eta)^4) \tag{4.5}$$

Dropping the big O notation and noting that $\eta = i\Delta\eta$, we substitute Equations 4.3, 4.4, and 4.5 into Equation 4.2. Rearranging we get a first-order in time accuracy, second-order in space finite difference approximation of our governing equation:

$$\frac{\Delta\tau}{(\Delta\eta)^2}\left[\theta_{i-1,j}\left(\frac{1}{2i} - 1\right) + \theta_{i,j}\left(1 + (\Delta\eta)^2\right) + \theta_{i+1,j}\left(-\frac{1}{2i} - 1\right)\right] = \theta_{i,j-1} + \dot{Q}\Delta\tau \tag{4.6}$$

External boundary conditions are given by:

$$\begin{aligned} &\theta_{1,j}\left(1 + \frac{2\Delta\tau}{(\Delta\eta)^2}\right) - \frac{2\Delta\tau}{(\Delta\eta)^2}\theta_{2,j} = \theta_{1,j-1} + \dot{Q}\Delta\tau, \\ &\theta_{I,j} = 0 \end{aligned} \tag{4.7}$$

Where the capital subscript I denotes the final node in the discretized domain. For a problem without phase change, the system of equations described by Equations 4.6 and 4.7 is complete. However, to track the interface and its effects on our solution, we need to make a modification. This modification, as previously alluded, is to substitute the equations near the interface with a three-point Lagrange interpolation [5]. For a given temperature profile $\theta(\eta)$ with known values $\eta = a_0$, a_1, and a_2, we state

$$\theta(\eta) = \sum_{n=0}^{2} l_n(\eta)\theta(a_n) \tag{4.8}$$

Where

$$\begin{aligned} l_n(\eta) &= \frac{p_x(\eta)}{(\eta - a_n)p_2'(a_n)}, \\ p_2(\eta) &= (\eta - a_0)(\eta - a_1)(\eta - a_2), \\ p_2'(a_n) &= \left.\frac{\partial p_2}{\partial\eta}\right|_{\eta=a_n} \end{aligned} \tag{4.9}$$

Equation 4.9 can be differentiated and applied to the discretized governing equation, setting a_2 to the interface position and a_0 and a_1 to the two nodes preceding it. A depiction of this grid setup is shown in Figure 4.1. The model iterates the following steps: The temperature profiles are calculated for a new time step using the interface position of the previous time step and the modified governing equations near the interface. From these profiles, p_2 is calculated. If p_2 remains between the same two nodes, the next time step is calculated. If p_2 travels past the front node, the systems of equations is updated to move the modified equation to the next pair of nodes, and calculation resumes.

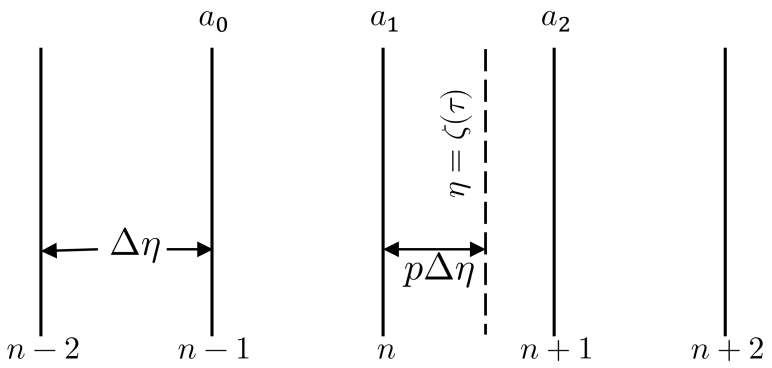

Figure 4.1: Moving boundary position under the fixed-grid method.

While interpolation of the interface position has the advantage of using a fixed grid, it is very unstable at high Stefan numbers. The size of any given time step must be sufficiently small such that the interface does not skip between any set of node pairs. Furthermore, brief testing of this implementation showed poor tracking during node shifts, a possible function of the relatively low order of interpolation. These issues could not be readily resolved, leading to abandoning of this method.

Variable-Grid Technique

The next finite difference model considered used two domains, each with a set number of grid points, that would shrink or expand in size such that the inner boundary of each domain corresponded to the interface location. This model is based off a variable-grid method that was proposed by Murray and Landis [36] and later implemented in cylindrical coordinates by Tien and Churchill [37].

For a given time step, the governing equation is solved in each phase, and the resulting profiles are used to update the position of the interface. With the new interface position, the grid points are redrawn, temperature profiles are remapped to their new locations, and the next step is evaluated. This process is demonstrated in Figure 4.2.

The movement of grid lines necessitates an additional term in our time derivative. For a given grid point at $i\Delta\eta$, partially differentiating with respect to τ yields:

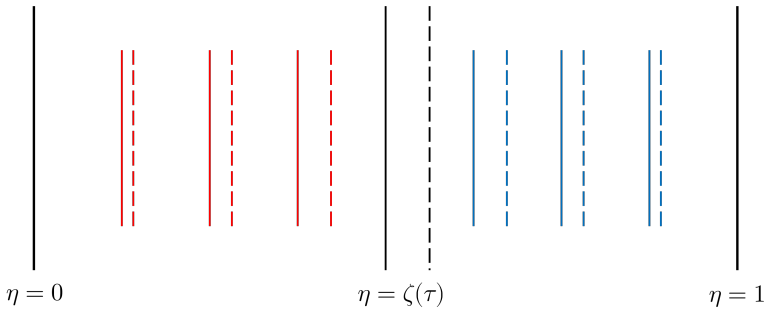

Figure 4.2: Representative grid for the variable-grid method. Red denotes liquid region nodes, blue denotes solid region nodes, and dashed lines denote nodal positions at the next time step.

$$\left(\frac{\partial\theta}{\partial\tau}\right)_i = \left(\frac{\partial\theta}{\partial\eta}\right)_\tau \left(\frac{\partial\eta}{\partial\tau}\right)_i + \left(\frac{\partial\theta}{\partial\tau}\right)_\eta \tag{4.10}$$

If we assume a given grid point's motion can be expressed by:

$$\frac{d\eta_i}{d\tau} = \frac{\eta_i}{\zeta(\tau)}\frac{d\zeta}{d\tau} \tag{4.11}$$

Then our governing equation becomes:

$$\left(\frac{\partial\theta}{\partial\tau}\right)_i = \frac{\eta}{\zeta(\tau)}\frac{d\zeta}{d\tau}\frac{\partial\theta}{\partial\eta} + \frac{\partial^2\theta}{\partial\eta^2} + \frac{1}{\eta}\frac{\partial\theta}{\partial\eta} + \dot{Q} \tag{4.12}$$

Equation 4.12 can be discretized following the same steps as before, with the interface equation being given a similar treatment. The solution process becomes:

1. Solve the numerical system of equations for time step j in the liquid and solid regions.
2. Using the interface equation, update the value of $\zeta(\tau)$.
3. Regenerate the mesh for both domains using the new boundary position of $\zeta(\tau)$.
4. Solve for time step $j + 1$.

While this method reduces the complexity of determining interface position over the fixed grid method, it has several shortcomings of its own. Because the sizing of the grid is dependent on interface location, for either low or high values of $\zeta(\tau)$, the size of the mesh in the liquid and solid domains becomes uneven. Furthermore, a mesh spacing that is sufficient for numerical convergence on a compressed grid may be too coarse when it is expanded, while a spacing that provides convergence for the expanded mesh will be unnecessarily fine when compressed and require very small time steps for accuracy.

Variable-Time Step Method

The final FDM model considered took a different approach from the previous two. While the other models adapted components of their spacial discretization to track the interface, the variable time step method, implemented adapted from the work by Samarskii and Vabishchevich [38], instead modulates time step size such that the interface always lands on a node.

To start, we rewrite our interface balance equation:

$$\frac{\Delta\eta}{\Delta\tau} \approx St \left(\left.\frac{\partial\theta_{sol}}{\partial\eta}\right|_{\eta=\zeta} - \left.\frac{\partial\theta_{liq}}{\partial\eta}\right|_{\eta=\zeta} \right) \tag{4.13}$$

And reorder the heat equation as:

$$\frac{\partial^2\theta}{\partial\eta^2} = \frac{\partial\theta}{\partial\tau} - \frac{1}{\eta}\frac{\partial\theta}{\partial\eta} - \dot{Q} \tag{4.14}$$

Once more denoting the interface node as I, we apply the following second-order approximations:

$$\begin{aligned} \theta_{I+1,j} &\approx \theta_{I,j} + \Delta\eta \left.\frac{\partial\theta_{sol}}{\partial\eta}\right|_{\eta=\zeta} + \frac{(\Delta\eta)^2}{2} \left.\frac{\partial^2\theta_{sol}}{\partial\eta^2}\right|_{\eta=\zeta}, \\ \theta_{I-1,j} &\approx \theta_{I,j} - \Delta\eta \left.\frac{\partial\theta_{liq}}{\partial\eta}\right|_{\eta=\zeta} + \frac{(\Delta\eta)^2}{2} \left.\frac{\partial^2\theta_{liq}}{\partial\eta^2}\right|_{\eta=\zeta} \end{aligned} \tag{4.15}$$

Substituting Equation 4.14 into Equation 4.15 yields approximations of the gradient at the interface:

$$\begin{aligned} \left.\frac{\partial\theta_{sol}}{\partial\eta}\right|_{\eta=\zeta} &= \frac{1}{\Delta\eta - \frac{(\Delta\eta)^2}{2\eta}} \left[\theta_{I+1,j} - \theta_{I,j} - \frac{(\Delta\eta)^2}{2}\frac{\partial\theta_{sol}}{\partial\tau} + \frac{(\Delta\eta)^2}{2}\dot{Q} \right], \\ \left.\frac{\partial\theta_{liq}}{\partial\eta}\right|_{\eta=\zeta} &= \frac{1}{\Delta\eta + \frac{(\Delta\eta)^2}{2\eta}} \left[\theta_{I,j} - \theta_{I-1,j} + \frac{(\Delta\eta)^2}{2}\frac{\partial\theta_{liq}}{\partial\tau} - \frac{(\Delta\eta)^2}{2}\dot{Q} \right] \end{aligned} \tag{4.16}$$

These can now be substituted into Equation 4.13. Consolidating terms and reordering to solve for $\Delta\tau$, we get

$$\begin{aligned} \Delta\tau &= \frac{\frac{\Delta\eta}{St} + \frac{(\Delta\eta)^2}{2}\left(\theta_{I,j} - \theta_{I,j-1}\right)\left(a^* - b^*\right)}{b^*\theta_{I-1,j} + \left(b^* - a^*\right)\theta_{I,j} + a^*\theta_{I+1,j} + \frac{(\Delta\eta)^2}{2}\dot{Q}\left(b^* - a^*\right)}, \\ a^* &= \frac{1}{\Delta\eta - \frac{(\Delta\eta^2)}{2}}, \\ b^* &= \frac{1}{\Delta\eta + \frac{(\Delta\eta^2)}{2}} \end{aligned} \tag{4.17}$$

Where a^* and b^* are coefficients meant to make the equation less cumbersome. Equation 4.17 allows us to approximate the amount of time required for the interface to move from one grid line to the next.

This allows us to use the heat equation discretization given by Equation 4.6 while stipulating that the nondimensional temperature at the interface node is $\theta_{I,j} = \theta_m = 1$. As the time jump is approximated by the previous time steps, an iteration is necessary to correct the time step size.

This method, of the three above, has shown the most promise and has been proven to provide results consistent with our analytical solution [39]. The reframing to treat time as the tracked variable over position simplifies the procedures and is relatively insensitive to the inclusion of an internal heat generation term. However, it too has limitations. While not a prevalent issue within the context of this thesis, the variable time step method is not easily extensible into multiple dimensions. It is also unable to extend to steady-state conditions; as the system approaches steady-state, $\Delta\tau$ tends towards infinity.

4.2 Enthalpy Method Theory

The three methods presented in Section 4.1, share a common characteristic, beyond all being extensions of the FDM. They are all front-tracking methods [5]. Front tracking is the explicit tracking or "capturing" of an interface by predicting its location and modifying the problem domain around the position of said domain. It is a common strategy in FDM schemes as a way to handle discontinuities, such as shock waves in gas dynamics, cracks in fracture analysis, or, in our case, phase change interfaces. The analytical solutions derived in Chapter 3 are, themselves, front tracking solutions. Using a numerical method based on the same principle can thus prove the mathematical soundness of the analytical solution, though it does not prove the soundness of assumptions.

The limitation of front tracking comes from when an interface is not well-defined or smooth. In one-dimensional problems such as ours, this requires the interface exists at single point for any given instant in time; for higher-dimensional problems, it requires that the interface boundary is continuous and smooth. We make the assumption in our problem setup that the phase change region is infinitesimally thin, which satisfies this requirement. In the classical Stefan problem, such an assumption is valid.

In reality, melting occurs across a finite region. Within this region, coined the mushy zone by Tien and Geiger [40], the liquid and solid phases exist simultaneously, and the whole of the material within the region is at melting temperature. Atthey, in researching spot welding, noted that the size of the mushy zone is driven by the magnitude of body heating [41]. Where phase change is driven primarily by internal heat generation in our studied problems, the potential for existence and impact of a mushy zone is inherent.

Since a mushy zone makes it difficult to define a phase change front to track, a method that does not explicitly track the phase change front is desired. One such method is to introduce an enthalpy function. Enthalpy, a thermodynamic state variable that measures overall heat content, is useful for phase change

because while temperature across the phase change front is constant, enthalpy is not. Rather, enthalpy increases with the liquid fraction within the mushy zone. Using the enthalpy method allows for implicit tracking of the entire mushy zone solely through enforcement of flux within the domain.

There are a number of ways to implement the enthalpy method [13]. The version we consider is an enthalpy-porosity method, developed by Voller and Prakash [42]. This method provides several benefits:

- Because the interface is not explicitly tracked, both phases do not need to exist concurrently. This allows for modeling before, during, and after the phase-change process in one model.
- The model is not limited to phase change occurring at a single temperature. A phase-change temperature range, such as what exists in binary metal alloys, can be used.
- Fluid motion is incorporated into the governing equation, allowing for convection and dendritic formation due to pull velocities to be modeled.
- The enthalpy-porosity method is easily extensible to more complex geometries and conditions.

These benefits come at a cost: the enthalpy-porosity model is significantly more complex and is not conducive to closed-form analytical solutions. As a numerical solution, it is prone to grid-induced error, as any region of phase change has an artificially enforced minimum size of the cell it exists within. Nonetheless, the model is a powerful tool for modeling phase-change phenomena.

The enthalpy-porosity model uses a variation of the energy equation:

$$\frac{\partial}{\partial t}(\rho H) + \nabla \cdot (\rho \vec{v} H) = \nabla \cdot (k \nabla T) + S \tag{4.18}$$

where H is a sum of the sensible heat, h, and the latent heat, ΔH; ρ is the material density; $\vec{v}$ is the velocity field; and S is the sum of all source terms. We can relate sensible enthalpy to temperature through specific heat:

$$h = h_{ref} + \int_{T_{ref}}^{T} c_P dT \tag{4.19}$$

And we relate latent heat to the heat of fusion Δh_f and liquid fraction β:

$$\Delta H = f(T) = \begin{cases} \Delta h_f & \text{for } T > T_m, \\ \Delta h_f \beta & \text{for } T = T_m, \\ 0 & \text{for } T < T_m \end{cases} \tag{4.20}$$

For pure metals, liquid fraction is solved for by inverting Eq. (4.20) and multiplying by c_P:

$$h = c_P \cdot f^{-1}(\Delta H) \tag{4.21}$$

Velocity field can be solved for using an appropriate set of flow equations. When they are used, a velocity sink term is added to material that is solid or mushy. For the purposes that we intend to use the enthalpy-porosity method, we can instead assume a uniform zero velocity field, precluding the need for coupled solving methods.

