

Control of Heat and Fluid Flow in Solidification Processes by an Inverse Method

Rui Xu ©

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS OF THE MSc Eng DEGREE
IN
CONTROL ENGINEERING
FACULTY OF ENGINEERING
LAKEHEAD UNIVERSITY
THUNDER BAY, ONTARIO, CANADA.**

December 17, 1999



**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

**395 Wellington Street
Ottawa ON K1A 0N4
Canada**

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

**395, rue Wellington
Ottawa ON K1A 0N4
Canada**

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-52088-9

Canada

Abstract

The objective of this research is to produce a control scheme by an inverse method that is capable of acting fast enough to be useful in a real-time manufacturing environment. The scheme should provide a transient boundary condition that produces a prescribed interfacial surface motion with a corresponding desired morphology at the phase interface in solidification problems. In this thesis, an inverse technique is presented for one and two-dimensional heat transfer problems with phase change. It calculates the required boundary temperature to provide a specified interface velocity at the solid-liquid interface. An entropy-based method is used to improve the stability of the proposed algorithm. The effects of free convection in the liquid are also considered. A control-volume-based finite element method is employed for the numerical solution of the conservation equations for mass, momentum and energy. Numerical examples are presented in order to demonstrate the promising capabilities and performance of the proposed formulation. In practical solidification problems, this method can lead to stable and meaningful results.

Acknowledgements

I would like to acknowledge Dr. G. F. Naterer for his encouragement, help and insightful comments during every aspect of this thesis. He supervised me patiently and his erudite knowledge made it possible to achieve good performance in my work. The assistance of Professors Natarajan and Saliba is also appreciated.

Financial support from the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged.

Certainly, the greatest contribution to this effort has been the patience and encouragement of my family, particularly my husband and my son.

Nomenclature

X_0	reference length
x, y	spatial coordinates
c	specific heat
k	thermal conductivity
L	latent heat
H	enthalpy
t	time
s, τ	local coordinate
Δt	time step
T	temperature
ρ	density
V, u, v	velocity
p	pressure
R	sensitivity coefficient
N	shape function
\dot{S}	source term
\dot{P}	entropy production rate
Pr	Prandtl number
Ra	Rayleigh number
Ste	Stefan number
e	internal energy

Greek

Φ	general scalar variable
Γ	thermophysical property
α	thermal diffusivity
β	thermal expansion coefficient
ν	kinematic viscosity

Subscripts

l	liquid
r	reference
s	solid
$p+1$	nodal point

Superscripts

m iterative counter

Contents

1	Introduction	1
1.1	Background	1
1.2	The Inverse Heat Conduction Problem (IHCP)	4
1.3	Inverse Heat Transfer Problem with Phase Change	6
1.4	Effects of Convection on Solidification	10
1.5	Scope of Present Study	11
2	Mathematical Model	13
2.1	Descriptions of Inverse Solidification Problems	13
2.1.1	Description of Inverse Stefan Problem	14
2.1.2	Description of Inverse Stefan Problem with Fluid Flow	15
2.2	Governing Equations	16
2.2.1	Definitions of Dimensionless Variables	16
2.2.2	Governing Equations	17
2.2.3	The General Conservation Equation	21
2.2.4	Special case: Stefan Problem without fluid flow	22
2.2.5	Enthalpy Equation of State	22
2.3	Boundary Conditions and Initial Conditions	23
3	Numerical Model	25
3.1	Discretization of Solution Domain	26
3.2	Discretization of Governing Equations	29
3.2.1	Conservation of Mass, Momentum Equations	29
3.2.2	Conservation of Energy Equation	32
3.2.3	Boundary Conditions	33
3.3	Sensitivity Coefficient	33
3.4	Entropy Based Method for Numerical Stability	36
3.4.1	The Second Law and Generalized Entropy	37
3.4.2	Improving Stability of Inverse Method	39
3.5	Solution Procedure	42

4 Applications and Results	44
4.1 Stefan Problem with Variable Interface Velocity	44
4.2 Stefan Problem with Constant Interface Velocity	48
4.3 Solidification of Pure Aluminum	56
4.3.1 One-dimensional Problem	56
4.3.2 Two-dimensional Problem	58
4.4 Solidification Problem for Pure Gallium with the Effects of Fluid Flow	60
5 Conclusions and Recommendations	66
5.1 Conclusions	66
5.2 Recommendations for Future Research	67

List of Figures

1.1	Sample experimental apparatus	2
1.2	One-dimensional solidification	8
2.1	Inverse Stefan problem, Case (a)	14
2.2	Inverse Stefan problem, Case (b)	14
2.3	Inverse Stefan problem with fluid flow	15
3.1	A single element	27
3.2	Control volume	28
4.1	$T(0, t)$ for pure gallium as a function of time for three different interface velocities ($\alpha = 1.3762 * 10^{-5} m^2/s$, $L = 80160 J/kg$, $c_p = 381.5 J/kg^{\circ}C$, $T_0 = 29.8^{\circ}C$)	50
4.2	Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.1)	51
4.3	Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.05)	51
4.4	Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.005)	52
4.5	Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.05, $Ste=0.2$)	53
4.6	Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.05, $Ste=0.05$)	54
4.7	Boundary temperature at $x=0$ with $\Delta t = 0.05$ (no entropy based modification)	55
4.8	Boundary temperature at $x=0$ with $\Delta t = 0.05$ (with entropy based modification)	55
4.9	One-dimensional solidification of pure aluminum	56
4.10	Boundary temperature for solidification process of pure aluminum (time step=125s)	58
4.11	Two-dimensional solidification of pure aluminum	59
4.12	Mesh structure	60

4.13	Boundary temperature at mid-point for solidification of pure aluminum	61
4.14	Boundary temperature for solidification of pure aluminum	62
4.15	Solidification of pure gallium in a rectangular enclosure	63
4.16	Boundary temperature for solidification with fluid flow	63
4.17	Distribution of velocity in liquid region (same scale for magnitude) . . .	64

List of Tables

4.1	Boundary temperature at $x=0$ for Stefan problem with variable interface velocity (with 40×5 element mesh)	46
4.2	Boundary temperature at $x=0$ for Stefan problem with variable interface velocity (with 20×5 element mesh)	47
4.3	Boundary temperature at $x=0$ for Stefan problem with variable interface velocity (with 10×5 element mesh)	47
4.4	Time step sensitivity (with $t=0.1$)	50
4.5	Time step sensitivity (with $t=0.8$)	52
4.6	Thermal properties of pure aluminum	57
4.7	Thermal properties of pure gallium	64

Chapter 1

Introduction

1.1 Background

In many industrial processes, thermal and fluid flow controls are required in a real-time mode to improve processing conditions. It has received considerable attention due to its wide practical applications. Specific examples would be in metallurgy, such as casting and molding processes. Controlling the heat transfer and fluid flow patterns is important since it largely affects the casting quality and production time in materials processing. The heat fluxes and velocities at the solidification interface determine the casting or molding properties and structures. They can be controlled by utilizing measurements, such as thermocouple temperature measurements, during a process and then adjusting the external boundary conditions at the inlet in a real-time mode to give desired results. Another example includes various aspects of biological tissue destruction or preservation by alternating freezing-thawing cycles. The survival rate of preserved blood cells is dependent upon the rate of temperature change during freezing and preservation. Information on the cooling rates that will insure optimum cell survival rates can be employed for improved control of the freezing process.

Demirci and Coulter [1] implemented this type of control successfully in molding processes. Given a mold shape with specified inlet gate and vent locations, one can identify the desired flow progression schemes. Experiments were conducted with a two-dimensional, irregular mold cavity to test the performance of the control unit during

actual injection molding processes. Two inlet gates and three different desired flow progression schemes were considered. The experimental apparatus in the injection mold filling experiments during this study is shown in Fig. 1.1. The assembly includes a two-dimensional transparent mold cavity with two inlets (supplied by two tubing pumps with corresponding pump controllers) and embedded sensors. Liquid was supplied from the resin tank. Also, a video camera was used to monitor the flow patterns arising during the injection mold filling experiments.