4.3 Implementation in Fluent

The reason for using the enthalpy-porosity model, despite its higher complexity and underutilized features relative to tailor-made numerical models, is that code for it is commercially available and well tested. Ansys Fluent is a CFD package wrapped within the Ansys Workstation framework, with capabilities to study laminar and turbulent flows, chemically reacting flows, multi-phase and fluid-structure based models, and, of course, melting and solidification, all in multiple dimensions. Furthermore, it allows for quick modification to solution methods, such as spatial and temporal order, numerical solvers, and relaxation factors.

Using a commercial software for research does have limitations. The actual code implementation of solvers is typically proprietary, and results in something of a "black box" solution; while solver settings can be modified, the true coding operations are hidden behind a Graphical User Interface (GUI). The GUI simplifies setup, particularly for users who may not have direct experience with numerical methods from the conceptual side, but it also can restrict awareness of what is truly happening. Many commercial softwares are also limited to lower-order implementations, since they are geared towards commercial settings, where expediency and stability are typically prioritized over absolute accuracy.

The final major issue of commercial solvers is more specific to our case than the general issues described above. Ansys Fluent and most CFD software are optimized for two- and three-dimensional problems involving fluid flow. These types of problems require a fair amount of computational overhead in the form of bookkeeping and stability checks. While this often results in higher speed and accuracy over open-source and handmade codes in complex flow problems [43], for simple domains such as ours, the resulting computational overhead leads to longer solution times. Ultimately, while these issues were present, the benefit and simplicity of using Ansys Fluent outweighed the costs for this work.

Cases in Ansys Workstation follow a predefined path, broken into 6 steps:

1. Analysis System: Choose the general type of numerical analysis and software package to use. Ansys comprises over a dozen software packages ranging in specialization from force buckling models to

internal combustion simulation. In our case, we choose fluid flow through Fluent.

2. Geometry: Create the physical domain the case exists in. Here, dimensional order is chosen, physical boundary conditions, such as planes of symmetry, axes of axisymmetry, and walls are defined, and unit systems are identified.

3. Mesh: Discretize the continuous domain(s) into a meshed network of nodes and elements. This step effectively converts the case we wish to analyze from the physical world defined by continuum mechanics and calculus into the computational world of algebra and numerics.

4. Setup: Determine application specific settings for the case, choose physics and flow models, define material properties, and choose numerical solvers. This step determines if an analysis will be steady-state or transient

5. Solution: Solve the numerical system. Determine convergence conditions, what values to write to an output file or plot, how often to write out said values, and if numerical conditions (such as continuity) must be preserved or solved. The solution step provides some limited visualization capabilities, including plots and contours that can be updated in real-time.

6. Results: Provide advanced result visualizations and output. This step can generate animations of transient effects, compare point-to-point changes, and create user-specified functions to compare against. For our solutions, we exported data and generated our solution plots through external codes, so this step is omitted.

Geometry and Mesh

Our problem domain, for both the CST and CSHF cases, is an infinitely long cylinder. A three-dimensional domain of infinite length is not easily or efficiently represented numerically, but since we are operating on similar assumptions to those presented in Chapter 3, we can reduce the computational domain to one dimension. First, we assume that our field equations are uniform for any given disc of the cylinder. This allows us to use a symmetry boundary condition on the top and bottom faces of our model, reducing it to an effectively two-dimensional disc with an arbitrary, unused height. Next, we can impose an axisymmetry boundary condition on the innermost edge of our mesh. The resulting domain is an arbitrarily thin rectangle, with the left boundary representing the centerline, the right boundary representing the wall, and the upper and lower boundaries identical to each other through symmetry. Since the scaling of geometry can be set to any arbitrary value so long as our nondimensional parameters

match those used in Chapter 3, we can ease postprocessing work by setting the constant parameters logically. As Fluent by default uses SI units, we set the radius as $r_0 = 1$ m.

This reduced geometry can be represented by a $n \times 1$ mesh of rectangles. A uniform grading was used such that each of the rectangles have uniform size. The uniform mesh was chosen over a centerline or wall-biased scaling since it provides consistent accuracy throughout the entire domain for both melting and solidification cases and both prescribed surface temperature and prescribed surface heat flux boundary conditions.

Discretization of differential equations introduces solution error due to the truncation of higher-order terms. This error is proportional to the size of a given discretized element raised to the power of the lowest-order term omitted. A model is considered "grid independent" when the truncation error is less than computer round-off error [44]. As solution complexity and stability requirements increase with discretization level, it is valuable to determine where a model becomes grid independent for its values of interest and discretize no further, maximizing accuracy while minimizing computational difficulty.

Grid independence is dependent on a number of factors, including discretization order, dependent variable observed, and any smoothing and iterative corrections incorporated by the numerical solver. For the case at hand, a 200×1 mesh was deemed sufficiently grid independent for interface position. This level of discretization offered a balance between solution accuracy and computational speed.

Setup

We use default launch settings for Fluent, with the exception of precision and solver processes. A double precision variable is used for calculations to increase precision of solutions, and four CPU cores are used to increase solution speed. Fluent is unable to perform true one-dimensional analysis, but we are able to approximate it through a two-dimensional, axisymmetric case with symmetry boundary conditions on the upper and lower walls to emulate an infinitely uniform dimension. The inner wall is given an axis boundary condition, and the outer a wall boundary condition.

The Fluent framework automatically assigns units and performs dimensional consistency checks, with a default assignment to SI units. It additionally performs basic sanity checks throughout calculations, including a temperature check to prevent values below absolute zero. While generally helpful, these systems have unintended consequences for our model, where the goal of consistency in dependent values with their nondimensional counterparts is desired. Fortunately, the nondimensional temperature formulations in both the CST and CSHF cases utilize a temperature difference and are, therefore, relative. This allows us to use the Celsius temperature scale to obtain dependent variable values equal to their nondimensional counterpart.

We now determine the thermophsycical properties of our domain. Materials in Fluent are divided into two types, one representing "fluid" materials and one representing "solid" materials. These designations are somewhat misleading; solid materials represent the material properties of domain boundaries, while fluid materials represent the material properties within the internal field of an element. In our model, both the liquid and solid region properties are defined by the fluid material, and only the outer wall's properties are defined by the solid material. We will assign the same values to both material types. Three properties are defined for solid materials:

- Density, $\rho \left[\frac{\text{kg}}{\text{m}^3}\right]$
- Specific heat, $C_p \left[\frac{\text{J}}{\text{kg-K}}\right]$
- Thermal conductivity, $\kappa \left[\frac{\text{W}}{\text{m-K}}\right]$

In addition to these properties, fluid materials require the following material definitions:

- Dynamic viscosity, $\mu \left[\frac{\text{kg}}{\text{m-s}}\right]$
- Pure solvent melting heat, $\Delta h \left[\frac{\text{J}}{\text{kg}}\right]$
- Solidus temperature, T_{sol} [C] (CST) / [K] (CSHF)
- Liquidus temperature, T_{liq} [C] (CST) / [K] (CSHF)

Dynamic viscosity, while required to be defined, will not actually be used in our model as no fluid motion is occurring. Pure solvent melting heat is equivalent to the heat of fusion, Δh_f. In the CST cases, it will be used to modulate the Stefan number through the similarity

$$St \sim \frac{1}{\Delta h} \tag{4.22}$$

For the CSHF cases, where a global Stefan number is not applicable, we will leave the melting heat at 1. Finally, we will address solidus and liquidus temperatures. For isothermal phase change, these two values are equal and equivalent to the melting temperature, T_m. We will set $T_m = 1$ and the wall temperature $T_0 = 0$. This sets dimensional and nondimensional temperature as equivalent to each other:

$$\theta \sim T \tag{4.23}$$

We next define the internal heat generation source term. Inputting our material properties into Equation 3.25 and Equation 3.79, for both sets of cases, we are able to relate dimensional and nondimensional internal heat generation by:

$$\dot{Q} \sim \dot{q} \tag{4.24}$$

The model now only requires value assignment of the wall boundary condition. As previously discussed, for the CST cases, we set the wall boundary condition to a prescribed temperature of $T_0 = 0$. For the CSHF case, inputting our material properties into Equation 3.80 gives us another direct relation:

$$Q'' \sim q'' \tag{4.25}$$

Solution

Having established the physical properties of our model, it is now time to explore its computational properties, starting with discretization methods employed on the various field variables. This includes gradient discretization, energy equation discretization, and transient formulation. Gradient settings represent the location of field variable values and the method of interpolation between such values. Gradient can either be measured by cell center or by node, and interpolated through the least squares method or Green-Gauss interpolation. The chosen setting, least squares cell based, provides fairly high accuracy at low computational cost.

Energy is discretized by a second-order upwind scheme. This differs from the centered second order discretization demonstrated in Equation 4.5 by biasing both points of interpolation in one direction. Upwind schemes are common in flow models since centered schemes can created nonphysical checkerboard fields. The one-dimensional nature of our model precludes the need for an upwind scheme, but is not likely to suffer as a result of using it. At the time of writing, centered schemes are not available in the Fluent settings.

For our transient formulation, we use a first-order implicit discretization, similar to that presented in Equation 4.3. First-order methods tend to be more stable and faster to solve, if less accurate than their higher-order counterparts. Comparisons between first-order and second-order schemes for transient discretization showed negligible changes in the solution curves for our model, so the first-order scheme was adopted to decrease solution time.

A fixed time step size of $5 \cdot 10^{-5}$ was used to keep the Courant number low. The Courant number is a general indicator of model stability that represents the relative speed at which data moves through a discretized element. In heat transfer and diffusion problems, for a given diffusivity α, spatial discretization Δx, and time step Δt, the Courant number is given by [44]:

$$C = \frac{\alpha \Delta t}{(\Delta x)^2} \tag{4.26}$$

For explicit models, a Courant number greater than 1 means that heat data is able to effectively skip elements, leading to divergent solutions in explicit models. Our model is implicit, precluding the need to satisfy the Courant number requirement. It is, nonetheless, good practice to maintain a low Courant number in order to preserve accuracy. Our model maintains a Courant number of $C = 2$.

The in-built implicit solvers of Fluent allow for over- and under-relaxation factors to be applied between iterations. These factors enable a trade-off between convergence speed and stability. An under-relaxation factor of 0.9 was applied to the energy equation solver; brief manipulation of the factor did not yield significant changes in convergence speed. We use an absolute convergence criteria of $res_{abs} = 10^{-10}$. This provides reasonable convergence speed without compromising accuracy.

A discussion on how the interface position is computed is required here. As previously mentioned, the enthalpy method employed by Fluent does not explicitly track interface position. Additionally, the creation of a mushy zone inherently makes defining its position somewhat vague. Tracking the final domain position that is fully solid will yield a different interface evolution than for tracking the initial domain position that is fully liquid; this difference will be exacerbated if the size of the mushy zone is large. An alternative method that mitigates this issue is found through average liquid fraction.

The Area Weighted Average (AWA) liquid fraction numerically integrates the liquid fraction, β, of each element across its area, averaging over the entire domain area. This value is effectively equivalent to nondimensional interface position for a non-mushy model, though it needs to be converted from an area to a radial position through the relation:

$$\zeta = \sqrt{\beta_{AWA}} \tag{4.27}$$

While Fluent does have the ability to report user-defined functions such as this, its functionality is limited. We opted instead for a data export and postprocessing through external Python scripts, inline with the process used for generating analytical solution data. Temperature and enthalpy data was exported at uniform time steps to comma-separated text files containing node number, radial position, static temperature, and enthalpy for each requested time step. This data can be accessed through a relevant I/O script and plotted against the analytical profiles through Python.

Initialization and calculation runs required a staged approach in order to allow for naturally-enforced temperature profiles consistent with the initial profiles used in the analytical solutions. Fluent initialization is geared towards uniform initial fields. However, field variable values in models are preserved

between simulation runs. Keeping this in mind, the following process was adopted:

1. Begin model as a steady-state analysis.
2. Define the material properties and internal heat generation value for the case.
3. Set model methods, controls, and convergence settings.
4. Set the wall boundary condition to prescribed temperature. For solidification cases, use T_m. For melting cases, use $T_m - \dot{Q}/4$.
5. Initialize the model. For melting cases, use an initial temperature value slightly less than T_m and vice-versa for solidification cases. This ensures no mushy values in the initial solution.
6. Calculate the steady-state solution.
7. Switch the model to transient analysis. Change boundary conditions and internal heat generation to match intended case.
8. Create the interface report definition and data export rule.
9. Solve to the required time.

Chapter 5: Results

The analytical solutions derived in Chapter 3 are compared against the solutions generated through the Ansys Fluent implementation described in Chapter 4. Both temporal evolutions and spatial evolutions are presented. Possible causes of deviation between the two solution methods are discussed, and key behaviors of the systems are characterized. This chapter is split into two main sections. The first presents results for the CST case, and the second for the CSHF case. For each, we first present the results of melting scenarios then the solidification scenarios.

5.1 Constant Surface Temperature Results

The CST case provides two dimensionless parameters that can be manipulated to affect the relative speed of phase change: $\dot{Q}$ and St. As $\dot{Q}$ also impacts temperature profiles, we opt to keep $\dot{Q}$ constant across cases and modulate St in our comparisons. We established in Equation 3.28 that $\dot{Q}$ must be greater than 4 for solid and liquid phases to exist at steady-state. As will be discussed in a later section, values of $\dot{Q}$ significantly higher than 4 lead to nonphysical solid region temperature profiles in the analytical model. A value of $\dot{Q} = 5.0$ was ultimately used, as it provides a middling steady-state interface position of $\zeta_{ss} \approx 0.447$ without significantly nonphysical effects. Four values of Stefan number are used: $St = 0.01, 0.1, 1.0$, and 10.0. This range of values represents very slow interface motion at $St = 0.01$, where the quasi-steady approximations made by Crepeau and Siahpush [27] are valid, through extremely fast interface motion, where the validity of the solutions generated in Chapter 3 is challenged.

Melting Results

We begin with comparing interface evolutions in time for melting. Figure 5.1 presents these values. The solutions generated from our work in Chapter 3 are represented by solid lines and denoted by an "Analytical" label in the legend. Those generated through Fluent are represented by the dotted lines and denoted by the "Numerical" label. Note that the symbols do not represent all time steps used by the numerical model.

At low Stefan numbers, we see good agreement between the two models, with the analytical model predicting slightly slower interface motion than the numerical. As the Stefan number increases and the phase change front moves faster, the disparity between profiles flips, with the analytical model achieving steady-state behavior faster. At $St = 10.0$, we see the analytical solution has a significantly steeper profile prior to steady state. For both sets of cases, we see the interface approaches the predicted steady-state position of $\zeta_{ss} \approx 0.447$.

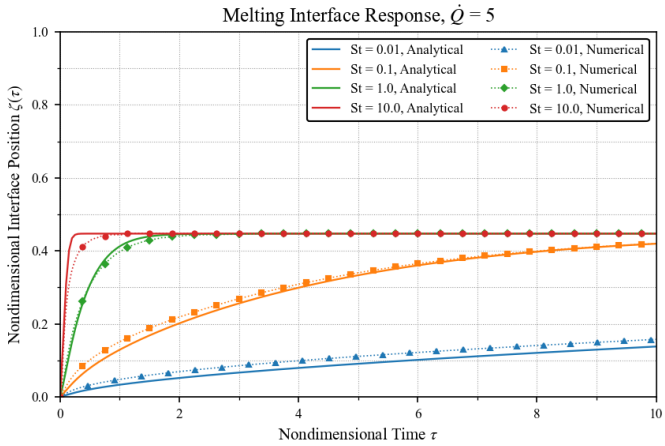

Figure 5.1: CST interface position as a function of time during melting at $\dot{Q} = 5.0$ for various values of St.

The believed reason for the numerical solution's over-prediction at lower Stefan numbers and under-prediction at higher is two competing mechanisms. At low Stefan numbers, the relative amount of stored energy in the form of phase change is very large compared to the amount required to change temperature outside the mushy zone. This allows the size of the mushy zone to grow relatively quickly, artificially increasing the speed of the averaged interface. As the Stefan number increases, energy stored in the mushy zone decreases, reducing the impact.