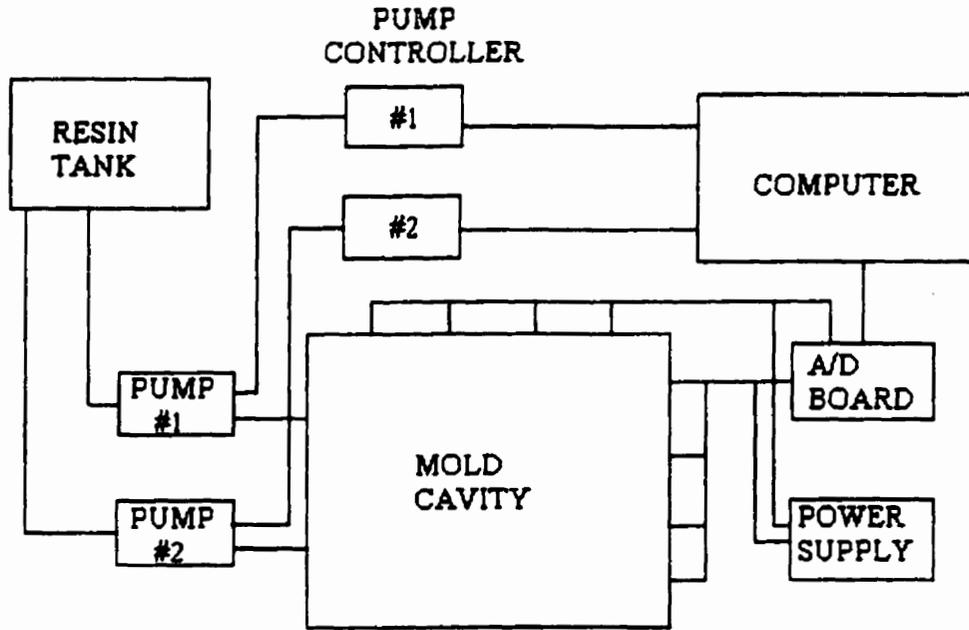


Figure 1.1: Sample experimental apparatus

The control method can be summarized as follows. At any given time, measurements of the interface position were obtained. Then, using a Fortran program, they compared this present interface position with the interface position along the desired flow progression path identified earlier. In this way, they determined the desired interface position that was closest to and following the present interface position. This position can be called the desired interface position of interest for the given time step. The algorithm was then repeated many times. At each instance, the input was set to the present interface position plus a different inlet flow rate combination. By running

this step repeatedly with different inlet flow combinations, the interface position can be calculated for each time step. Also, they compared the resulting predicted interface position from each output with the desired interface position of interest and found the predicted interface position that was closest to the desired interface position at the given time level. The inlet flow rate combination that gave the predicted interface position was the best choice for the current time step. This procedure was then repeated for subsequent time steps until the mold cavity is filled.

In this thesis, a deterministic control strategy, based on solutions of the governing transport equations, will be adopted, rather than neural network control in the previous example.

In this procedure, we can have a "forward model" or an "inverse model". In the forward modeling case, a present interface position and an inlet flow rate combination are given and the model is expected to predict the shape and location of the next interface. In the inverse modeling case, a present interface position and a next interface position are given and it is expected to predict the inlet flow rate combination that provides these two consecutive interface positions. It was shown in the previous study that an inverse model can be more useful than a forward model because the inverse model directly calculates the appropriate inlet flow rate combination at any particular time. The forward model, on the other hand, gives one of many possible subsequent interface positions. A search method must then be employed to find the optimal inlet flow rate combination at each time step that leads to the best of all possible subsequent interface positions.

The control scheme using a forward model requires the numerical calculation of the direct heat transfer problem. Conversely, the control scheme using an inverse model requires the numerical calculation of the inverse heat transfer problem.

Most heat transfer problems are concerned with the determination of temperatures at interior points when certain initial and boundary conditions are given, such as speci-

fied temperature or heat flux boundary conditions. These problems are called "direct" problems because the solutions involve a direct integration of differential equations with known initial conditions. Numerous research studies have been performed on this topic. The methods for these problems have been well developed.

On the other hand, the inverse heat transfer problem is concerned with the estimation of the required boundary heat flux or temperature values based on internal measurements or a desired internal process behaviour. It is well known that an inverse problem is much more difficult to solve numerically than a direct problem. Its solution strongly depends upon the amount of error in the prescribed data for two main reasons. Firstly, in the direct problem, any fluctuations in the applied heat flux are damped as the heat diffuses through the solid medium. In the inverse problem, the opposite situation occurs. The fluctuations or noise in the measurements will be amplified in the projection back to the boundary, and the resulting surface condition estimates can be easily overwhelmed by the noise in the interior measurement. Secondly, the physics of heat conduction introduces a natural lag between the applied boundary temperature or heat flux and the temperature response at interior points. Thus, a step change in the surface temperature or heat flux will not be fully felt in the interior until a finite amount of time has passed.

The inverse heat transfer problem has been applied to two main areas: (i) the Inverse Heat Conduction Problem (IHCP), and (ii) problems involving phase change.

1.2 The Inverse Heat Conduction Problem (IHCP)

In the Inverse Heat Conduction Problem, the desired surface temperatures or heat fluxes are estimated by utilizing transient temperature measurements at one or more interior locations of the specified domain.

The inverse heat conduction problem has been the subject of considerable research. Several methods have been proposed to solve the IHCP. It is worth mentioning that

Stolz [2] presented one of the first studies of inverse heat transfer problems with a numerical method. Stolz formulates the inverse problem as though it is a direct problem and obtains an integral equation for the unknown surface condition. The solution is obtained by numerically inverting the integral equations. This procedure becomes unstable if the time step becomes small. Burggraf [3] found an exact series solution to a one-dimensional inverse problem. The temperature and heat flux histories are given at a single internal point. Approximate results are found if discrete or experimental data are used.

The well-known methods for the solution of the IHCP are the sequential function specification method [4], the regularization method [5] and the combined function specification and regularization (CFSR) method.

(i) The Sequential Function Specification Method

Beck [4] has stabilized the IHCP by using several future-time temperatures with a least-squares method to calculate components of the heat flux at a given time. This method is called the Sequential Function Specification Method. It can improve the numerical stability of IHCP solutions effectively. The main idea of this method is given as follows. A temporary assumption is made that the heat fluxes or the temperatures are constant over r future times; this assumption will be subsequently removed. An estimate is sought for the value of heat flux or temperature, constant over r future time steps, that minimizes the least squares error between the computed and measured sensor temperatures. Many authors have applied this approach, i.e. Bass et al [6] and Osman and Beck [7].

(ii) The Regularization Method

The Regularization method was proposed by Tikhonov and Arsenin [5]. It is a procedure that modifies the least squares approach and provides stability by adding smoothing factors that are intended to reduce the influence of measurement error in

the data. An augmented sum-of-squares function is minimized. This method involves the whole domain in the sense that it utilizes all of the data to estimate simultaneously the heat flux components. Thus, if the solution is needed over a large domain and a long period of time, the dimensions of the matrices and the number of computations involved increase accordingly. The influence of the regularization component is determined by the magnitude of a regularization parameter. Different criteria are found in the literature for the selection of this parameter, i.e. Tikhonov and Arsenin [5], Murio [8], Scott and Beck [9].

(iii) Combined Function Specification-Regularization (CFSR) Method

This method is also presented for the solution of the IHCP by Beck and Murio [10] and Osman and Dowding [11]. It is used to stabilize and smooth the estimates of the surface heat flux distribution. It differs from the global regularization methods in that the solution is found sequentially, thereby greatly improving computational efficiency. This method was shown to be very competitive with the global regularization methods in terms of the heat flux estimates. In studies by Beck and Murio [10], Tikhonov regularization was applied to the time variation of heat flux. Also, Osman and Dowding [11] apply the regularization to the spatial variation of the heat flux.

1.3 Inverse Heat Transfer Problem with Phase Change

Problems involving phase change are accompanied by either absorption or release of latent energy (melting or solidification). A moving boundary exists and separates the two phases of differing thermo-physical properties where latent energy is either absorbed or liberated. The heat transfer processes now become more complicated in comparison to the IHCP.

In inverse problems with phase change, the temperature and the interface velocity may be prescribed at the phase interface. The required temperature and heat flux at

the stationary boundary of the domain are then unknown and must be determined by the analysis. Due to the presence of the moving interface, the above problems may be strongly non-linear. There are only limited studies on the subject of inverse heat transfer with phase change.

We will consider previous methods that have been proposed for the solution of this problem. They can be categorized into the analytical method, front-fixing method, front-tracking method and the fixed domain method.

(i) Analytical Method

There are only a limited number of analytical solutions available, which are mainly for one-dimensional problems. Rubinsky and Shitzer [12] derived analytic infinite-series solutions to the one-dimensional inverse Stefan problem in Cartesian and spherical coordinates. Conduction heat transfer was assumed in the liquid region. Frederick and Grief [13] presented another method, which can be used to obtain the solution of problems involving phase change. The solution in one of the phases is specified as a known single-phase solution. An inverse analysis then determines the solution for the other phase.

(ii) Front-fixing Method and Front-tracking Method

In the Front-fixing method and the Front-tracking method, the solid and liquid regions are treated separately and the phase change interface is explicitly handled as a moving boundary.

For the one-dimensional case (Fig. 1.2), the problem involves consideration of the conservation of energy in the domain by dividing it into two distinct domains, Ω_l and Ω_s , where $\Omega_l + \Omega_s = \Omega$. The energy conservation is written as two separate equations in the liquid (Ω_l) and solid (Ω_s) regions, respectively, i.e.

$$\rho_l c_l \frac{\partial T}{\partial t} = k_l \frac{\partial^2 T}{\partial x^2} \quad (1.1)$$

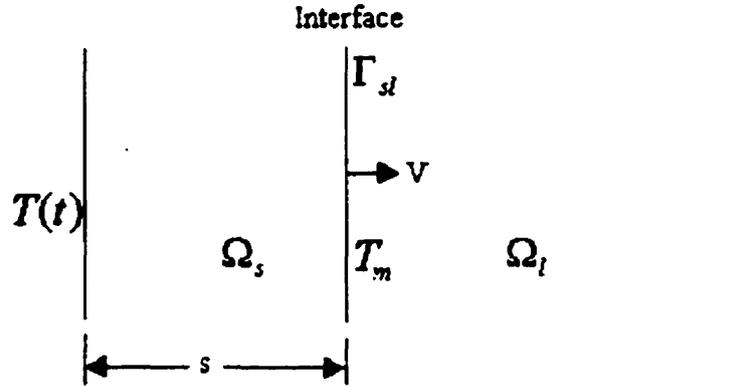


Figure 1.2: One-dimensional solidification

$$\rho_s c_s \frac{\partial T}{\partial t} = k_s \frac{\partial^2 T}{\partial x^2} \quad (1.2)$$

where the subscripts l and s denote liquid and solid, respectively. The complete description of the problem involves, in addition to the initial conditions and the appropriate external boundary conditions, the interface conditions on the phase change boundary, Γ_{sl} , which are

$$T_{sl} = T_m \quad (1.3)$$

$$-k_s \left(\frac{\partial T}{\partial x} \right)_s = \rho_s L \frac{ds}{dt} - k_l \left(\frac{\partial T}{\partial x} \right)_l \quad (1.4)$$

where sl represents the position of the interface. ds/dt is the interface velocity and T_m is the phase change temperature. Eq. (1.4) states that the heat transferred by conduction in the solidified portion is equal to the heat entering the interface by latent heat liberation at the interface and the heat coming from the liquid by conduction.

In the front-fixing method, the moving interface is fixed with a proper coordinate transformation and the interface becomes effectively stationary. For inverse problems where the interface position is specified a priori, it will be shown that these transformation techniques offer significant advantages. However, these methods introduce numerical complications. Zabaras [14] used a front-fixing method to obtain the solution of several one-dimensional inverse solidification problems. A finite element method

with a future time-stepping technique was employed in this study.

Front-tracking methods involve deforming and moving finite elements or finite difference grids. The mesh is continuously moving to adapt to the freezing interface motion. Katz and Rubinsky [15] developed a "front-tracking" finite element method for the study of the inverse one-dimensional conduction heat transfer problem with phase change. An inverse method for the study of two-dimensional problems with phase change was first reported by Hsu and Rubinsky [16]. An inverse finite element method was described for the analysis of two-dimensional stationary arc welding processes. This method can be used to predict the position of the solid-liquid interface as a function of several variables. Zabarvas and co-workers [17, 18] solved two-dimensional phase change problems by a front-tracking method using the boundary element formulation. The boundary element analysis was used in conjunction with Beck's sensitivity analysis. The solutions can be stabilized by a future time-stepping technique.

These two methods offer the advantage of being able to clearly separate the two phases along a boundary delineated by the discretization mesh. However, the mathematical algorithms required to appropriately adjust the mesh are generally very complex. They are usually difficult to implement, computationally intensive and can seldom be incorporated into existing numerical codes.

(iii) Fixed-domain Methods

Fixed-domain methods do not change the initial spatial discretization mesh. They involve the solution of a continuous system with an implicit representation of the phase change. The problem is formulated in such a way that the interface conditions become implicit in a new form of the equations, which applies over the entire fixed domain. For example, in the Enthalpy Method, a single energy conservation equation is written for the whole domain as

$$\rho \frac{\partial H}{\partial t} = k \frac{\partial^2 T}{\partial x^2} \quad (1.5)$$

where H is the enthalpy function and only transient heat conduction is considered. The enthalpy change is written in terms of specific heat multiplied by temperature change.

Then, appropriate heat source terms, or modified (apparent) specific heat, involving latent heat, are introduced and a single differential equation is solved over the whole domain (solid and liquid). This method has the great advantage of simpler numerical implementation and an easier incorporation into existing codes. Voller [19] introduced an enthalpy method to solve the inverse one-dimensional Stefan problem. The discretized equation is obtained by a control volume approach and a time-stepping scheme. In solving this equation, finite differences are used and the value of the liquid fraction is used to track the progress of the solid-liquid phase interface. The determination of time step and numerical grid is based on the prescribed movement of the phase interface. The method requires that at any time step the phase interface moves from one node point to the next point. An iterative technique is adopted such that the solution provides a specified movement of the phase interface.

1.4 Effects of Convection on Solidification

In some cases, it is sufficient that phase change problems can be analyzed with conduction heat transfer only. However, convection cannot be neglected in other cases of practical interest. Liquid flow during solidification affects the morphology of the solid/liquid interface, solidification rate and the temperature distribution. It may be necessary to take into account convection effects in the inverse solidification problem.

Convection heat transfer problems are normally divided into two broad categories, namely forced convection and free (or natural) convection. In forced convection, the fluid motion is due to externally applied pressure or viscous forces. When the fluid motion is produced by buoyancy forces, which are usually due to temperature or concentration gradients in the flow, the process is called free or natural convection.

There is an important difference between forced and natural convection. In the for-

mer case, the flow is externally imposed and is often independent of the temperature field. The flow field can thus be obtained independent of the heat transfer processes and then used in the determination of the temperature field. In natural convection, on the other hand, the flow and heat transfer mechanisms are typically linked, since the flow itself arises due to the temperature differences in the fluid. As a result, the flow field cannot be obtained independent of the temperature field and the two processes must be considered simultaneously. Because of these complexities, natural convection flows are generally more difficult to solve (analytically and numerically) as compared to corresponding forced flow circumstances.