The under-prediction of the numerical model at higher Stefan numbers is believed to be an artifact of numerical diffusion. Numerical models, particularly of lower order, tend to exhibit a "smoothing out" of sharp profiles. Numerical diffusion can be rectified by increasing discretization order to include higher-order terms, but this often results in less stable models and oscillatory values. An alternative improvement would be to implement an adaptive mesh that increases resolution within and nearby the phase change region.

Figure 5.2 shows the analytical solution temperature profiles as a function of position at various time steps for the $St = 0.01$, melting case. The numerical analog, to be discussed below, is shown in Figure 5.3. The temperature profiles for analytical cases are generated by calculating the interface position, $\zeta(\tau)$, through Equation 3.65, and inputting that position into Equation 3.45 and Equation 3.62.

For the temperature plots, the dashed line, labeled θ_m in the legend, represents the nondimensional melting temperature. Above this line, the material is expected to be fully liquid, and below, fully solid. The point at which the curve first intercepts θ_m corresponds to the nondimensional interface position, $\tau(\zeta)$, as referenced in Figure 5.1. Thus, the domain to the left of this point is liquid, and the domain to the right is solid. We see in Figure 5.2 an unexpected phenomenon: The solid region curves near the

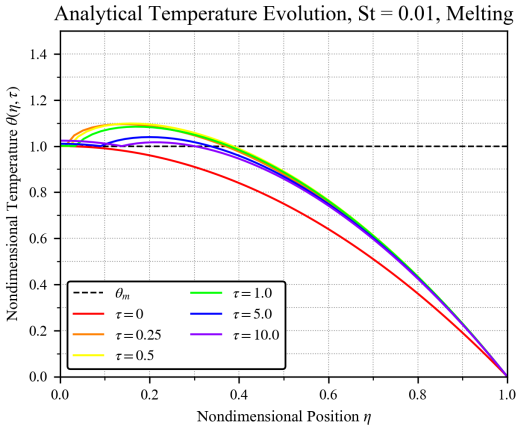

Figure 5.2: Analytically generated temperature profiles for the CST melting case at various time steps with $\dot{Q} = 5.0$, $St = 0.01$.

interface extend above θ_m. This nonphysical "overheating" phenomenon is believed to be a consequence of neglecting the mushy zone.

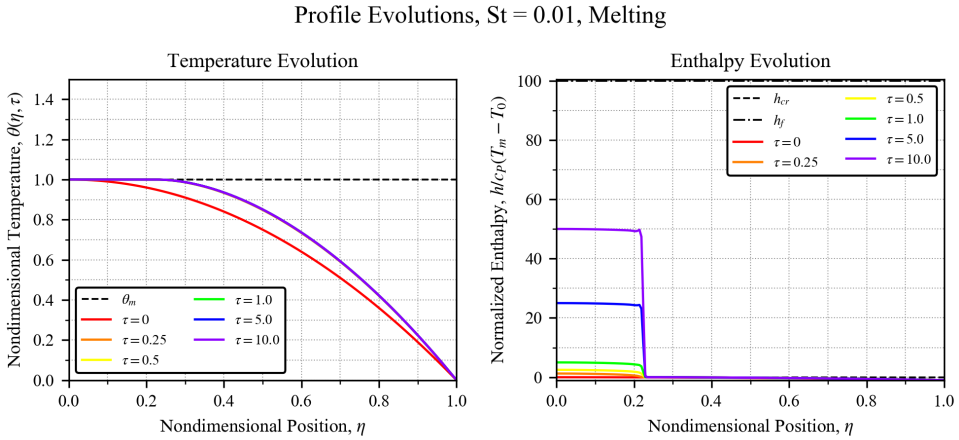

Figure 5.3: Numerically generated profiles for the CST melting case at various time steps with $\dot{Q} = 5.0$, $St = 0.01$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Figure 5.3, as well as the rest of the figures showing profiles for the numerical model, is split into two graphs. The lefthand plot shows nondimensional temperature as a function of nondimensional position for various time steps, similar to the analytical profiles. The horizontal dashed line again represents the nondimensional melting temperature, θ_m.

On the righthand side, the same time steps are used, but temperature is replaced with a normalized enthalpy value, given by:

$$Normalized\ Enthalpy = \frac{h}{c_P(T_m - T_0)} \tag{5.1}$$

Instead of a single horizontal line representing the point of phase change, the enthalpy evolution plots contain two. The lower line, labeled h_{cr}, represents the critical enthalpy value below which the material is fully solid. The upper line, labeled h_f, represents the fusion enthalpy value, above which the material is fully molten. The gap between these two values is proportional to the heat of fusion, Δh_f, and represents that amount of energy that a material stores as it undergoes phase change. Just as water on a burner will maintain a steady temperature as it begins to boil, a solid material in this stage of the phase change process will also maintain a steady temperature until it has become fully molten.

Figure 5.3 presents the numerically generated temperature and enthalpy profiles at $St = 0.01$ for the melting case, using the same set of time steps as those used in Figure 5.2. At first brush, it appears only two time steps, $\tau = 0$ and $\tau = 10.0$, are plotted. In actuality, the temperature profiles of the intermediate time steps are nearly identical to that of $\tau = 10.0$. This is more obvious in the context of the enthalpy profiles. For all time steps other than $\tau = 0$, there is a plateau at $\eta \approx 0.225$ before which the material is in a mushy, and therefore isothermal, state.

The cause of this plateau derives from the isothermal nature of phase change. Conducting an energy balance at the outer end of the mushy zone, there are two sources: the internal heat generation source and an outward flux into the fully solid region. Because the mushy zone is isothermal, there is no driven heat flux, until a portion of the material becomes fully liquid and can change temperature again. This leads to a temporarily static temperature profile.

The profile developments of the mushy zone may also explain the overheating phenomenon exhibited by the analytical model. The period of greatest overheating occurs at earlier time steps, where the enthalpy in the mushy zone varies with position; later time steps show a uniform enthalpy value in the mushy zone followed by a step change in value at the region boundary. This suggests that the impact of the mushy zone is greatest during its initial formation.

We look next at the case of melting at $St = 0.1$. The temperature profiles for the analytical solution to this case are presented in Figure 5.4. This case, like the $St = 0.01$ case, does not reach steady-state in the window of time used. However, we see with later time steps the temperature profiles tend towards a parabolic curve with a more seamless transition between the liquid and solid regions.

We also see less of the overheating phenomenon in this model compared to Figure 5.2. This, however, is more likely a consequence of the time steps used. As the Stefan number effectively represents interface speed in our model, the order of magnitude change from $St = 0.01$ to $St = 0.1$ results in a commensurate

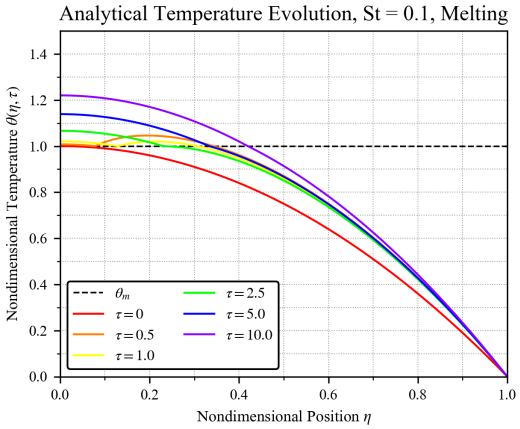

Figure 5.4: Analytically generated temperature profiles for the CST melting case at various time steps with $\dot{Q} = 5.0$, $St = 0.1$.

change in temperature profiles. To demonstrate, if we compare the curve of $\tau = 5.0$ in Figure 5.2 to $\tau = 0.5$ in Figure 5.4, we see they are virtually identical. This suggests that the amplitude of overheating has not changed; only the period of time over which it occurs has.

Figure 5.5 shows the temperature and enthalpy profiles for the $St = 0.1$ melting case using the numerical model. Focusing first on the temperature profiles, we see that once more, for time steps $\tau = 0.5$ and $\tau = 1.0$, where there is no fully liquid material, the profiles are identical. The remainder of the temperature profiles match very closely with those of Figure 5.4.

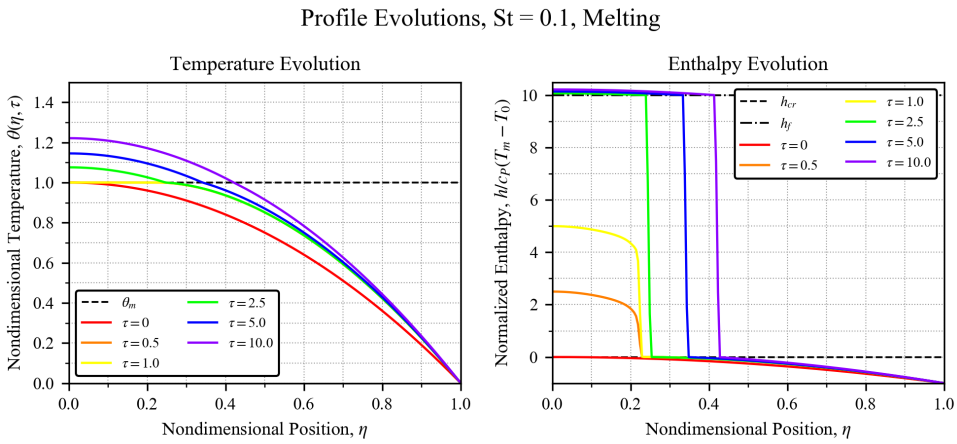

Figure 5.5: Numerically generated profiles for the CST melting case at various time steps with $\dot{Q} = 5.0$, $St = 0.1$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Referring to the enthalpy plot, the reduced gap between h_{cr} and h_f allows us to see more of the development of the mushy zone. At the lower time steps we can see there is not a sharp drop at the edge of the zone. By the time the fully liquid region has formed, the step drop from h_f to h_{cr} typical of a

sharp interface exists. This reinforces the idea that the overheating phenomena exhibited at early time steps in the analytical models is related to the models' lack of capturing the formation, expansion, and eventual collapse of the mushy zone region. It also appears the amplitude of the overheating is related to the shape of the enthalpy profile within the mushy zone, though the exact mechanics are not understood.

Figure 5.6 shows the analytical temperature profiles at $St = 1.0$. This range represents the higher end of Stefan number values for which the quasi steady-state assumption is valid. Note that a different series of time steps is presented for this case than for the $St = 0.01$ and $St = 0.1$ cases, as steady-state is reached much faster. We can see that by the final plotted time step of $\tau = 2.0$, the model has nearly reached steady-state, as evidenced by the similarity in profiles between $\tau = 1.0$ and $\tau = 2.0$. We also see that any overheating phenomena has been all but eliminated by $\tau = 0.25$, showing that the period where the impact of neglecting the mushy zone is transient and dependent on the Stefan number.

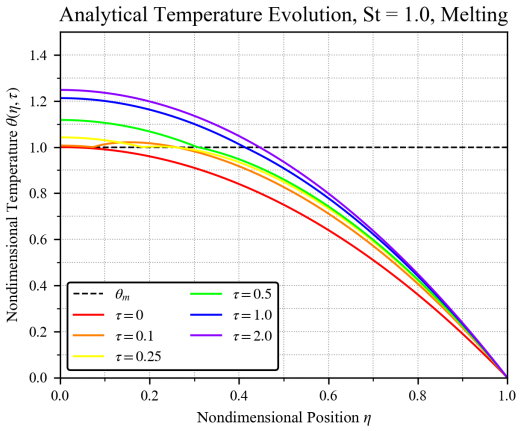

Figure 5.6: Analytically generated temperature profiles for the CST melting case at various time steps with $\dot{Q} = 5.0$, $St = 1.0$.

The numerical model results for this case are shown in Figure 5.7. Similar to the overheating in Figure 5.6, the isotherm from the mushy zone is present at $\tau = 0.1$ but has been eliminated by $\tau = 0.25$. Overall, the temperature profiles track quite well in relation to those presented in the analytical model, presenting a strong case for the validity of our analytical model in this range of interface speed.

An interesting observation is find in relation to the enthalpy plots for this case: We can see at $\tau = 0.25$ that though a plateau has begun to form, it is not yet vertical. This matters because the profile for $\tau = 0.25$ has a liquid region, which demonstrates the existence, and therefore impact, of the mushy zone is not limited to the period of time prior to formation of a liquid region. A partial cause for this comes from the continued reduction in the gap between h_f and h_{cr} relative to the previous cases; the profiles formed in this region are now on a similar scale to those outside the region.

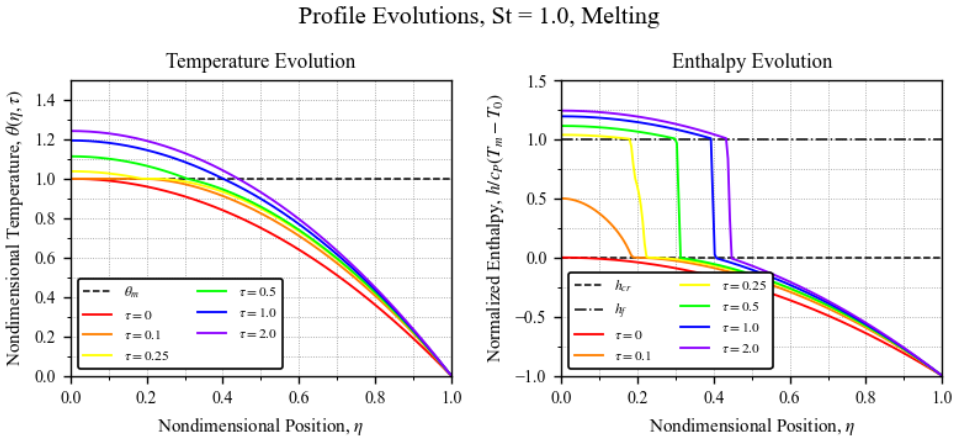

Figure 5.7: Numerically generated profiles for the CST melting case at various time steps with $\dot{Q} = 5.0$, $St = 1.0$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

The case shown in Figure 5.8 represents the fastest interface motion in our analysis, with an increase in the Stefan number by a factor of 1000 over our first set of results. At this rate, the quasi steady-state assumption is no longer valid. We can see that by $\tau = 0.05$, our earliest time step, any possible overheating effects have dissipated. Looking at the enthalpy plots for the numerical version of this case in Figure 5.9, we can see the gap between h_f and h_{cr} is much smaller than the overall range of the rest of the enthalpy profile. The significance of any profile that develops in the mushy zone, and as a result, any overheating phenomena, is thus much less than for lower Stefan number cases.

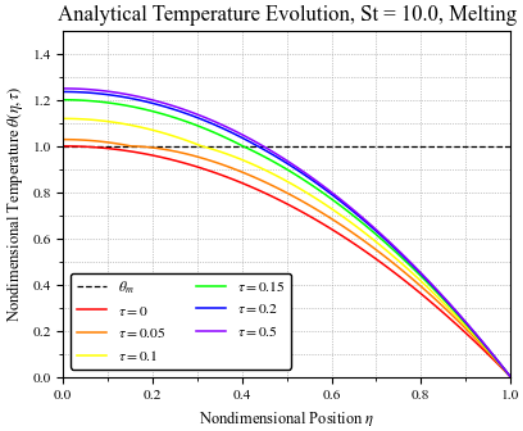

Figure 5.8: Analytically generated temperature profiles for the CST melting case at various time steps with $\dot{Q} = 5.0$, $St = 10.0$.

Figure 5.8 shows that steady-state is achieved very quickly, as there is very little change in profile between $\tau = 0.2$ and $\tau = 0.5$. This is not quite the case in the numerical model, shown in the lefthand plots of Figure 5.9. Here, the temperature profiles now lag visibly from the analytical case. This is to

be expected, given the difference in the interface evolutions for these two cases from Figure 5.1. Aside from the lag in interface position, the agreement in temperature profiles is good between the numerical and analytical model, is there is very little impact from overheating or a mushy zone. We do see in the enthalpy plot that the plateau from h_f to h_{cr} never becomes entirely vertical. This may simply be a consequence of using a finite volume method; the plateau will always span at least the width of one discretized element. However, it may also be a result of the speed of the interface.

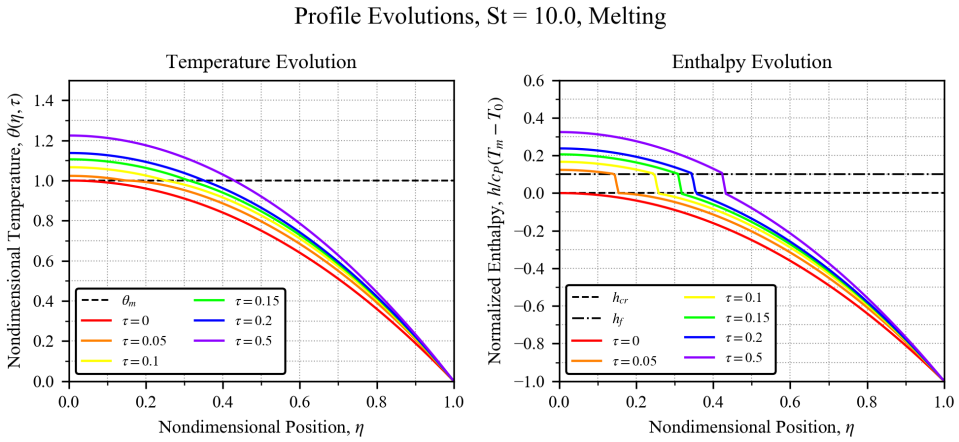

Figure 5.9: Numerically generated profiles for the CST melting case at various time steps with $\dot{Q} = 5.0$, $St = 10.0$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Solidification Results

We now turn to solidification cases for the CST problem. The same notations will be used as with the melting cases, as will the same values of $\dot{Q}$ and St; as such, we will expect similar behavior in the models at steady-state, and roughly equivalent interface speed. Figure 5.10 compares the analytically and numerically generated interface positions as a function of time for the four solidification cases.

Similar to the melting case, we see strong agreement in the interface position for low Stefan numbers and progressively weaker agreement as the Stefan number increases. In all cases, the interface asymptotically approaches the expected steady-state position of $\zeta \approx 0.447$. However, some of the behavior differs from the melting case. There is no initial over-prediction in interface speed for the numerical model at lower Stefan numbers, and the disagreement between the two models is more significant for the higher Stefan number cases than it was for the same Stefan number melting cases.

Figure 5.11 presents the temperature profiles at various time steps for $St = 0.01$. Unlike in the melting case, the wall boundary condition for our initial profile is different from the rest of the time steps. This step change in boundary condition results in an immediate change in slope at the interface for earlier

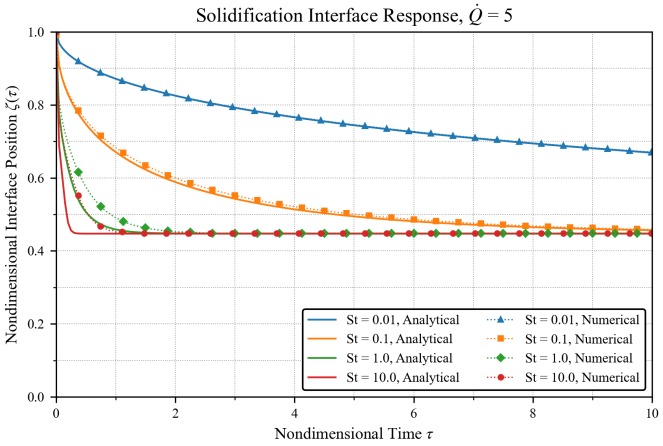

Figure 5.10: CST interface position as a function of time during solidification at $\dot{Q} = 5.0$ for various values of St.

time steps, followed by a smoothing in the profiles over time. Another key difference from the melting case is that we do not see any overheating or otherwise nonphysical phenomena.

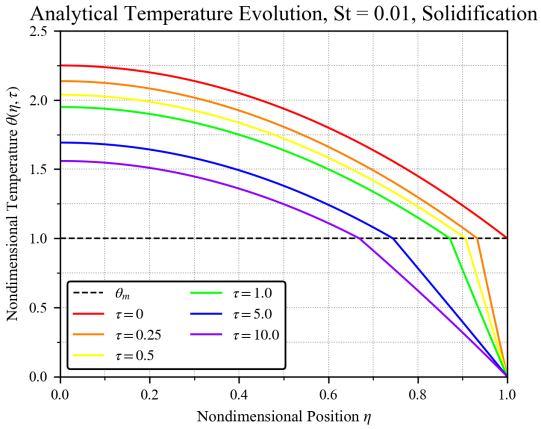

Figure 5.11: Analytically generated temperature profiles for the CST solidification case at various time steps with $\dot{Q} = 5.0$, $St = 0.01$.

The results for the numerical model at $St = 0.01$ are presented in Figure 5.12. Just as the agreement in interface evolution was very strong for the analytical and numerical model for this case, the temperature profiles are virtually identical. We again see a step change in wall temperature after the initial profile leading to a change in slope at the interface that smooths over time.

Looking to the enthalpy plots, we see part of the reason for the agreement: from the outset, the profile between h_f and h_{cr} is a vertical drop. No mushy region exists, and the sharp interface assumption appears entirely valid. The reason no mushy zone exists for the solidification cases is uncertain. It may

be a consequence of the numerical model's method for determining enthalpy values in the mushy zone. Additionally, work done by Yao et. al [45] suggests that the solidification process is more stable and less prone to mushy zone development.

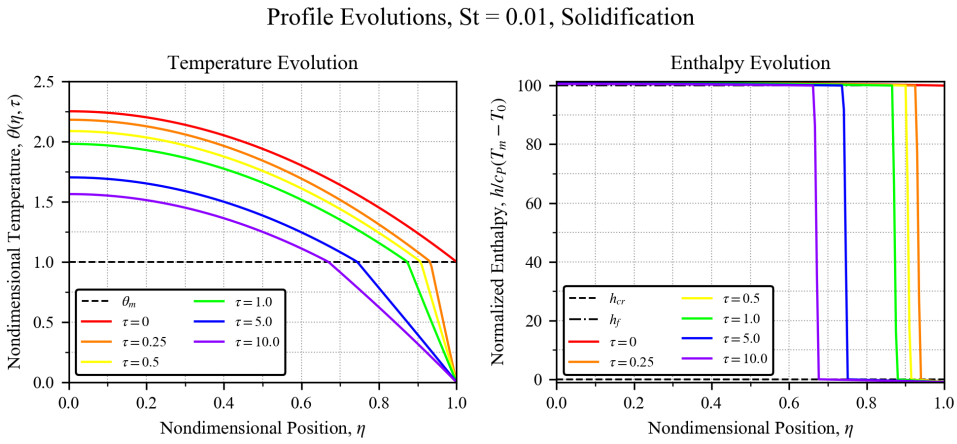

Figure 5.12: Numerically generated profiles for the CST solidification case at various time steps with $\dot{Q} = 5.0$, $St = 0.01$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Figure 5.13 shows the analytical model temperature profiles for solidification at $St = 0.1$, representing an order of magnitude increase in interface speed. We can see even with the quicker interface motion the slope change at the interface occurs for earlier time steps. Closer to steady-state, we observe a parabolic profile with a seamless slope between the liquid and solid regions. The change in temperature profile decreases significantly as the model approaches steady-state, as evidenced by the similarity in profiles between $\tau = 5.0$ and $\tau = 10.0$ as compared to $\tau = 0.5$ and $\tau = 1.0$.

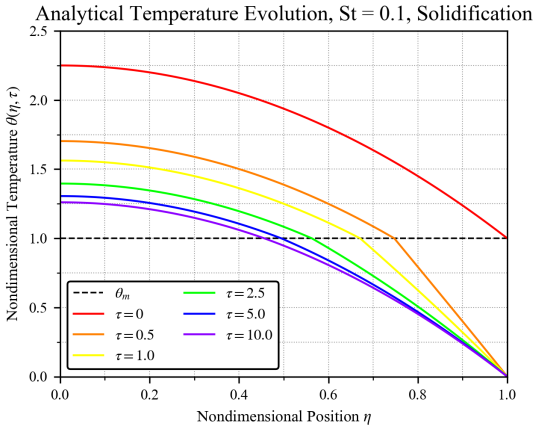

Figure 5.13: Analytically generated temperature profiles for the CST solidification case at various time steps with $\dot{Q} = 5.0$, $St = 0.1$.

Figure 5.14 tells a similar story. The agreement between the analytical and numerical model temperature profiles is still very strong, and there is a visible change in slope at the phase change interface that smooths over time. As with the $St = 0.01$ case, the enthalpy profiles show a step change from h_f to h_{cr} from the earliest time steps; no mushy zone has formed.

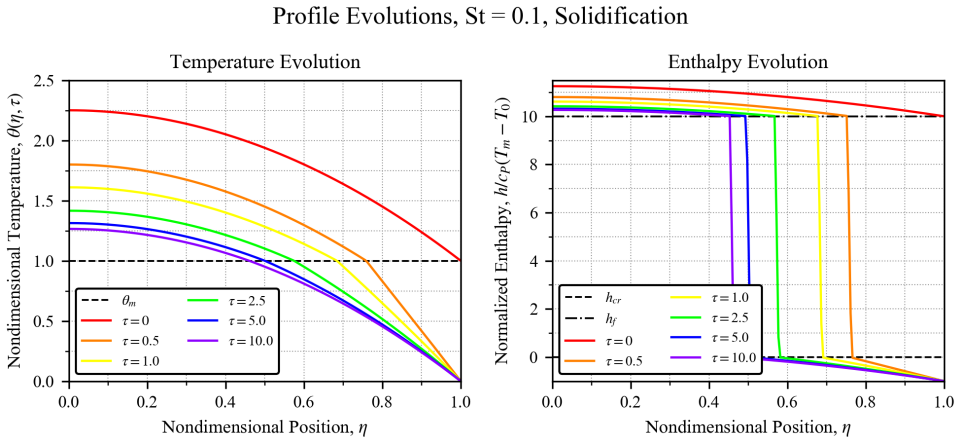

Figure 5.14: Numerically generated profiles for the CST solidification case at various time steps with $\dot{Q} = 5.0$, $St = 0.1$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Increasing the Stefan number to $St = 1.0$, Figure 5.15 presents the temperature profiles generated by the analytical model. This rate of interface motion again represents the upper limits of the quasi-steady interface assumption. It can be seen that for this case, though the early temperature profiles exhibit the slope change at the interface, it smooths out much more quickly than for the lower Stefan number cases. We also see at the final time steps the model has approached steady-state, as evidenced by the similarity in profiles between $\tau = 1.0$ and $\tau = 2.0$.

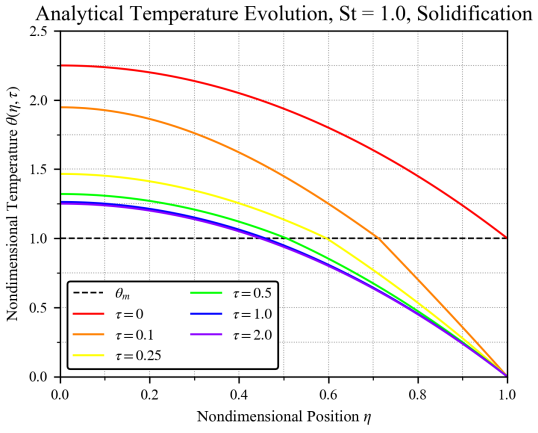

Figure 5.15: Analytically generated temperature profiles for the CST solidification case at various time steps with $\dot{Q} = 5.0$, $St = 1.0$.

The numerically generated temperature and enthalpy profiles for solidification at $St = 1.0$ are presented in Figure 5.16. Here, there is a more evident difference in profile shapes generated as compared to those presented in Figure 5.15. The slope change at the interface is much less pronounced in the numerical model, with only a slightly visible change for $\tau = 0.1$. The slower interface motion is also noticeable, as there is more change in profile between $\tau = 1.0$ and $\tau = 2.0$. As has been the case for the previous solidification cases, the enthalpy profiles do not exhibit any formation of a mushy zone.

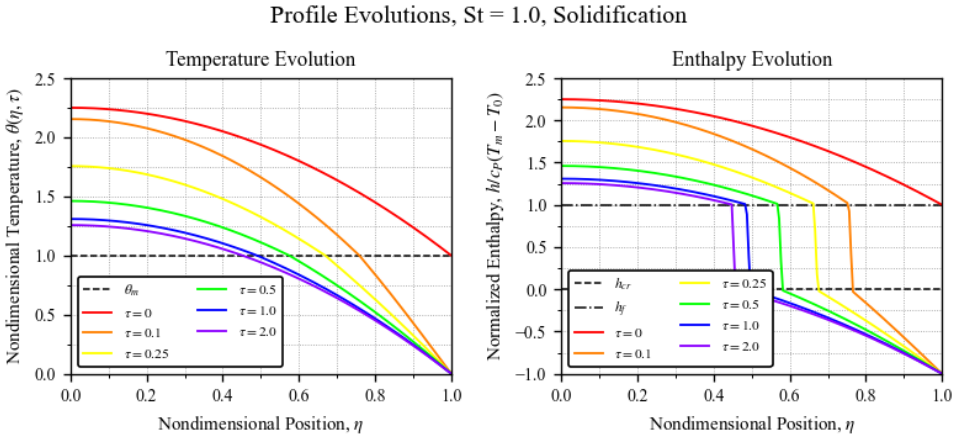

Figure 5.16: Numerically generated profiles for the CST solidification case at various time steps with $\dot{Q} = 5.0$, $St = 1.0$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Our final two sets of CST problem results are displayed in Figure 5.17, representing the analytical model solutions, and Figure 5.18, representing the numerical. At a Stefan number of $St = 10.0$, these represent extremely fast phase change motion, where quasi-steady assumptions are invalid. Focusing first on the analytical temperature profiles, we see even by the earliest time step of $\tau = 0.05$, the slope change between the liquid and solid regions has already been smoothed out, and the overall profile shape is parabolic.

Shifting our focus to the numerical temperature profiles in Figure 5.18, we see some noticeable differences compared to the analytical profiles. This matches with the lag in interface motion, noted in the discussion of Figure 5.10.

One difference of interest occurs with the $\tau = 0.05$ temperature profile: The temperature change near the centerline is almost nonexistent, significantly less than for the same time step in Figure 5.17. This is likely a consequence of numerical diffusion, and may have been exacerbated by the upwind discretization scheme used to solve the energy equation for the numerical model. Unsurprisingly, we see no mushy zone formation in the enthalpy profiles. Not only are all transitions from h_f to h_{cr} sharp, the gap between the two values is once again much smaller than the overall size of the profiles.

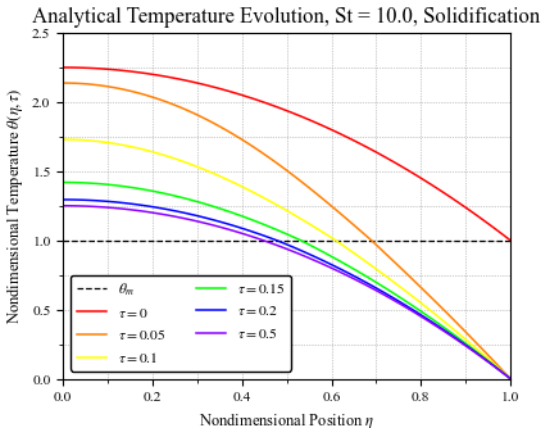

Figure 5.17: Analytically generated temperature profiles for the CST solidification case at various time steps with $\dot{Q} = 5.0$, $St = 10.0$.