Several papers have been published on the numerical calculation of the direct solidification problem with natural convection. One of the first significant investigations was by Sparrow et al. [20]. They solved an axially symmetric solidification problem using an implicit finite difference method. Marshall et al. [21] and Strada and Heinrich [22] developed a finite element method that computed heat transfer rates for steady-state natural convection in a rectangular enclosure involving high Rayleigh numbers. McDaniel and Zabaras [23] presented their results on two-dimensional solidification and melting problems for pure metals using a least-squares front-tracking finite element method. Only few studies have considered the numerical simulation of inverse solidification with the effects of natural convection. Voller [24] published results that extend the inverse method to solidification and melting problems of metals in which fluid flow occurs in the melt.

In this thesis, the effects of natural convection on inverse solidification problem will be considered.

1.5 Scope of Present Study

It is apparent that numerical simulation of heat transfer plays a very important role in the control strategy as mentioned earlier. In particular, an inverse method for heat transfer problems is useful for control implementation in solidification processes.

This thesis will address the numerical simulation of the inverse heat transfer problem with phase change for solidification processes. A Fixed Domain Method will be used for one and two-dimensional solidification problems such that the phase interface moves uniformly and satisfies a desired progression. The solidification problem with and without fluid flow will be considered. During the numerical calculation, since the Second Law of Thermodynamics should be satisfied, an entropy-based method is introduced to improve the numerical stability of the formulation. A Finite Element Method is employed for the numerical discretization. In order to demonstrate the promising capabilities and performance of the current formulation, some examples involving conduction with phase change will be presented. These include one-dimensional and two-dimensional problems with an interface moving at a constant velocity or a specified variable velocity. Also, the numerical results for an example that includes the effects of fluid flow will be presented.

It is anticipated that the contributions of the present work can be realized in three regards. Firstly, a control volume based finite element discretization is employed in the formulation of the inverse procedure. In this way, the model is conservation based while retaining the geometric flexibility of the finite element method. Secondly, an entropy based technique is proposed for the improvement of numerical stability in the inverse method. It considers local violations of the Second Law, due to discretization errors, as a criterion for a corrective strategy in the computations. The magnitude of negative entropy production can be used in a quantitative correction of the apparent thermal conductivity. It will be shown that this approach provides an effective alternative to previous inverse stabilizing techniques such as future time stepping. Finally, the model is extended to a problem considering natural convection. As a result, the formulation appears capable of handling phase change with fluid flow. The above developments can provide important contributions in terms of control engineering in coupled thermal/fluid processes involving solidification and melting applications.

Chapter 2

Mathematical Model

2.1 Descriptions of Inverse Solidification Problems

A solidification problem occurs when a substance has a transformation point whereby it changes from liquid to solid with an emission of latent heat. The essential feature of such problems is the existence of a moving interface with respect to the substrate, i.e. in some cases, a solidification problem requires only motion with respect to the material which itself is in motion.

Heat transfer problems dealing with phase change at a constant temperature (i.e. pure materials) are referred to as Stefan problems. The Stefan problem involves the solidification or melting of a pure material and it is characterized by a distinct moving phase change boundary with respect to the material at which a heat balance condition has to be satisfied. The way in which this interface moves is typically controlled by the boundary temperature; in other words, the temporal location of the interface between the phases is given and the required temperature at the fixed boundary is calculated. This is the so-called inverse Stefan problem. For the inverse design problem, it is desirable to approximate the boundary temperature history that would produce a prescribed interfacial motion. With this information, one can hopefully develop a controlled experiment that produces the described boundary condition to ensure a desired motion of the phase interface in a practical application.

2.1.1 Description of Inverse Stefan Problem

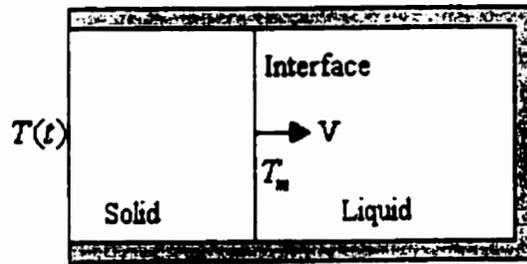


Figure 2.1: Inverse Stefan problem, Case (a)

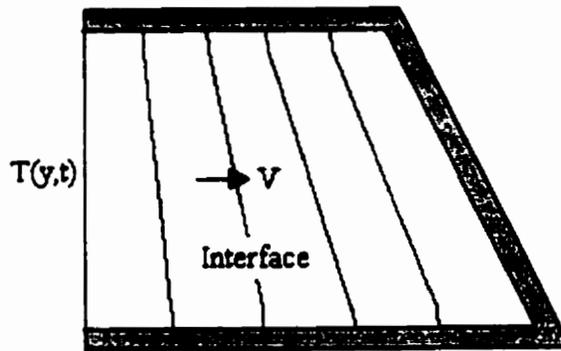


Figure 2.2: Inverse Stefan problem, Case (b)

Consider solidification in a region (Figs. 2.1-2.2) occupied initially by a pure liquid at a temperature $T_{in}(x)$, with T_m denoting the melting temperature. Since the top, bottom and right boundaries of the cavity are insulated, no temperature gradients will occur in the liquid region normal to these boundaries and the position of the interface will be controlled by the temperature at the left boundary of the cavity. The conductivity, K , density, ρ , and specific heat, c , are assumed to be temperature independent. The melting temperature, T_m , is given. For case (a), it is assumed that the interface moves in the x direction and the shape of the interface is a vertical straight line at all times and the velocity of the interface can be constant or variable. For case (b), it is assumed that the interface velocity in the y direction varies according to position and the shape

of the interface is a straight line. In the above example, all points on the interface reach the right boundary at same time. We will solve these problems to assess how the temperatures of the left boundary should vary with time to produce the desired interface motion. In case (a), the temperature at the left boundary will be uniform and it will only change with time, $T_0(t)$. It can be treated as a one-dimensional heat transfer problem although the discretized domain will be two-dimensional. In case (b), we have a two-dimensional problem, where the temperature at the left boundary changes with both time and position y , $(T_0(t, y))$.

2.1.2 Description of Inverse Stefan Problem with Fluid Flow

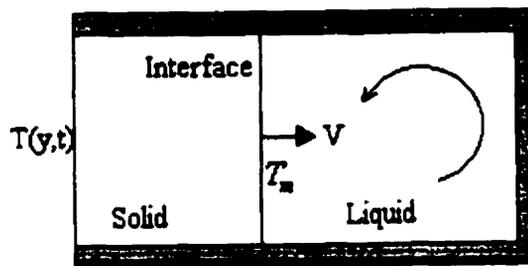


Figure 2.3: Inverse Stefan problem with fluid flow

Consider solidification in a rectangular region (Fig. 2.3) occupied initially by a pure liquid of temperature $T_m(x)$. When the liquid is superheated, fluid flow will play a role in heat transfer. The position of the interface is controlled by the temperature of the left boundary in the cavity. Given the interface velocity and assuming that this interface moves uniformly with a vertical profile at all times, then, due to fluid flow and convection, the temperature of the left boundary would not be uniform. We will solve this problem to assess how the temperature of the left boundary should vary. This temperature at the left boundary will be a function of time and position y , $(T_0(t, y))$.

The problem occurs in a two-dimensional domain and fluid flow in the liquid region makes it a fully two-dimensional inverse heat transfer problem with phase change.

2.2 Governing Equations

The processes of heat transfer and fluid flow are governed by fundamental principles from thermodynamics and mechanics. These principles are the conservation of mass, momentum, and energy. The conservation principles are general statements that are applied in a local sense, leading to the continuity, momentum and energy equations. These equations describe physical processes and they are the mathematical statements of three fundamental principles: (1) mass is conserved; (2) $F = ma$ (Newton's Law); (3) energy is conserved. The phase change process occurs within the energy conservation equation.

2.2.1 Definitions of Dimensionless Variables

The equations of heat transfer and fluid flow may be non-dimensionalized to achieve certain objectives. Firstly, it would provide conditions upon which dynamic and energy similarity may be obtained for geometrically similar problems. Secondly, the solution of these equations would usually provide convenient values within limits between zero and one.

In this work, the governing equations will be written in a dimensionless form. We will define the following dimensionless variables,

$$\Gamma_k^* = \frac{\Gamma_k}{\Gamma_l} \quad (2.1)$$

$$\Phi_k^* = \frac{\Phi_k}{\Phi_0} \quad (2.2)$$

$$x^* = \frac{x}{X_0} \quad (2.3)$$

$$t^* = \frac{\alpha_l t}{X_0^2} \quad (2.4)$$

where Γ represents any material property, such as conductivity or density. Φ represents a general dependent variable. X_0 is a reference length and α_l is the thermal diffusivity in the liquid. For the individual scalar fields, the reference scalar variables are

$$v_0 = \frac{\alpha_l}{X_0} \quad (2.5)$$

$$p_0 = \rho_l v_0^2 \quad (2.6)$$

$$e_0 = \frac{e - e_0}{c_l \Delta T_0} \quad (2.7)$$

$$\Delta T_0 = T_i - T_w \quad (2.8)$$

The subscripts i and w refer to reference temperature, they can be taken as maximum and minimum temperatures, respectively.

The key dimensionless quantities are the Stefan number (Ste), Prandtl number (Pr) and the Rayleigh number (Ra). They are defined as

$$Ste = \frac{c_l \Delta T_0}{L} \quad (2.9)$$

$$Pr = \frac{\nu}{\alpha_l} \quad (2.10)$$

$$Ra = \frac{g \beta \Delta T_0 X_0^3}{\nu \alpha_l} \quad (2.11)$$

where L is the latent heat. β is the thermal expansion coefficient and ν is kinematic viscosity.

In all subsequent governing equations, all variables are in a dimensionless form without the superscripts * (removed for brevity).

2.2.2 Governing Equations

The solidification problem with fluid flow may arise from natural convection in the liquid and two-phase regions. The governing equations are obtained by applying the principles of conservation of mass, momentum and energy to a finite control volume located at a given position in the region. In this situation, mass, momentum and energy equations need to be considered simultaneously.

(i) Mass equation

Consider the model for a finite control volume fixed in the region. The fundamental physical principle that mass is conserved is applied to this control volume. It means that

$$\left(\begin{array}{c} \text{time rate of decrease of} \\ \text{mass inside control volume} \end{array} \right) = \left(\begin{array}{c} \text{Net mass flow out of control} \\ \text{volume through surface} \end{array} \right) \quad (2.12)$$

Assuming the fluid is incompressible and the density remains constant, the mass equation is written as:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (2.13)$$

When phase transition is considered, the this mass balance is applied in each phase separately, and then results from all phases are summed to give the mixture equation. The form of the resulting equation appears identical to the above equation, but dependent scalars (i.e. u, v) are interpreted instead as the mass fraction-weighted sum of individual phase values, i.e. $v = f_s v_s + f_l v_l$ is the mixture velocity, where f_s and f_l refer to solid and liquid fractions, respectively.

(ii) Momentum equation

Similarly, the momentum equation is obtained for a control volume as follows.

$$\left(\begin{array}{c} \text{Mass acceleration} \\ \text{in control volume} \end{array} \right) = \left(\begin{array}{c} \text{Body forces on} \\ \text{control volume} \end{array} \right) + \left(\begin{array}{c} \text{Surface forces on} \\ \text{control volume} \end{array} \right) \quad (2.14)$$

It can be written as

(x-direction)

$$\frac{\partial}{\partial t}(\rho u) + \nabla \cdot (\rho u \vec{V}) = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + B_x + G_x \quad (2.15)$$

(y-direction)

$$\frac{\partial}{\partial t}(\rho v) + \nabla \cdot (\rho v \vec{V}) = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + B_y + G_y \quad (2.16)$$

where p can be expressed as

$$p = p_0 + \rho g(h - y) + p_d \quad (2.17)$$

Here p_d refers to the dynamic pressure component and p_0 refers to the hydrostatic pressure component. Also, h refers to free surface height; in this thesis, free surface motion will not be considered.

In Eqs.(2.15)-(2.16), the terms on the left hand side of the equations represent time rate of momentum change inside the control volume and the advection momentum flux, respectively. The terms on the right hand side of the equations refer to the pressure force, viscous forces, body forces and momentum production due to phase interactions, respectively. For example, momentum production due to phase interactions may arise due to fluid acceleration through pockets or channels in the solid dendritic matrix of the two-phase mixture. Darcy's Law will be employed for these terms since the process appears analogous to fluid flow through a porous medium. In particular, Darcy's Law is given by

$$K_x G_x = \nu_l (f_l u_r) \quad (2.18)$$

where K_x refers to x-direction permeability and $u_r = u_l - u_s$ represents the x-component relative phase velocity. The permeability coefficient decreases as f_l (liquid fraction) decreases to a limiting value of zero as the multiphase region becomes completely solid.

In adopting mixture quantities in the above momentum equations, their form appears very similar to the corresponding single phase equations. In writing the conservation equations in terms of these mixture quantities, we then obtain a standard form for the fluid flow equations (i.e. 3 equations for mixture u,v,p) rather than multiple equations for individual phase quantities. However, the source terms, G_x and G_y , include permeability effects (as discussed above), as well as convection terms (i.e. convection differences between mixture and individual phase quantities) resulting from summation of the individual phase equations.

We will assume that the fluids are Newtonian fluids, i.e. shear stress in fluid is proportional to time-rate-of-strain (velocity gradients). The Newtonian constitutive

relationships apply to the phase stress tensor, such that

$$\tau_{xx} = \lambda \nabla \cdot V + 2\mu \frac{\partial u}{\partial x} \quad (2.19)$$

$$\tau_{yy} = \lambda \nabla \cdot V + 2\mu \frac{\partial v}{\partial y} \quad (2.20)$$

$$\tau_{xy} = \mu \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) \quad (2.21)$$

where

$$\lambda = -\frac{2}{3}\mu \quad (2.22)$$

This result holds under conditions with moving monotonic gases (Stokes relation). In certain situations, such as ultrasonic waves involving more complex fluids, experiments also indicate the validity of the Stokes relation. This relation is often adopted in order that the viscous transport can be written in terms of a single viscosity coefficient.

Also, we will assume that the fluid is Boussinesq-incompressible; in other words, ρ is constant except in the body force term of the y momentum equation. The Boussinesq approximation assumes a linear dependence of density on temperature. Therefore,

$$\rho' = \rho\beta(T - T_r) \quad (2.23)$$

where ρ is the density in the ambient medium, T_r is the ambient temperature and β is the thermal expansion coefficient.

Substituting Eqs.(2.19)-(2.21) and Eq.(2.23) into Eqs.(2.15)-(2.16) and using Eq.(2.13), we obtain the following momentum equations for an incompressible fluid :

(x-direction)

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + G_x \quad (2.24)$$

(y-direction)

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + g\beta(T - T_r) + G_y \quad (2.25)$$

where we have assumed a zero solid phase velocity. The above equations apply to the liquid region in the phase change problem.

(iii) Energy equation

Consider the model of a finite control volume fixed in the region. The fundamental physical principle that energy is conserved is applied to this control volume. It means

$$\left(\begin{array}{c} \text{Rate of increase of energy} \\ \text{inside the control volume} \end{array} \right) = \left(\begin{array}{c} \text{Net flux of heat into} \\ \text{the control volume} \end{array} \right) \quad (2.26)$$

The conservation equation for energy then has the following form.

$$\frac{\partial}{\partial t}(\rho H) + \nabla \cdot (\rho \vec{V} H) = \nabla \cdot (k \nabla T) \quad (2.27)$$

where H is the phase enthalpy and ρ is density. For most solid-liquid phase change materials, the assumption may be employed that

$$\Delta H \approx \Delta e \quad (2.28)$$

where e is internal energy and it is defined as

$$e = \int_{T_0}^T c_{r,k}(\zeta) d\zeta + e_{r,k}(T) \quad (2.29)$$

where $c_{r,k}(T)$ represents the effective specific heat of phase k .