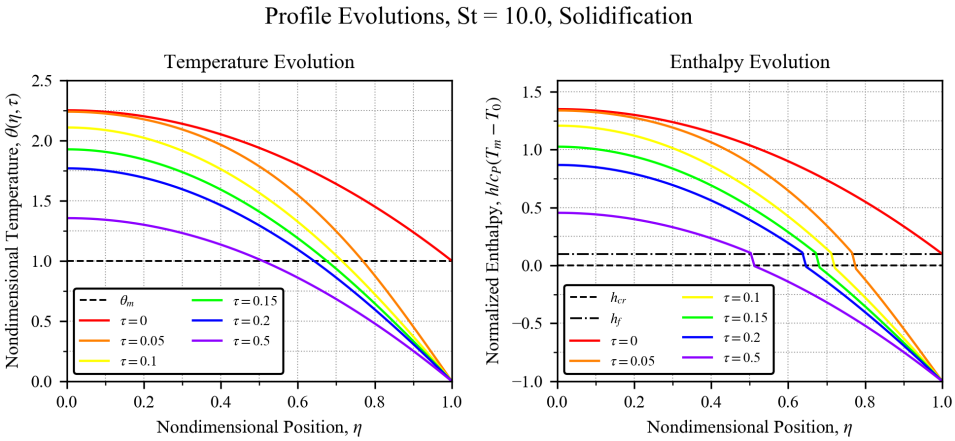

Figure 5.18: Numerically generated profiles for the CST solidification case at various time steps with $\dot{Q} = 5.0$, $St = 10.0$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Overall, the CST solution demonstrates good tracking of the interface relative to the numerical model that diminishes as the Stefan number approaches and passes a value of 1. However, as a result of neglecting the mushy zone, the melting cases display a nonphysical overheating phenomenon at lower time steps. The period of time for which this issue is prevalent increases as Stefan number decreases due to higher relative heat of fusion values. The overheating phenomenon does not appear in solidification cases, leading to better temperature profile agreements, particularly at low Stefan numbers. However, increases in the Stefan number for solidification cases lead to a larger disagreement in interface position as a function of time.

5.2 Constant Surface Heat Flux Results

We now turn to the CSHF case. Where the CST case could use $\dot{Q}$ and St to modulate the speed of phase change, the lack of a constant wall temperature makes the Stefan number a poor choice for control in the CSHF case. Instead, the CSHF case can use $\dot{Q}$ and Q''. As before, we keep our internal heat generation constant at $\dot{Q} = 5$, using Q'' to determine if the model is melting or solidifying and at what speed.

Referencing Equation 3.81 and applying $\dot{Q} = 5$, we get the relation:

$$Q'' \begin{cases} < 2.5 \rightarrow \text{Melting} \\ = 2.5 \rightarrow \text{Equilibrium} \\ > 2.5 \rightarrow \text{Solidification} \end{cases} \tag{5.2}$$

We use three values of heat flux representing slow, middling, and fast interface change for both the melting and solidification cases. For the melting scenarios, these correspond to flux values of $Q'' = 2.4$, $Q'' = 2.0$, and $Q'' = 1.5$, respectively; for the solidification scenarios, they correspond to $Q'' = 2.6$, $Q'' = 3.0$, and $Q'' = 4.0$.

Melting Results

As with the CST models, we start with the phase change interface evolution in time for the three melting cases. Figure 5.19 presents compares the interface curves for the analytical model, represented by the solid lines, and the numerical model, represented by the dotted lines with symbols. One of the notable differences between the CST and CSHF problems is visible here: the form of steady-state. Unlike the CST problem, the interface in the CSHF problem does not asymptotically approach a value. Instead, the interface steady-state values are dictated by the physical boundaries of our problem domain. The constant flux means the model will transition from fully solid to fully molten for melting cases, and vice-versa for solidification. Additionally, the temperature profiles do not reach a true steady-state value.

Compared to the CST results, the agreement between the analytical and numerical models is much weaker. This appears to be primarily driven by a slope difference at earlier time steps; at later time steps, curves concurrent to a particular heat flux become parallel.

Figure 5.20 presents temperature profiles at various time steps for the $Q'' = 2.4$ case. The dashed line at θ_m once again represents the nondimensional melting temperature. Note that the definition of nondimensional temperature differs between the CST and CSHF problems; where phase change occurred in the CST problem at $\theta = 1$, it occurs at $\theta = 0$ for the CSHF problem.

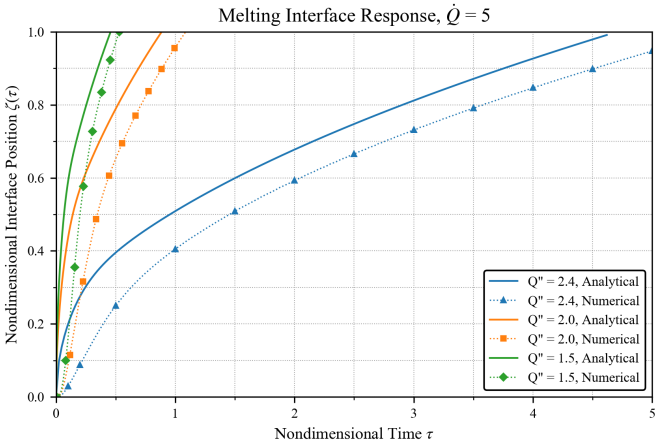

Figure 5.19: CSHF interface position as a function of time during melting at $\dot{Q} = 5.0$ for various values of Q''.

Throughout the entire range of time steps, the profile at the interface between the liquid and solid regions remains smooth, and the shape of the temperature profiles remains constant, with only a vertical translation reflecting the advancement of time. This is one of the key differences between the CST and CSHF problems: The temperature profiles in the CSHF problem do not become static as the transient portion of our solution decays. Another key difference is that the CSHF solution does not exhibit the overheating phenomenon observed at low time steps in the CST melting temperature profiles.

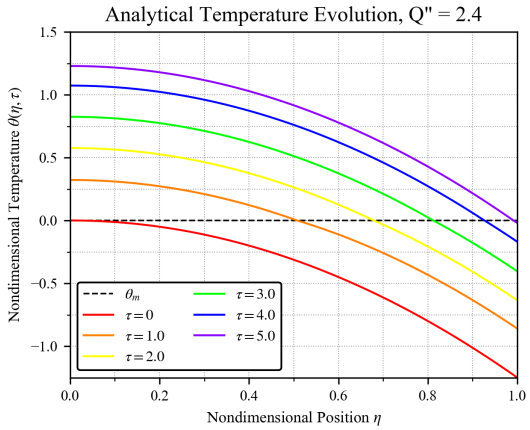

Figure 5.20: Analytically generated temperature profiles for the CSHF melting case at various time steps for $\dot{Q} = 5.0$, $Q'' = 2.4$.

The temperature and enthalpy profiles for the same flux case, $Q'' = 2.4$ using the numerical model are presented in Figure 5.21. Starting with the temperature profiles, we see that, similar to Figure 5.20, the shape of the profiles remains constant at each time step. However, the profile height changes more slowly

and more consistently. There is no noticeable mushy zone formation or effects by the first displayed time step, though this may have more to do with being a relatively large jump in time. It is possible the mushy zone has formed, expanded, and reduced in the time between $\tau = 0$ and $\tau = 1.0$, which would explain the difference in temperature profiles relative to the analytical solution.

Because the CSHF problem does not have a constant wall temperature, we cannot use Equation 5.1. We instead define our normalized enthalpy for the CSHF case as:

$$Normalized\ Enthalpy = \frac{h}{\Delta h_f} \tag{5.3}$$

Where we have set $h_{cr} = 0$ as a reference point. For all CSHF cases, this leads to a gap between h_f and h_{cr} of 1. Keeping these changes in mind and looking now at the enthalpy profiles of Figure 5.21, we see that by $\tau = 1.0$ the mushy zone profile is a step change from h_f to h_{cr}, matching our expectations from the temperature profiles.

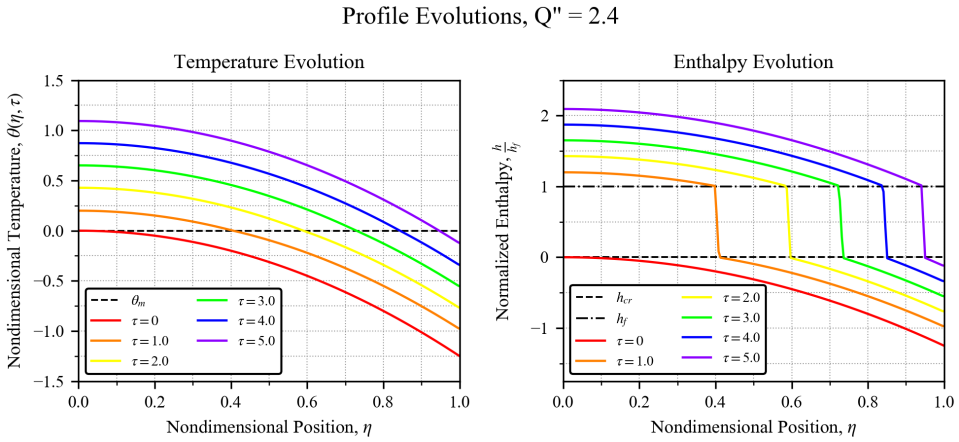

Figure 5.21: Numerically generated profiles for the CSHF melting case at various time steps with $\dot{Q} = 5.0$, $Q'' = 2.4$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

We now decrease our surface flux to $Q'' = 2.0$, increasing melting speed. Figure 5.22 presents the analytical solution temperature profiles for this case. Here we see some interesting departures from the previous case, most notably in profile shape. Where the profiles of Figure 5.20 maintained a consistent shape across all time steps and a seamless transition between liquid and solid regions, a visible change in profile near the interface is observed for all but the initial and final time steps of Figure 5.22. At $\tau = 0.05$, the solid region profile appears almost linear, tending towards a more seamless profile by $\tau = 0.1$, then beginning to concave upwards for later time steps. Of note, though the temperature profiles exhibit the fluctuations in shape, no overheating is observed. The irregularity of these profiles places some doubt on the validity of our weakly time-dependent assumption.

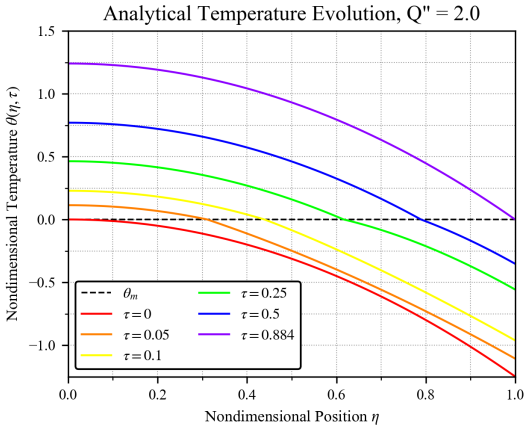

Figure 5.22: Analytically generated temperature profiles for the CSHF melting case at various time steps for $\dot{Q} = 5.0$, $Q'' = 2.0$.

Turning to the numerical model solutions for $Q'' = 2.0$, Figure 5.23 presents the temperature and enthalpy profiles for the case. A note here: The last time step used for the numerical model differs from the last used for the analytical model. This is because $\tau = 0.884$ represents the last point in time for the analytical model in which both phases exist; above this point, some values in the solution become complex, leading to the solution code exiting. This issue does not exist for the numerical model, which is able to handle single-phase analysis. The final time step was thus extended to $\tau = 1.0$. The numerical solution's slower interface motion results in the profile at this time step still containing two phases.

Aside from the slower motion, the numerical temperature profiles exhibit different behavior from the analytical profiles, particularly at low time steps. Here, we do see mushy zone behavior and an associated isotherm at θ_m for time steps up through $\tau = 0.25$. Unlike the mushy zone formation in the CST case, the temperature profiles do not become temporarily static here. This is understandable, as the flux boundary condition does not create a temporary equilibrium in energy balance while the fully liquid region forms. After the mushy zone has contracted to negligible size, we observe a slope change between the liquid and solid regions at the interface, in contrast to the CST melting cases, where the transition from liquid to solid becomes more smooth in time.

Shifting attention to the enthalpy plot, we see the mushy zone profiles for earlier time steps take on a different shape compared to the mushy zone development profiles of the CST melting case. As time progresses and the mushy zone region becomes negligible in size, we see the characteristic plateau in enthalpy from h_f to h_{cr} form. The difference in mushy zone profile shape may help to explain the lack of overheating and irregularity in solid region profile shape at low time steps that are found in Figure 5.22.

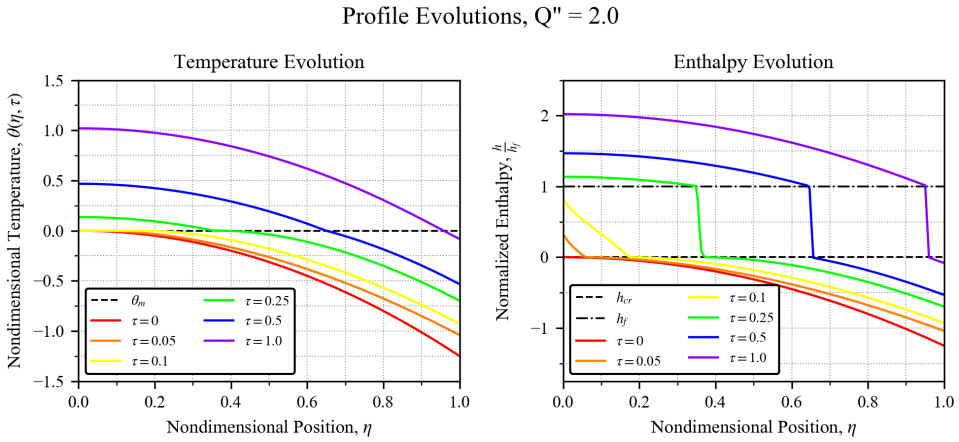

Figure 5.23: Numerically generated profiles for the CSHF melting case at various time steps with $\dot{Q} = 5.0$, $Q'' = 2.0$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Figure 5.24 presents the temperature profiles generated from the analytical model for melting at $Q'' = 1.5$, the fastest of the melting cases. At $\tau = 0.05$, the solid region is visibly concave up, a significant departure from the initial temperature profile. The deflection in shape levels out near $\tau = 0.1$, where the overall temperature appears smooth and parabolic, and reverses for late time steps, where the slope change between liquid and solid regions at the interface once again become visible. As with the final time step of Figure 5.22, the profile of $\tau = 0.456$ represents the last calculated time step available before the model has become entirely molten.

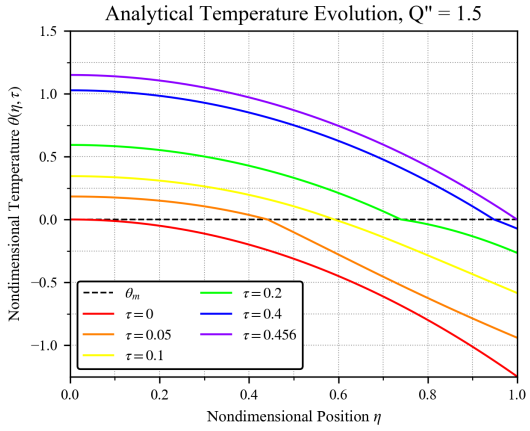

Figure 5.24: Analytically generated temperature profiles for the CSHF melting case at various time steps for $\dot{Q} = 5.0$, $Q'' = 1.5$.

The temperature profiles generated by the numerical model for this case, shown in the lefthand plot of Figure 5.25, differ significantly from those generated by the analytical model. Most notably, the early time steps do not exhibit deflection towards a concave up profile but instead contain traces of the mushy

zone in the form of an extended isotherm at θ_m. The disconnect in melting speed between the analytical and numerical temperature models makes it difficult to compare concurrent time steps, so it is unclear if the development and contraction of the mushy zone in the numerical model coincide with the change in solid region deflection in the analytical model.

The righthand side of Figure 5.25 shows the enthalpy profiles for melting at $Q'' = 1.5$. As expected from the isotherms in the temperature profiles, the mushy zone shows developments at earlier time steps before tending towards its characteristic plateau. At the beginning of the mushy zone development, we see a negative concave up curve similar to the mushy zone developments in Figure 5.23, but somewhere between $\tau = 0.1$ and $\tau = 0.2$, this profile changes and begins to look similar to the mushy zone profiles generated at low Stefan number CST cases.

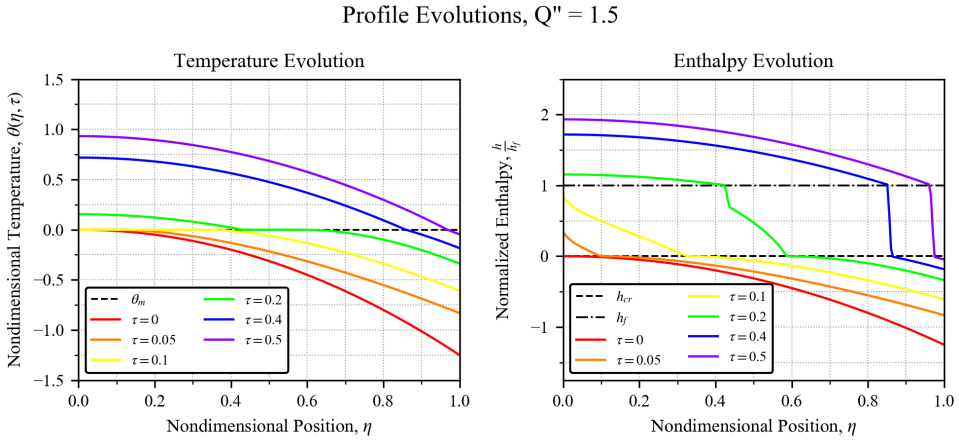

Figure 5.25: Numerically generated profiles for the CSHF melting case at various time steps with $\dot{Q} = 5.0$, $Q'' = 1.5$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Solidification Results

Turning now to results of the solidification cases, Figure 5.26 compares the interface evolution in time of the analytical model against the numerical model for the three heat flux values used. As was the case with the melting model, steady-state is not achieved by an asymptotic approach to a particular value, but rather by entirely reaching one phase.

Similar to the melting cases, the agreement between the analytical and numerical models for the solidification cases is weaker than it was for the CST problem. Unlike the CST problem, the disagreement is most prevalent with slower interface motion, where the surface heat flux is closer to balanced with the internal heat generation. The disagreement also increases as time increases and more of the domain becomes solid.

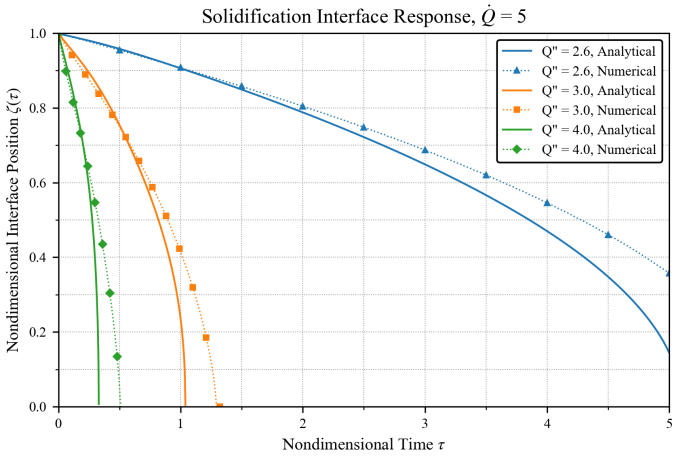

Figure 5.26: CSHF interface position as a function of time during solidification at $\dot{Q} = 5.0$ for various values of Q''.

Temperature profiles from the analytical solution at various time steps for solidification at $Q'' = 2.6$, representing the slowest of the solidification cases, are shown in Figure 5.27. Similar to the $Q'' = 2.4$ melting case (Figure 5.20), we see uniformly-shaped curves throughout the entire time series, except at $\tau = 5.0$. At this time step, we see a significant change in profile between the liquid and solid regions, as well as a larger change in interface position from the previous time step. Note that this region of time also corresponds with the greater disagreement in interface position between the numerical and analytical models. Similar to solidification CST cases, we do not see unusual behavior at early time steps that might suggest a solidification mushy zone.

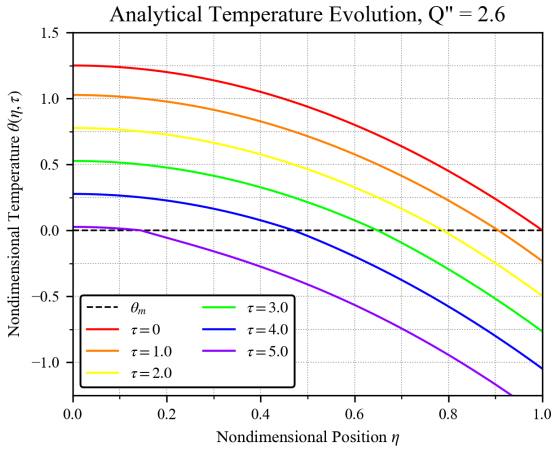

Figure 5.27: Analytically generated temperature profiles for the CSHF solidification case at various time steps for $\dot{Q} = 5.0$, $Q'' = 2.6$.

The results of the numerical model for this case are presented in Figure 5.28. This time, we do not see a slope change at the interface for the last time step, and all temperature profiles maintain the same general shape. This likely has to do with how much farther from the centerline the interface is compared to the analytical model at the same time step. Referencing the enthalpy plots, we can confirm no solidification mushy zone effects are visible.

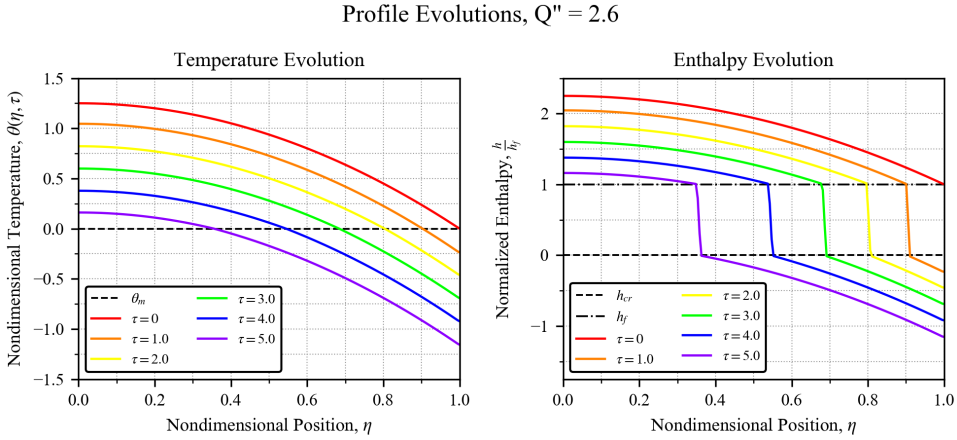

Figure 5.28: Numerically generated profiles for the CSHF solidification case at various time steps with $\dot{Q} = 5.0$, $Q'' = 2.6$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Figure 5.29 presents the temperature profiles generated by the analytical solution for solidification at $Q'' = 3.0$. We see that even at earlier time steps and higher interface values the shape of the temperature profiles has begun to change, with a small but noticeable change in slope between liquid and solid regions at the interface. At $\tau = 1.0$, where the interface is near the centerline, we once again see a much more significant change in profile shape, with the solid region appearing nearly linear.

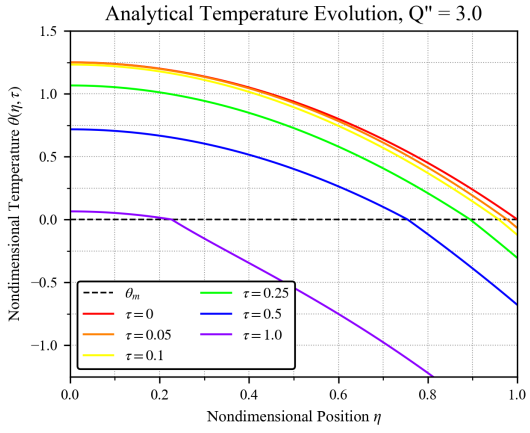

Figure 5.29: Analytically generated temperature profiles for the CSHF solidification case at various time steps for $\dot{Q} = 5.0$, $Q'' = 3.0$.

The equivalent results to this case using the numerical model are presented in Figure 5.30. We see good agreement in the temperature profiles at early time steps, with both shape and interface position nearly matching through $\tau = 0.5$. However, as with the $Q'' = 2.6$ case, the late time step, low interface value profiles differ both in shape and interface speed. While some of this change may be attributed to numerical diffusion exacerbated by the upwind energy scheme, the difference between the two models is too great for diffusion to be the sole contributor.

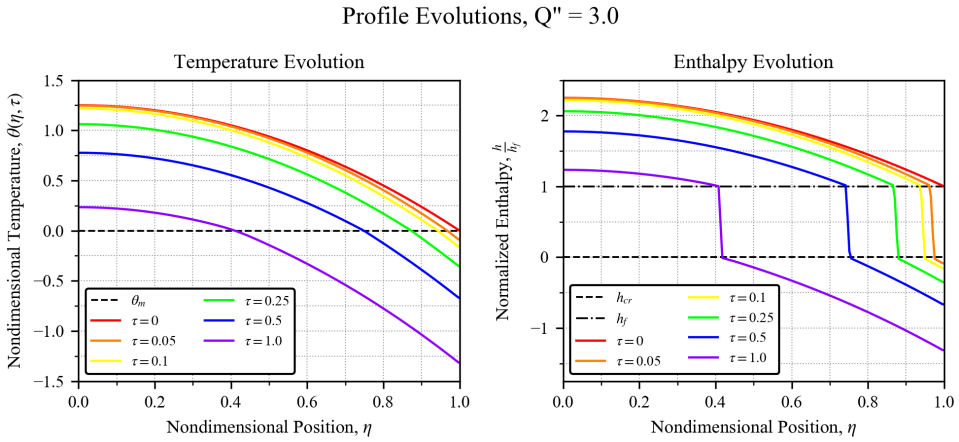

Figure 5.30: Numerically generated profiles for the CSHF solidification case at various time steps with $\dot{Q} = 5.0$, $Q'' = 3.0$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

We now turn to the $Q'' = 4.0$ case, representing the fastest solidification CSHF case. Figure 5.31 shows the temperature profiles for the analytical model. By $\tau = 0.05$, there is already a noticeable slope change between regions at the interface, and the solid region temperature profile as almost linear. Interestingly, the temperature near the centerline decreases considerably more slowly than towards the wall; this is due to the relatively high level of surface flux relative to the thermal conductivity.

Figure 5.32 presents temperature and enthalpy profiles generated by the numerical model for the solidification at $Q'' = 4.0$ case. Note that the presented series of time steps for this figure is different from Figure 5.31. Though this prevents complete comparison of the two models in time, it allows us to observe the behavior of the numerical model at late stages of solidification.

The early time step temperature profiles exhibit similar behavior to the analytically generated profiles, with temperature change being dominant near the wall and delaying towards the centerline. This is consistent with the similar interface positions at early time steps shown in Figure 5.26. There is some slope change between regions at the interface at early time steps, though the solid region profile does not have as linear an appearance. Beyond $\tau = 0.2$, the interface positions and subsequent temperature profiles between the two models begins to diverge. At $\tau = 0.4$, the slope change has become more significant,

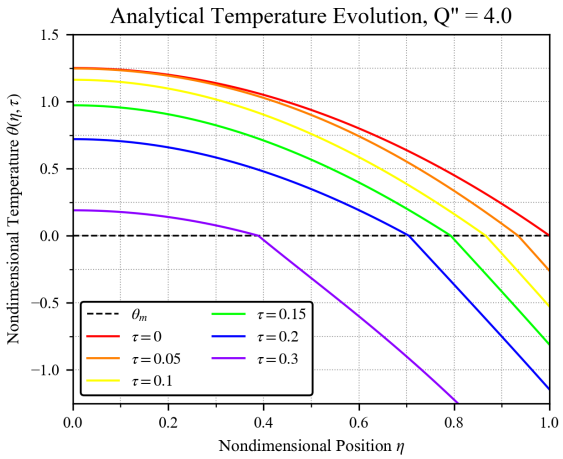

Figure 5.31: Analytically generated temperature profiles for the CSHF solidification case at various time steps for $\dot{Q} = 5.0$, $Q'' = 4.0$.

and we see a flattening out of the liquid region profile; by $\tau = 0.5$, this profile is nearly uniform near θ_m, and the solid region profile is no longer parabolic. Looking at the $\tau = 0.5$ curve in the enthalpy plot, the mushy zone profile is still a step jump from h_f to h_{cr}, suggesting that the solid region has not become mushy.

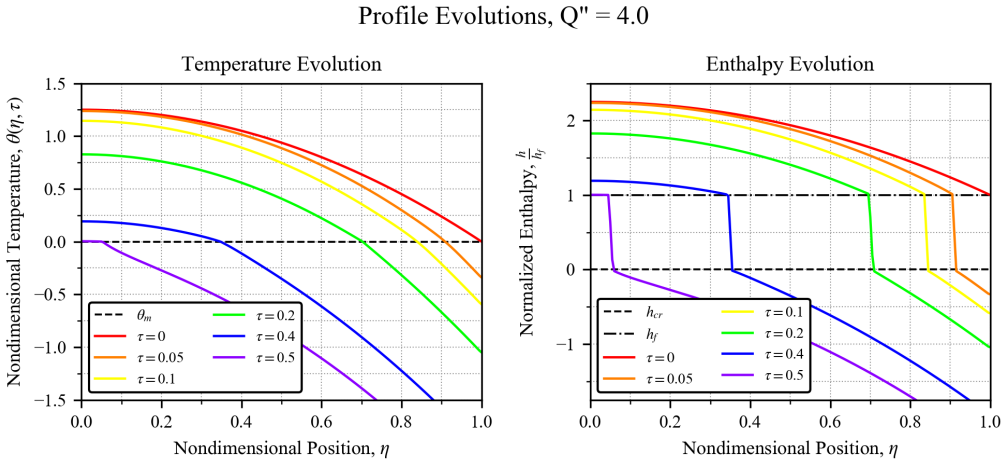

Figure 5.32: Numerically generated profiles for the CSHF solidification case at various time steps with $\dot{Q} = 5.0$, $Q'' = 4.0$. Temperature profiles are on the left side, while enthalpy profiles are on the right side.

Overall, we see poor agreement between the solution to the CSHF problem derived in Chapter 3 and the enthalpy method implementation of the equivalent. The models disagree most when the interface is near the centerline of the domain. Though omitting the mushy zone in our analytical formulation does not result in the overheating phenomenon we saw for the CST solutions, we do see impact in the form of deflections in the temperature profiles for higher flux melting cases. The solidification cases once more

do not exhibit any mushy zone effects, though the models still disagree at late time steps. We conclude these errors likely originate from an inaccurate theoretical model in that our separation of θ into θ_{ss} and θ_{tr} is not valid for the CSHF problem.

Chapter 6: Summary and Conclusions

This thesis explores a scenario of solid-liquid phase change driven by internal heat generation in cylindrical coordinates, based on analysis of the classical Stefan problem. The scenario is representative of states of meltdown in nuclear fuel rods. Though the problem of phase change as a whole is well researched, phase change driven by internal heat generation, particularly in finite geometries, is less well-explored. Few closed-form solutions exist to this class of problems and often are limited over-idealized cases with restrictive assumptions.

In Chapter 3, we derive solutions for transient melting and solidification with internal heat generation for a cylindrical geometry in one dimension assuming constant thermal properties and a sharp interface. Both cases of constant surface temperature and constant surface heat flux are considered. Nondimensional parameters are created to simplify the governing equations and generalize the solution. Assuming superposition, the governing equations are separated into transient and steady-state components. The steady-state components are solved using standard integration techniques, and the transient through separation of variables. The solutions are then used to generate a first-order ordinary differential equation describing the motion of the phase change front in time. The solution is implemented in Python in order to generate interface and temperature profiles. This process is repeated for both forms of boundary condition.