In the above equations, the specific heat is set constant in each phase region and the density is set constant throughout the domain. Problems that involve a density difference across the phase change require an additional advection term in the heat transfer equation to account for shrinkage driven flows. However, this effect will not be considered in the present work.

2.2.3 The General Conservation Equation

The foregoing equations can be considered as particular cases of the general equation

$$\frac{\partial}{\partial t}(\rho \Phi) + \nabla \cdot (\rho \vec{V} \Phi) = \nabla \cdot (\vec{J}) + \dot{S} \quad (2.30)$$

The general equation consists of four terms; they are the unsteady term, the convection term, the diffusion term and the source term, respectively.

2.2.4 Special case: Stefan Problem without fluid flow

The solidification problem without fluid flow involves heat conduction and the release (/absorption) of latent heat. Since no fluid flow arises, mass and momentum conservation do not need to be considered. The governing equation for two-dimensional, solid-liquid phase transition of a pure material, is written as follows.

$$\frac{\partial}{\partial t}(\rho e) = \nabla \cdot (k \nabla T) \quad (2.31)$$

After a supplementary equation of state has been specified, $e=e(T)$, then the energy equation may be written in terms of the temperature field alone.

2.2.5 Enthalpy Equation of State

An enthalpy equation of state is also required in order to write the energy equation in terms of temperature alone. The piecewise linear equation of state can be expressed in one equation, where the subscripts $k=1,2$ refer to the solid and liquid phases, respectively, as

$$e = e_{r,k}(T) + c_{r,k}(T)(T - T_{r,k}) \quad (2.32)$$

where the subscript r denotes a reference value.

Reference values can be obtained by integration of the Gibbs equation from a point in the liquid region to another point in the solid region. These values have been consistently set to

($k=1$)

$$e_{r,1} = 0 \quad (2.33)$$

$$T_{r,1} = 0 \quad (2.34)$$

$$c_{r,1} = \frac{c_s}{c_l} \quad (2.35)$$

(k=2)

$$e_{r,2} = \frac{\bar{c}}{c_l}(T_l - T_s) + \frac{c_s}{c_l}T_s + \frac{1}{Ste} \quad (2.36)$$

$$T_{r,2} = T_l \quad (2.37)$$

$$c_{r,2} = 1 \quad (2.38)$$

In this way, latent heat transfer occurs through the above $\frac{1}{Ste}$ term since phase change requires an energy change corresponding to the latent heat of fusion.

2.3 Boundary Conditions and Initial Conditions

The governing equations above present several different possibilities. The temperature fields and flow fields are quite different for these cases, although the governing equations are similar. This is partly because they have different boundary conditions and different initial conditions. The boundary conditions and the initial conditions dictate the particular solutions to be obtained from the governing equations. So, the formulation of a problem requires appropriate boundary and initial conditions.

Spatial boundary conditions in direct heat transfer problems are of three general types. They may be stated in a simplified mathematical form as follows.

1. Dirichlet Boundary condition: specified value of surface temperature. i.e.

$$\Phi = f_1 \quad (2.39)$$

where Φ =temperature and f_1 = its specified value.

2. Neumann Boundary condition: specified value of outward normal heat flux or temperature gradient. i.e.

$$\frac{\partial \Phi}{\partial n} = f_2 \quad (2.40)$$

3. Robin boundary condition: convection condition, i.e.

$$A \frac{\partial \Phi}{\partial n} + B\Phi = f_3 \quad (2.41)$$

For the case of heat transfer with fluid flow, the boundary conditions on the flow variables are expressed in terms of fluid velocities. Zero velocity conditions are prescribed along all solid boundaries. Other possibilities include Neumann (zero flux) conditions along a line of symmetry, or prescribed values at a flow inlet.

For the inverse problem, with the exception of the boundary where the temperature needs to be controlled, all boundary conditions can be treated as in a direct problem (Dirichlet, Neumann and Robin boundary conditions). At the controlling boundary, the temperatures are unknown and they represent the solutions of the problem. The temperature at this boundary is obtained by iterations following an initial estimate; in other words, at the beginning of each time step, an estimated temperature is initially given. Usually, it is set equal to the temperature at the last time step. During each time step in the solution, the controlling boundary temperature (i.e. $x=0$) is updated at each iteration until the predicted movement of the interface agrees, within a given tolerance, with the specified movement. The update used in this work is given by:

$$T_0^{(m+1)} = T_0^m + \frac{T_p^{m+1} - T_p^m}{R_p} \quad (2.42)$$

where m is the iterative counter, the subscript p refers to nodal point p and R is the sensitivity coefficient (discussed in next chapter).

Since the problem is time dependent (transient), i.e. the variables change with time, the initial condition should give the distribution of the temperature and the velocities over the entire region at the initial time, $t = 0$.

Chapter 3

Numerical Model

Finite Element Method and Finite Difference (Volume) Methods are the most common numerical methods for heat transfer analysis. The principal advantages of the finite element approach are the ability to handle irregular geometries and treatment of difficult boundary conditions. Also, the overall flexibility of the approach in terms of changing geometries or physical properties, without altering the solution algorithm, is a major benefit. On the other hand, finite element codes for heat transfer and fluid flow are still far from optimal and are currently undergoing further development. They may be less efficient or less easy to code initially in comparison to finite difference methods.

A finite element method is a mathematical procedure for solving a partial differential equation in an integral sense across a finite element. It requires that an integral representation of a partial differential equation is constructed. The solution of a physical problem by a finite element method follows a well-defined sequential process. Firstly, the physical region is discretized into elements. The number, type, and allocation of elements are often a matter of judgment. Secondly, interpolation or shape functions are selected for the elements. The interpolation functions represent the assumed form of the spatial solution in the elements and are related to the number of nodes in the elements. Thirdly, the matrix equations for an individual element are formulated using the integral statement for the element. Fourthly, the matrix equations for the overall system, consisting of all the elements, are assembled. The global equations have the same form as the element equations but they have a larger dimension. Finally, the

global equations are solved. Post-processing of results can also be pursued.

Some aspects of the finite element program discussed in this chapter (i.e. fluid flow algorithm) have been developed elsewhere [25] in the context of direct problem. The contributions of this thesis appear through the extensions to inverse problems. As a result, the entire formulation is described as it relates to implementation of the inverse procedure.

3.1 Discretization of Solution Domain

The first step in a finite element approach is to subdivide the volume into a set of subvolumes while nodes are distributed throughout the domain. A two-dimensional domain and linear, four-noded quadrilateral elements are used. A typical element is shown in Fig. 3.1 with the local non-orthogonal coordinate system. These coordinates are denoted by s and t . The range of both coordinates is from -1 to 1, with the nodes numbered from 1 to 4. Figure 3.2 shows the control volume which is divided into four sub-control-volumes (SCVs). The boundaries of sub-control-volumes are called sub-surfaces (SS). An integration point (ip) is defined as the midpoint of each sub-surface. Shape functions are employed to relate the local and global coordinates.

A general variable, Φ , global coordinates, and derivatives of the general variable Φ are expressed in the following form.

$$\Phi(s, t) = \sum_{k=1}^4 N_k(s, t)\Phi_k \quad (3.1)$$

$$x(s, t) = \sum_{k=1}^4 N_k(s, t)x_k \quad (3.2)$$

$$y(s, t) = \sum_{k=1}^4 N_k(s, t)y_k \quad (3.3)$$

$$\frac{\partial \Phi}{\partial x} = \sum_{k=1}^4 \frac{\partial N_k}{\partial x} \Phi_k \quad (3.4)$$

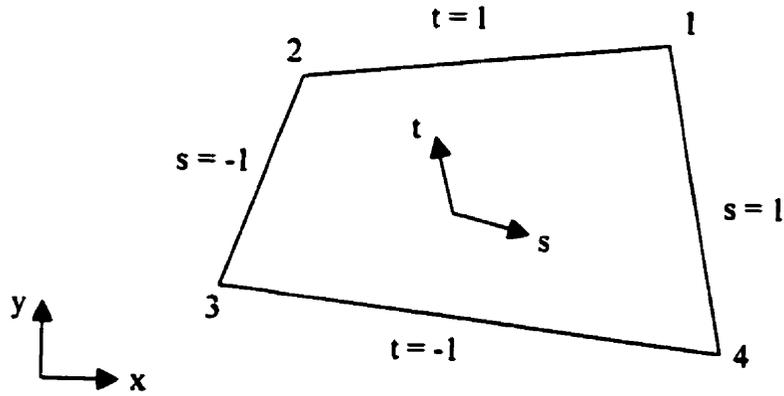


Figure 3.1: A single element

$$\frac{\partial \Phi}{\partial y} = \sum_{k=1}^4 \frac{\partial N_i}{\partial y} \Phi_i \quad (3.5)$$

where finite element shape functions N_i are used to relate global Cartesian coordinates and scalar values, Φ , to local element values in a bilinear fashion. They represent the local shape functions N_i ($i=1,2,3,4$) for the linear quadrilateral elements at the local coordinate (s,t) .

$$N_1(s, t) = \frac{1}{4}(1 + s)(1 + t) \quad (3.6)$$

$$N_2(s, t) = \frac{1}{4}(1 - s)(1 + t) \quad (3.7)$$

$$N_3(s, t) = \frac{1}{4}(1 - s)(1 - t) \quad (3.8)$$

$$N_4(s, t) = \frac{1}{4}(1 + s)(1 - t) \quad (3.9)$$

However, the x and y derivatives of the shape functions are not yet known. They are determined using the chain rule with the following results.

$$\begin{bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{bmatrix} = \frac{1}{J} \begin{bmatrix} \frac{\partial y}{\partial t} & -\frac{\partial y}{\partial s} \\ -\frac{\partial x}{\partial t} & \frac{\partial x}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial N_i}{\partial s} \\ \frac{\partial N_i}{\partial t} \end{bmatrix} \quad (3.10)$$

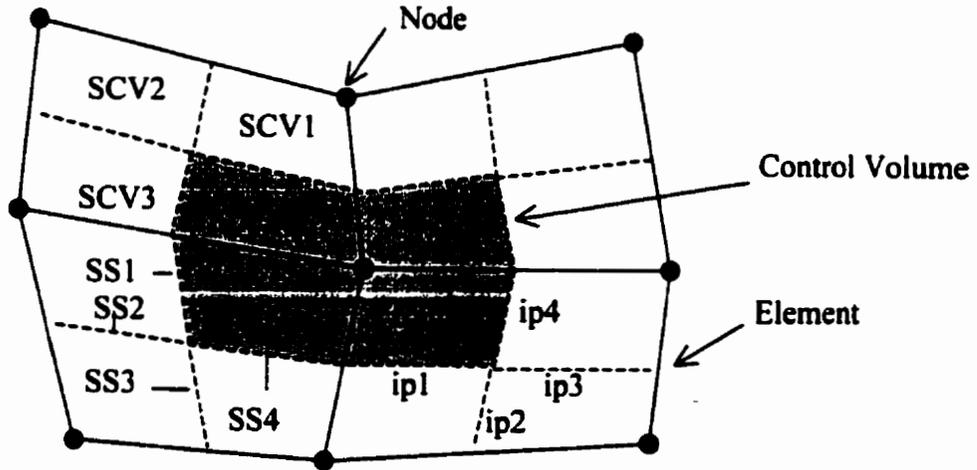


Figure 3.2: Control volume

where

$$J = \frac{\partial x}{\partial s} \frac{\partial y}{\partial t} - \frac{\partial y}{\partial s} \frac{\partial x}{\partial t} \quad (3.11)$$

is the Jacobian determinant. Also,

$$\frac{\partial x}{\partial s} = \sum_{k=1}^4 \frac{\partial N_k}{\partial s} x_k \quad (3.12)$$

$$\frac{\partial x}{\partial t} = \sum_{k=1}^4 \frac{\partial N_k}{\partial t} x_k \quad (3.13)$$

$$\frac{\partial y}{\partial s} = \sum_{k=1}^4 \frac{\partial N_k}{\partial s} y_k \quad (3.14)$$

$$\frac{\partial y}{\partial t} = \sum_{k=1}^4 \frac{\partial N_k}{\partial t} y_k \quad (3.15)$$

These geometric quantities are computed at an element level at each time step.

3.2 Discretization of Governing Equations

The discrete control volume equations can be derived by integration of the continuum equations, Eqs. (2.13), (2.24), (2.25) and (2.27) over finite control volumes and time intervals. This includes a net surface flow balance with source and temporal storage volume integrals. For SCV1 within the shaded control volume (Fig. 3.2), Eqs. (2.13), (2.24), (2.25) and (2.27) are integrated to yield the following integral conservation laws,

$$\frac{\partial}{\partial t} \int_V \rho \Phi(t) dV + \int_S \vec{F}^a \cdot d\vec{n} + \int_S \vec{F}^d \cdot d\vec{n} = \int_V S_\Phi dV \quad (3.16)$$

The vector \vec{F}^a denotes advection flux and \vec{F}^d denotes diffusion flux. Also, S_Φ represents the source term. Integration over a time interval $\Delta t = t^{n+1} - t^n$ yields the equation

$$\left(\frac{\rho J_1}{\Delta t} \right) \bar{\Phi}^{n+1} = \left(\frac{\rho J_1}{\Delta t} \right) \bar{\Phi}^n + \sum_{i=1}^4 \int_{S_i} \vec{F}^a \cdot d\vec{n} |^{n+1} + \sum_{i=1}^4 \int_{S_i} \vec{F}^d \cdot d\vec{n} |^{n+1} + \bar{S}_\Phi \quad (3.17)$$

where J_1 is the SCV1 area. The control volume (node n) equation will be completed when each element SCV contribution is assembled into the global system of discrete equations.

3.2.1 Conservation of Mass, Momentum Equations

The algebraic representations of the integration point equations are required to form the integration point equations. The current approach is based on the co-located algorithm developed by Schneider and Raw [26]. The algebraic approximations of the integration point values are given as follows.

(i) Convection Operator

The convection operator at an integration point is represented by an upstream difference,

$$C(\Phi) = \rho V \frac{\partial \Phi}{\partial s} \approx \rho V \left[\frac{\Phi_{ip_i} - \Phi_u}{L_c} \right] \quad (3.18)$$

where V represents the fluid velocity magnitude,

$$V = \sqrt{u^2 + v^2} \quad (3.19)$$

and s is the local streamwise direction

$$ds = \frac{u}{V}dx + \frac{v}{V}dy \quad (3.20)$$

Also, L_c is the convection length scale and Φ_u represents the upwind value of Φ . The upwind values, Φ_u were determined by two different models as described in ref. [25].

(ii) Diffusion Operator

The diffusion operator in all transport equations is represented differentially by the Laplacian, $L(\Phi) = \nabla^2(\Phi)$. It is approximated by a central difference,

$$\nabla^2\Phi|_{ip} \approx \frac{\sum_{j=1}^4 N_j\Phi_j - \Phi_i}{L_d^2} \quad (3.21)$$

where L_d is a diffusion length scale. It was approximated by extensions from one-dimensional results. In particular,

$$L_d^2 = \frac{\Delta x^2}{2} + \frac{3\Delta y^2}{8} \quad (3.22)$$

where Δx and Δy refer to length scales perpendicular and tangential, respectively, to the sub-surface in question.

(iii) Pressure Gradient Operator

An approximation to the local pressure gradient in the x and y directions was also required. Shape functions were employed in this approximation,

$$\frac{\partial p}{\partial x}|_{ip} = \sum_{i=1}^4 \frac{\partial N_i}{\partial x} P_i \quad (3.23)$$

where the upper-case convention represents nodal variable dependence. The pressure gradient in the y direction is constructed in a similar fashion.

(iv) Transient operator

A transient term at an integration point is approximated by a backward-difference,

$$\left. \frac{\partial \Phi}{\partial t} \right|_{ip} \approx \frac{\Phi_{ip}^{n+1} - \Phi_{ip}^n}{\Delta t} \quad (3.24)$$

where the superscripts $n+1$, n denote current and previous time steps, respectively.