Verification through numerical means is explored in Chapter 4. Three models rooted in creating finite difference approximations of the governing equations are proposed, discussed, and ultimately rejected. The first of these models is a fixed grid, fixed time model that tracks interface position by interpolating between nodes; it was rejected due to stability issues at high Stefan numbers. The second proposed method, a variable grid, fixed time method, uses separate domains for the liquid and solid regions that expand and contract such that their interior boundaries match the interface position. Its inconsistent accuracy prevented the model from being used. The final finite difference model is a fixed grid, variable time model that solves for time required to the interface to move from one spatial node to the next. While it showed the most promise of the three models, it was not used due to poor extensibility and inability to model steady-state.

We choose to use an alternative style of numerical model based on finite volume approximation known as the enthalpy method. This model creates approximate solutions to the weak form of the energy equation. Its implicit tracking of liquid fraction over an explicit interface allows for analysis before phase change begins and after steady-state is achieved, as well as the development of a mushy zone, a region of material in a state of phase change between fully solid and fully liquid. This model is chosen because

it tests the validity of our assumption of a sharp interface, and because its implementation is available through the commercial software Ansys Fluent. The model uses a second-order upwind in space, first-order implicit in time discretization of the energy equation to solve dimensionally equivalent cases to those derived in Chapter 3.

The veracity of our solutions is tested across several cases in Chapter 5. For the cases of constant surface temperature, phase change direction is dictated by initial profile, and speed is controlled by modulating the Stefan number. From slowest change to fastest, we test the models at $\dot{Q} = 5.0$ and $St = 0.01$, $St = 0.1$, $St = 1.0$, and $St = 10.0$, for both melting and solidification. We find that agreement between the two models, particularly of interface position as a function of time, is strong at low Stefan numbers and gets weaker above $St = 1.0$. For these faster cases, the analytical model predicts faster interface motion. The melting cases exhibit nonphysical overheating at lower Stefan numbers, a consequence of the sharp interface assumption.

For the case of constant surface heat flux, phase change speed is controlled be the level of outward surface heat flux relative to the rate of internal heat generation. The heat generation is kept constant at $\dot{Q} = 5.0$. For melting, we use values of $Q'' = 2.4$, $Q'' = 2.0$, and $Q'' = 1.5$, with phase change speed increasing as flux decreases. For solidification, we use values of $Q'' = 2.6$, $Q'' = 3.0$, and $Q'' = 4.0$, with phase change speed increasing with flux. These cases show much weaker agreement than the constant surface temperature cases, with disagreement increasing when the phase change front moves quickly or when the interface is near the centerline. The disagreement is believed to be a result of using superposition.

No body of research is truly complete, and this body is no exception. Throughout the development of this thesis, several areas for future improvement and discovery were observed. In both the CST and CSHF cases, our analyses were limited to a single value of internal heat generation in order to manage the scope of work. A factorial analysis modulating both internal heat generation and values of the external boundary condition would provide broader insight into the behavior of the system, including what influences mushy zone formation. Another limitation of our model arises from using purely nondimensional parameters. Scenarios involving material properties consistent with modern fuel rods would lend a better understanding of the range where our solutions make physical sense.

The two boundary conditions we considered are fairly common in use for heat transfer. A third boundary condition, known as the mixed boundary condition, exists, which essentially is a linear combination of the two previous boundary conditions. It is representative of convective heat transfer at the surface. Solutions involving this boundary condition would increase the range of physical applications our work pertains to. If a close-formed solution exists, however, it is likely to suffer from the same consequences of using superposition as the CSHF case did. This problem would also likely require its own set of nondi-

mensional parameters, though the reformulation may help to define a set this allows consistent definition of temperature and heat generation for the previous two problems. While we note characteristics of our analytical solutions that are nonphysical or inaccurate, we do not formally establish regions of validity for each analytical solution. An improvement to the verification process would be to define these regions. Ultimately, this would require the additional case testing discussed earlier.

The validity of the superposition principle in separating our temperature profiles is questionable, particularly in the case of surface flux boundary conditions, due to nonhomogeneity of the separated boundary conditions. This is believed to be a major factor in the disagreement between analytical and numerical models for the CSHF cases. It may be possible to derive solutions that do not lead to this issue. Additionally, a reformulation of the governing equations such that the dependent variable is enthalpy instead of temperature could feasibly allow for mushy zone development and preclude the need for an explicit energy balance to define interface. This, however, would involve derivation from the energy equation and is not likely to yield closed form solutions.

Regarding the numerical model, the primary improvement necessary in future models is implementation of noncommercial code. The Ansys Fluent framework was convenient and necessary within the timeframe of the thesis work as it generated trustworthy results without excess testing or code refactoring. However, it is limited to low-order discretization in Cartesian coordinate systems. An in-house code could remedy both of these limitations while generating solutions at a faster rate due to reduced computational overhead. Such improvements would allow for more extensive case testing in a feasible timeframe. It would also allow for incorporation of the same nondimensional paradigm used in our analytical solutions, whereas the Fluent framework required identifying dimensional values that yielded equivalent nondimenionsal values.

If Fluent were to be used again, employment of its expanded capabilities over analytical solutions would be beneficial. This includes multidimensional domains, thermophysical properties that vary with energy or phase, and the existence of fluid, convective, and radiative effects. Modulation of dimensional values while keeping nondimensional values constant would also be able to determine the veracity of the nondimensional systems we used, and implementing adaptive meshing around the region of phase change would be increase the resolution of phenomena in the mushy zone. In either the case of continued Fluent use or in-house code, a formal convergence study should be implemented to add confidence in the final results.

References

[1] G. Lamé and B. Clapeyron. Memoire sur la solidification par refroidissement d'un globe liquide. *Annales Chimie Physique*, 47:250–256, 1831.

[2] J. Stefan. Uber die theorie der eisbildung, insbesondere uber die eisbildung im polarmeere. In *Sitzungsberichte der k.k. Akademie der Wissenschaften in Wien*, pages 965–983, 1889.

[3] I. I. Danilyuk. On the Stefan problem. *Russian Mathematical Surveys*, 40(157), 1985.

[4] H. S. Carslaw and J. C. Jaeger. *Conduction of Heat in Solids.* Oxford University Press, Amen House, London, 1959.

[5] J. Crank. *Free and Moving Boundary Problems.* Clarendon Press, Oxford, 1984.

[6] H. Hu and S. Argyropoulus. Mathematical modelling of solidfication and melting: a review. *Modelling Simulation Material Science Engineering*, 4(371), 1996.

[7] M. Ulvrova, S. Labrosse, N. Coltice, P. Raback, and P. Tackley. Numerical modelling of convection interacting with a melting and solidification front: Application to the thermal evolution of the basal magma ocean. *Physics of the Earth and Planetary Interiors*, 206(51), 2012.

[8] F. I. Dragomirescu, K. Eisenschmidt, C. Rohde, and B. Weigand. Perturbation solutions for the finite radially symmetric Stefan problem. *International Journal of Thermal Sciences*, 104(386):386–395, 2016.

[9] S. Singh and S. Kumar. Freezing of biological tissues during cryosurgery using hyperbolic heat conduction model. *Mathematical Modelling and Analysis*, 20(4):443–456, 2015.

[10] A. V. Fedorov and A. V. Shul'gin. Modeling of combustion of a magnesium particle (Stefan problem). *Combustion, Explosion, and Shock Waves*, 45(6):651–656, 2009.

[11] J. Lorenzo-Trueba and V. R. Voller. Analytical and numerical solution of a generalized stefan problem exhibiting two moving boundaries with application to ocean delta formation. *Journal of Mathematical Analysis and Applications*, 366:538–549, 2010.

[12] C. R. Regis, R. M. Cotta, and J. Su. Improved lumped analysis of transient heat conduction in a nuclear fuel rod. *International Communications in Heat and Mass Transfer*, 27(3), 2000.

[13] V. R. Voller. Fast implicit finite-difference method for the analysis of phase change problems. *An International Journal of Computation and Methodology*, 17, 1989.

[14] V. R. Voller and C. R. Swaminathan. General source-based method for solidification phase change. *An International Journal of Computation and Methodology*, 19(2), 1991.

[15] J. Caldwell and C-C. Chan. Spherical solidification by the enthalpy method and the heat balance integral method. *Applied Mathematics Modelling*, 24:45–53, 2000.

[16] K. P. Furlong and D. S. Chapman. Heat flow, heat generation, and the thermal state of the lithosphere. *Annual Review of Earth and Planetary Sciences*, 41(1):385–410, 2013.

[17] S. A. Weinstein. Catastrophic overturn of the earth's mantle driven by multiple phase changes and internal heat generation. *Geophysical Research Letters*, 20(2):101–104, 1993.

[18] H. Jiang, J. Zhao, C. Wang, and X. Liu. Effect of electric current pulses on solidification of immiscible alloys. *Materials Letters*, 132:66–69, 2014.

[19] M. C. Kumar, A. J. Sudha, D. Ponraju, and S. Athmalingam. Effect of internal heat generation on solidification of molten fuel droplet during its interaction with coolant in a nuclear reactor. *IOP Conference Series: Materials Science and Engineering*, 912:042006, 2020.

[20] J. Tang, M. Huang, Y. Zhao, S. Maqsood, and X. Ouyang. Numerical investigations on the melting process of the nuclear fuel rod in RIAs and LOCAs. *International Journal of Heat and Mass Transfer*, 124:990–1002, 2018.

[21] C. An, F. C. Moreira, and J. Su. Thermal analysis of the melting process in a nuclear fuel rod. *Applied Thermal Engineering*, 68(1–2):133–143, 2014.

[22] C. An and J. Su. Lumped parameter model for one-dimensional melting in a slab with volumetric heat generation. *Applied Thermal Engineering*, 60(1–2):387–396, 2013.

[23] A. Shrivastava, B. Williams, A. Siahpush an B. Savage, and J. Crepeau. Numerical and experimental investigation of melting with internal heat generation within cylindrical enclosures. *Applied Thermal Engineering*, 67(1–2):587–596, 2014.

[24] Z-T. Yu, L-W. Fan, Y-C. Hu, and K-F. Cen. Perturbation solution to heat conduction in melting or solidification with heat generation. *Heat Mass Transfer*, 46(479):479–483, 2010.

[25] D. Slota. Direct and inverse one-phase Stefan problem solved by the variational iteration method. *Computers and Mathematics with Applications*, 54(2007):1139–1146, 2006.

[26] L. M. Jiji and S. Gaye. Analysis of solidification and melting of pcm with energy generation. *Applied Thermal Engineering*, 26(5–6):568–575, 2005.

[27] J. Crepeau and A. Siahpush. Approximate solutions to the Stefan problem with internal heat generation. *Heat Mass Transfer*, 44(787):787–794, 2008.

[28] M. Bechiri and K. Mansouri. Analytical study of heat generation effects on melting and solidification of nano-enhanced PCM inside a horizontal cylindrical enclosure. *Applied Thermal Engineering*, 104:779–790, 2016.

[29] D. McCord, J. Crepeau, A. Siahpush, and J. A. F. Brogin. Analytical solutions to the Stefan problem with internal heat generation. *Applied Thermal Engineering*, 103(2016):443–451, 2016.

[30] L. Barannyk, J. Crepeau, P. Paulus, and A. Siapush. Fourier-bessel series model for the Stefan problem with internal heat generation in cylindrical coordinates. In *Proceedings of the 2018 26th International Conference on Nuclear Engineering*, London, England, 2018.

[31] L. Barannyk, S. Williams, O. Ogidan, J. Crepeau, and A. Sakhnov. On the Stefan problem with internal heat generation and prescribed heat flux conditions at the boundary. In *Proceedings of the ASME 2019 Heat Transfer Summer Conference*, Bellevue, Washington, 2019.

[32] T. Bergman, A. Lavine, F. Incropera, and D. Dewitt. *Fundamentals of Heat and Mass Transfer*. John Wiley & Sons Inc., Hoboken, New Jersey, 2011.

[33] D. Poulikakos. *Conduction Heat Transfer*. Prentice Hall, Englewood Cliffs, NJ, 1994.

[34] M. Heath. *Scientific Computing: An Introductory Survey*. McGraw-Hill, New York, New York, 2002.

[35] J. Crank. Two methods for the numerical solution of moving-boundary problems in diffusion and heat flow. *Quarterly Journal of Mechanics and Applied Mathematics*, 10:220–231, 1957.

[36] W. D. Murray and F. Landis. Numerical and machine solutions of transient heat-conduction problems involving melting or freezing. *Journal of Heat Transfer*, 81:106–112, 1959.

[37] L. C. Tien and S. W. Churchill. Freezing front motion and heat transfer outside an infinite isothermal cylinder. *A.I.Ch.E Journal*, 11(5):790–793, 1965.

[38] A. A. Smarskii and P. N. Vabishchevich. *Computational Heat Transfer, Volume 1*. Wiley, Hoboken, 1995.

[39] J. C. Crepeau, A. Y. Sakhnov, and V. S. Naumkin. Stefan problem with internal heat generation: Comparison of numerical modeling and analytical solution. *Journal of Physics: Conference Series*, 1369, 2019.

[40] R. H. Tien and G. E. Geiger. A heat-transfer analysis of the solidification of a binary eutectic system. *Journal of Heat Transfer*, 89(3):230–233, 1967.

[41] D. R. Atthey. A finite difference scheme for melting problems. *IMA Journal of Applied Mathematics*, 13(3):353–366, 1974.

[42] V. R. Voller and C. Prakash. A fixed grid numerical modelling methodology for convection-diffusion mushy region phase-change problems. *International Journal of Heat Mass Transfer*, 30(8):1709–1719, 1987.

[43] P. Welahettige and K. Vaagsaether. Comparison of OpenFOAM and ANSYS Fluent. In *Proceedings of the 9th EUROSIM Congress on Modelling and Simulation*, Porsgrunn, Norway, 2016.

[44] R. Pletcher, J. Tannehill, and D. Anderson. *Computational Fluid Mechanics and Heat Transfer*. CRC Press, Boca Raton, Florida, 2012.

[45] C. Yao, B. T. F. Chung, and G. X. Wang. Mushy zone equilibrium solidification of a semitransparent layer subject to radiative and convective cooling. *International Journal of Heat and Mass Transfer*, 45(11), 2002.

Appendix A: CST Analytical Solution Python Script

Listing A.1: cstcodedef script

```
# ============================================================
# Library Imports
# ============================================================
import numpy as np ## provides matrix support
from scipy.special import j0, j1, y0, y1, jn_zeros ## Bessel fcn family
from scipy.optimize import bisect ## basic root solver for transcendentals
from scipy.integrate import quad, solve_ivp ## quad for integration,
                                            ## solve_ivp for ODE solution


eps = 1e-4

# General order: eta, tau, zeta, Q, St, lmda, lmda_t, nterms, melt
# ============================================================
# Liquid Region Definitions
# ============================================================
def lambdas_liquid(zeta, nterms):
    ## Generates the eigenvalues for the liquid region.
    z_0_n = jn_zeros(0, nterms)
    lmda = z_0_n/zeta
    return lmda


def Phi_liquid(eta, Q, melt):
    ## Initial temperature profile in the liquid region.
    if melt == True:
        return 1
    else:
        return (Q/4)*(1 - eta**2) + 1
```

```
def theta_liquid_ss(eta, zeta, Q):
    ## Steady-state liquid temperature profile.
    return (Q/4)*(zeta**2 - eta**2) + 1

def A_n(zeta, Q, lmda, melt):
    ## Fourier coefficient for the liquid region. Index info is
    # communicated by lmda.
    def A_n_num_int(eta, zeta, Q, lmda, melt):
        # Numerator integrand
        term1 = Phi_liquid(eta, Q, melt) - theta_liquid_ss(eta, zeta, Q)
        term2 = j0(lmda*eta)*eta
        return term1*term2
    def A_n_den_int(eta, lmda):
        # Denominator integrand
        return j0(lmda*eta)**2*eta

    # Integrate numerator and denominator separately then combine.
    A_n_num = quad(A_n_num_int, eps, zeta-eps,
                   args=(zeta, Q, lmda, melt), limit=100)[0]
    A_n_den = quad(A_n_den_int, eps, zeta-eps,
                   args=(lmda), limit=100)[0]
    A_n = A_n_num/A_n_den
    return A_n

def theta_liquid(tau, zeta, Q, nterms, melt):
    ## Liquid temperature profile for a given tau and zeta.
    # Initialize output arrays.
    eta = np.linspace(eps, zeta-eps, 51)
    theta = np.zeros_like(eta)

    # Identify eigenvalues.
```

```
    lmda = lambdas_liquid(zeta, nterms)

    # Sum terms
    for n in range(nterms):
        A = A_n(zeta, Q, lmda[n], melt)
        theta += A*np.exp(-lmda[n]**2*tau)*j0(lmda[n]*eta)
    # Add steady-state impact
    theta += theta_liquid_ss(eta, zeta, Q)
    return eta, theta

# ============================================================
# Solid Region Definitions
# ============================================================
def lambdas_solid(zeta, nterms):
    ## Generates the eigenvalues for the solid region.
    def B_n_transcendental(lmda):
        # Transcendental form of the eigenfunction.
        return j0(lmda*zeta)*y0(lmda) - y0(lmda*zeta)*j0(lmda)

    def transcendental_roots(nterms):
        ## Find n roots of the quasi-periodic transcendental equation.
        # Initialize roots array.
        roots = np.zeros(nterms)
        margin = 1e-8
        # Manually find first root.
        left, right = margin, 4/(1 - zeta)
        roots[0] = bisect(B_n_transcendental, left, right, maxiter=2000)
        for n in range(nterms):
            left = roots[n-1] + margin
            right = roots[n-1] + 3.4/(1 - zeta)
            roots[n] = bisect(B_n_transcendental, left, right, maxiter=2000)
        return roots
```

```
    lmda_t = transcendental_roots(nterms)
    return lmda_t

def Phi_solid(eta, Q, melt):
    ## Generate initial solid temperature profile.
    if melt == True:
        return 1 - eta**2
    else:
        return 1

def theta_solid_ss(eta, zeta, Q):
    ## Generate steady-state solid temperature profile.
    term1 = (Q/4)*(1 - eta**2)
    term2 = (Q/4)*(zeta**2 - 1) + 1
    term3 = np.log(eta)/np.log(zeta)
    return term1 + term2*term3

def fn_til(eta, lmda_t):
    ## Generates the eigenfunction for the solid region.
    return y0(lmda_t*eta) - (y0(lmda_t)/j0(lmda_t))*j0(lmda_t*eta)

def fn_til_prime(eta, lmda_t):
    ## Derivative of eigenfunction for the interface equation.
    f_p = -lmda_t*(y1(lmda_t*eta) - (y0(lmda_t)/j0(lmda_t))*j1(lmda_t*eta))
    return f_p

def B_n(zeta, Q, lmda_t, melt):
    ## Fourier coefficient for the solid region.
    def B_n_num_int(eta, zeta, Q, lmda_t, melt):
        # Numerator integrand
        term1 = Phi_solid(eta, Q, melt) - theta_solid_ss(eta, zeta, Q)
```

```
        term2 = fn_til(eta, lmda_t)*eta
        return term1*term2
    def B_n_den_int(eta, lmda_t):
        # Denominator integrand
        return fn_til(eta, lmda_t)**2*eta

    # Integrate numerator and denominator separately then combine.
    B_n_num = quad(B_n_num_int, zeta+eps, 1-eps,
                   args=(zeta, Q, lmda_t, melt), limit=100)[0]
    B_n_den = quad(B_n_den_int, zeta+eps, 1-eps,
                   args=(lmda_t), limit=100)[0]
    B_n = B_n_num/B_n_den
    return B_n

def theta_solid(tau, zeta, Q, nterms, melt):
    ## Solid temperature profile for a given tau and zeta.
    # Initialize output arrays.
    eta = np.linspace(zeta+eps, 1-eps, 51)
    theta = np.zeros_like(eta)

    # Identify eigenvalues.
    lmda_t = lambdas_solid(zeta, nterms)

    # Sum terms
    for n in range(nterms):
        B = B_n(zeta, Q, lmda_t[n], melt)
        theta += B*np.exp(-lmda_t[n]**2*tau)*fn_til(eta, lmda_t[n])
    # Add steady-state impact
    theta += theta_solid_ss(eta, zeta, Q)
    return eta, theta

def interface_solution(tau_final, Q, St, nterms, melt):
```

```
## Generates tau-zeta pairs for a given case.
# Initialize time
if melt == True:
    zeta0 = eps
else: zeta0 = 1 - eps
mdt = 5e-3

def dzeta_dtau(tau, zeta, Q, St, nterms):
    ## Interface differential equation.
    # Initialize
    dzdt = 0
    # printing term included for telemetry during run
    print('{:.5f} \t {:.5f}'.format(float(tau), float(zeta)))

    # Generate eigenvalues
    lmda = lambdas_liquid(zeta, nterms)
    lmda_t = lambdas_solid(zeta, nterms)

    # Sum terms
    for n in range(nterms):
        A = A_n(zeta, Q, lmda[n], melt)
        liquid_term = A*lmda[n]*np.exp(-lmda[n]**2*tau)*j1(lmda[n]*zeta)

        B = B_n(zeta, Q, lmda_t[n], melt)
        solid_term = B*np.exp(-lmda_t[n]**2*tau) \
            *fn_til_prime(zeta, lmda_t[n])

        dzdt += liquid_term
        dzdt += solid_term

    # Add steady-state and Stefan number impacts
    dzdt += ((Q/4)*(zeta**2 - 1) + 1)/(zeta*np.log(zeta))
```

```
        dzdt *= St
        return dzdt

    ## Determine time range of run.
    t_range = [0, tau_final]
    tau = np.linspace(0, tau_final, 201)

    ## Full solution Set dense_output to True and view the soln varable
    # for additional debug information.
    soln = solve_ivp(dzeta_dtau, t_range, [zeta0],
                     method='RK45', t_eval=tau,
                     args=(Q, St, nterms),
                     dense_output=False, max_step=mdt)

    # Isolate output variables
    zeta = soln.y[0, :]
    tau = tau[:len(zeta)]
    return tau, zeta

    return 1

def theta_full(tau, zeta, Q, nterms, melt):
    ## Generate combined temperature profiles for a given tau and zeta.
    eta = np.zeros(102)
    theta = np.zeros_like(eta)
    eta[:51], theta[:51] = theta_liquid(tau, zeta, Q, nterms, melt)
    eta[51:], theta[51:] = theta_solid(tau, zeta, Q, nterms, melt)
    return eta, theta
```

Appendix B: CSHF Analytical Solution Python Script

Listing B.1: cshfcodedef script

```
# ==========================================================
# Library Imports
# ==========================================================
import numpy as np ## provides matrix support
from scipy.special import j0, j1, y0, y1, jn_zeros ## Bessel fcn family
from scipy.optimize import bisect ## basic root solver for transcendentals
from scipy.integrate import quad, solve_ivp ## quad for integration,
                                            ## solve_ivp for ODE solution

eps = 1e-4

# General order: eta, tau, zeta, Q_dot, Q_dp, lmda, lmda_t, n_roots, melt
# ==========================================================
# Liquid Region Definitions
# ==========================================================
def lambdas_liquid(zeta, nterms):
    ## Generates the eigenvalues for the liquid region.
    z_0_n = jn_zeros(0, nterms)
    lmda = z_0_n/zeta
    return lmda

def Phi_liquid(eta, Q_dot, melt):
    ## Initial temperature profile in liquid region.
    if melt == True:
        return 0
    else:
        return (Q_dot/4)*(1 - eta**2)
```

```
def theta_liquid_ss(eta, zeta, Q_dot):
    ## Steady-state liquid temperature profile.
    return (Q_dot/4)*(zeta**2 - eta**2)

def A_n(zeta, Q_dot, lmda, melt):
    ## Fourier coefficient for the liquid region. Term information is
    # communicated by lmda.
    def A_n_num_int(eta, zeta, Q_dot, lmda):
        ## Numerator integrand
        term1 = Phi_liquid(eta, Q_dot, melt) \
            - theta_liquid_ss(eta, zeta, Q_dot)
        term2 = j0(lmda*eta)*eta
        return term1*term2
    ## Denominator integrand
    def A_n_den_int(eta, lmda):
        return j0(lmda*eta)**2*eta

    # Integrate numerator and denominator separately then combine
    A_n_num = quad(A_n_num_int, eps, zeta-eps,
                   args=(zeta, Q_dot, lmda), limit=100)[0]
    A_n_den = quad(A_n_den_int, eps, zeta-eps,
                   args=(lmda), limit=100)[0]
    A_n = A_n_num/A_n_den
    return A_n

def theta_liquid(tau, zeta, Q_dot, Q_dp, nterms):
    ## Liquid temperature profile for given tau and zeta.
    # Initialize output arrays.
    eta = np.linspace(eps, zeta-eps, 51)
    theta = np.zeros_like(eta)
```

```
    # Determine regime.
    if (Q_dot/2) > Q_dp:
        melt = True
    else:
        melt = False
    # Identify eigenvalues.
    lmda = lambdas_liquid(zeta, nterms)

    # Sum terms
    for n in range(nterms):
        A = A_n(zeta, Q_dot, lmda[n], melt)
        theta += A*np.exp(-lmda[n]**2*tau)*j0(lmda[n]*eta)
    # Add steady-state impact
    theta += theta_liquid_ss(eta, zeta, Q_dot)
    return eta, theta

# ============================================================
# Solid Region Definitions
# ============================================================
def lambdas_solid(zeta, nterms):
    ## Generates the eigenvalues for the solid region.
    def B_n_transcendental(lmda):
        return j0(lmda*zeta)*y1(lmda) - y0(lmda*zeta)*j1(lmda)

    def transcendental_roots(nterms):
        ## Find n roots of the quasi-periodic transcendental equation.
        # Initialization
        roots = np.zeros(nterms)
        margin = 1e-8
        # Manually perform first root
        left, right = margin, 3.9/(1 - zeta)
        roots[0] = bisect(B_n_transcendental, left, right, maxiter=2000)
```

```
        # Iterate the rest of the roots
        for i in range(1, nterms):
            left = roots[i-1] + margin
            right = roots[i-1] + 3.65/(1 - zeta)
            roots[i] = bisect(B_n_transcendental, left, right,
                maxiter=2000)
        return roots

    lmda_t = transcendental_roots(nterms)
    return lmda_t

def Phi_solid(eta, Q_dot, melt):
    ## Generate initial solid temperature profile.
    if melt == True:
        return -(Q_dot/4)*eta
    else:
        return 0

def theta_solid_ss(eta, zeta, Q_dot, Q_dp):
    ## Generates steady-state solid temperature profile.
    term1 = (Q_dot/4)*(zeta**2 - eta**2)
    term2 = Q_dot/2 - Q_dp
    term3 = np.log(eta/zeta)
    return term1 + term2*term3

def fn_til(eta, lmda_t):
    ## Generates the eigenfunction for the solid region.
    return j0(lmda_t*eta) - (j1(lmda_t)/y1(lmda_t))*y0(lmda_t*eta)

def fn_bar(zeta, lmda_t):
    term1 = -j1(lmda_t*zeta)
    term2 = (j1(lmda_t)/y1(lmda_t))
```

```
    term3 = j1(lmda_t*zeta)
    return term1 + term2*term3

def B_n(zeta, Q_dot, Q_dp, lmda_t, melt):
    ## Fourier coefficient for the solid region.
    # Numerator integrand.
    def B_n_num_int(eta, zeta, Q_dot, lmda_t):
        # Determine regime
        if (Q_dot/2) > Q_dp:
            melt = True
        else:
            melt = False
        # Set term
        term1 = Phi_solid(eta, Q_dot, melt) \
            - theta_solid_ss(eta, zeta, Q_dot, Q_dp)
        term2 = fn_til(eta, lmda_t)*eta
        return term1*term2
    # Denominator integrand
    def B_n_den_int(eta, lmda_t):
        return fn_til(eta, lmda_t)**2*eta

    # Integrate numerator and denominator separately and combine.
    B_n_num = quad(B_n_num_int, zeta+eps, 1-eps,
                   args=(zeta, Q_dot, lmda_t), limit=100)[0]
    B_n_den = quad(B_n_den_int, zeta+eps, 1-eps,
                   args=(lmda_t), limit=100)[0]
    B_n = B_n_num/B_n_den
    return B_n

def theta_solid(tau, zeta, Q_dot, Q_dp, nterms):
    ## Solid temperature profile for given tau and zeta.
    # Initialize output arrays.
```

```
    eta = np.linspace(zeta+eps, 1-eps, 51)
    theta = np.zeros_like(eta)

    # Set regime.
    if (Q_dot/2) > Q_dp:
        melt = True
    else:
        melt = False

    # Identify eigenvalues.
    lmda_t = lambdas_solid(zeta, nterms)

    # Sum terms
    for n in range(nterms):
        B = B_n(zeta, Q_dot, Q_dp, lmda_t[n], melt)
        theta += B*np.exp(-lmda_t[n]**2*tau)*fn_til(eta, lmda_t[n])
    # Add steady-state impact
    theta += theta_solid_ss(eta, zeta, Q_dot, Q_dp)
    return eta, theta

def interface_solution(tau_final, Q_dot, Q_dp, nterms):
    ## Generates tau-zeta pairs for a given case.

    # Determine regime and settings
    """
    Note: Stability issues require slightly different solver settings
    between the melting and solidification case -- values of zeta ~ 1
    are highly stiff, necessitating an implicit solver for viable time
    steps. This problem is less acute for melting cases, and better
    accuracy is achieved using the explicit Runge-Kutta 45 solver.
    """
    if (Q_dot/2) > Q_dp:
```

```
        melt = True
        method = 'RK45'
        zeta0 = eps
        mdt = 2e-3
    else:
        melt = False
        method = 'Radau'
        zeta0 = 1 - eps
        mdt = 2e-3

    def dzeta_dtau(tau, zeta, Q_dot, Q_dp, nterms):
        ## Interface differential equation
        # initialize value
        dzdt = 0
        # printing term included for telemetry during run
        print('{:.5f} \t {:.5f}'.format(float(tau), float(zeta)))

        # Generate eigenvalues
        lmda = lambdas_liquid(zeta, nterms)
        lmda_t = lambdas_solid(zeta, nterms)

        # Sum terms
        for n in range(nterms):
            A = A_n(zeta, Q_dot, lmda[n], melt)
            liquid_term = A*lmda[n] \
                *np.exp(-lmda[n]**2*tau)*j1(lmda[n]*zeta)

            B = B_n(zeta, Q_dot, Q_dp, lmda_t[n], melt)
            solid_term = B*lmda_t[n]*np.exp(-lmda_t[n]**2*tau) \
                *fn_bar(zeta, lmda_t[n])

            dzdt += liquid_term
```

```
            dzdt -= solid_term
        # Add steady-state impact
        dzdt += (Q_dot/2 - Q_dp)/zeta
        return dzdt

    ## Determine time range of run.
    t_range = [0, tau_final]
    tau = np.linspace(0, tau_final, 201)

    ## Full solution. Set dense_output=True and view soln variable
    # for additional and debug information.
    soln = solve_ivp(dzeta_dtau, t_range, [zeta0],
                     method=method, t_eval=tau,
                     args=(Q_dot, Q_dp, nterms),
                     dense_output=False, max_step = mdt)

    zeta = soln.y[0, :]
    tau = tau[:len(zeta)]
    return tau, zeta

def theta_full(tau, zeta, Q_dot, Q_dp, nterms):
    ## Generates combined temperature profiles by calling functions.
    eta = np.zeros(102)
    theta = np.zeros_like(eta)
    eta[:51], theta[:51] = theta_liquid(tau, zeta, Q_dot, Q_dp, nterms)
    eta[51:], theta[51:] = theta_solid(tau, zeta, Q_dot, Q_dp, nterms)
    return eta, theta
```