Now, we can add the various operators to form the complete integration point equation approximation. These integration point equations, at element level, can be inverted to determine the integration point variables in terms of the nodal point variables. The above integration point operators are substituted into the momentum transport equation, i.e.

$$\frac{\partial \hat{v}}{\partial t} + C(\hat{v}) = -\nabla p + L(\hat{v}) + \bar{B} + \bar{S} \quad (3.25)$$

Assembling each of the component operators in Eq. (3.25) and inverting the 4 element integration point equations for \hat{v}_{ip} explicitly in terms of the nodal values,

$$\{\hat{u}\} = [IC^{uu}]\{U\} + [IC^{up}]\{P\} + \{RS^u\} \quad (3.26)$$

and similarly for the y-momentum equation,

$$\{\hat{v}\} = [IC^{vv}]\{V\} + [IC^{vp}]\{P\} + \{RS^v\} \quad (3.27)$$

where the IC matrices represent Influence Coefficient matrices and $\{RS\}$ refers to the right side source vector.

The conservation of mass equation can then be obtained from Eq.(3.17) with $\Phi(\vec{x}, t) = 1$, $\vec{F}^d = 0$ and $\vec{F}^a = \hat{v}$. For the conservation of momentum equation, $\Phi(\vec{x}, t) = \vec{v}(\vec{x}, t)$ and the mean mixture diffusion, advection and source term components of Eq. (3.17) are modelled by

$$\vec{F}_j^d |_{ip} = (Pr) \sum_{k=1}^2 \nu_k \sum_{i=1}^4 U_{j,i} \quad (3.28)$$

$$\bar{F}_j^a |_{ip} = \rho \hat{v}^n \sum_{i=1}^4 (IC)_i^U U_{j,i} + \sum_{i=1}^4 \delta_{i,j} P_i \quad (3.29)$$

$$S_{U_j} = \hat{u}_j^n \hat{v}^n |_{ip} - \sum_{i=1}^4 \hat{v}_{k,i}^n \hat{u}_{k,j}^n + (Pr Ra_T) \frac{g_j}{g} (T - T_r) \quad (3.30)$$

where the subscripts i, j, k denote local nodes, coordinate directions and phases, respectively.

The entire fluid flow system is a pressure-velocity coupled system of equations whose solution consists of the three variables u, v and p at each nodal point. Meanwhile, there are three equations (mass equation and two momentum equations). This means that the number of equations is equal to the number of variables. But the mass equation, apparently having no direct link to pressure, but rather an indirect link through the integration point equations, is an additional constraint on the velocity field. The mass equation is considered to be the main equation for pressure and the momentum equations provide further equations for the velocity components.

The temperature-velocity coupling appears through the buoyancy terms in S_{U_j} . Although the pressure-velocity coupling was resolved simultaneously in these equations, the velocity-temperature coupling remains segregated. In other words, the energy equation obtains temperature separately, and inter-equation iterations (momentum-energy equations), rather than a simultaneous coupling, are performed until acceptable convergence is achieved.

3.2.2 Conservation of Energy Equation

The conservation of energy equation can be obtained by substituting $\Phi(\vec{x}, t) = e(\vec{x}, t)$ in Eq. (3.17). The diffusion, advection and source terms are approximated by

$$\bar{F}^d(T_i) |_{ip} = k \sum_{i=1}^4 \frac{\partial N_i}{\partial x_j} T_i \quad (3.31)$$

$$\bar{F}^a(T_i) |_{ip} = \hat{v} \sum_{i=1}^4 (IC)_i^E e_i(T_i) \quad (3.32)$$

$$S_T^{e,a}(T_{r,i}) = -\hat{v}_n \sum_{i=1}^4 (IC)_i^T [e_{r,i} - c_{r,i} T_{r,i}] + \sum_{k=1}^2 \sum_{i=1}^4 [\hat{v} - \hat{v}_k] (IC)_i^T e_{k,i}^n(T_i) \quad (3.33)$$

A Crank-Nicolson scheme is used to evaluate diffusion terms and sources at the intermediate time level. In particular, this intermediate time level refers to the midpoint between the current time level and previous time level. For example, if we let the superscripts n+1 and n refer to current and previous step, then the diffusion term in Eq. (3.31) is evaluated as

$$\vec{F}^d(T_i)|_{ip} = k \sum_{i=1}^4 \frac{\partial N_i}{\partial x_j} \left(\frac{T_i^{n+1} + T_i^n}{2} \right) \quad (3.34)$$

The coefficients at step n+1 remain as active coefficients in the implicit solution whereas terms at step n are grouped together in the source term. The time marching procedure then becomes second order accurate and algorithm essentially second order accurate in space.

3.2.3 Boundary Conditions

At a solid wall, all velocity components are specified as zero. The momentum equations for the elements at solid wall are replaced by these equations (all velocity components are zero) in the global system. Furthermore, boundary conditions for the energy conservation equation may be specified for Dirichlet, Neumann or Robin conditions. The specific types of conditions are problem dependent and in the example problems (next chapter). we will identify these specific conditions.

For an inverse problem, the temperature at the boundary where the temperature is required and predicted is initially unknown. Thus, it is considered a problem variable instead of a boundary condition. It is obtained through an iterative procedure.

3.3 Sensitivity Coefficient

The sensitivity coefficient will be defined as follows,

$$R_p = \frac{\partial T_p}{\partial T_0} \quad (3.35)$$

where the subscript 0 refers to boundary and p refers to nodal position.

This coefficient essentially measures the influence of changes in the boundary temperature T_0 on the temperature at the point P (inside domain). The range of sensitivity coefficient R is between 0 and 1. The value of R expresses the temperature connection between the point P and the boundary. As the sensitivity coefficient, R, becomes larger, then the influence of the changes at the boundary temperature to the point P becomes stronger. It is apparent that the closer the point lies relative to the boundary, then the bigger the sensitivity coefficient becomes. This coefficient is used to update the boundary temperature in the current inverse method.

Furthermore, if we write T_p^{m+1} in terms of a Taylor series expansion, then it can be expressed as

$$T_p^{m+1} = T_p^m + \frac{\partial T_p}{\partial T_0}(T_0^{m+1} - T_0^m) + \frac{1}{2!} \frac{\partial^2 T_p}{\partial T_0^2}(T_0^{m+1} - T_0^m)^2 + \dots \quad (3.36)$$

We will neglect the higher order terms and only the first 2 terms are retained. If we rearrange Eq.(3.36), then it becomes the same as the boundary temperature equation described earlier in Eq.(2.42).

Now, we will consider how to perform the numerical calculation of the sensitivity coefficient. Substituting Eq.(2.32) into Eq.(2.31), we find that the heat conduction equation becomes an equation in terms of temperature alone. Then, taking derivatives with respect to T_0 on both sides of the equation, the terms in Eq.(3.31) and Eq.(3.33) will be taken (derivatives, one by one). The first term on the right hand side of Eq.(3.33) is not a function of T_0 (i.e. zero derivative). Then, we obtain a resulting equation in terms of the sensitivity coefficient, R. The resulting equations can be written in the following form:

$$\left(\frac{\rho J_1}{\Delta t}\right) R^{n+1} = \left(\frac{\rho J_1}{\Delta t}\right) R^n + \sum_{i=1}^4 \int_{S_i} \bar{F}^d \cdot d\bar{n}|^{n+1} + \bar{S}_\Phi \quad (3.37)$$

where J_1 refers to sub-control volume area. Also,

$$\bar{F}^d(R_i)|_{ip} = k \sum_{i=1}^4 \frac{\partial N_i}{\partial x_j} R_i \quad (3.38)$$

$$S_T^{e,n} = \sum_{k=1}^2 \sum_{i=1}^4 [\hat{v} - \hat{v}_k] (IC)_i^T e_{k,i}^n(R_i) \quad (3.39)$$

Furthermore, we have the following boundary condition where

$$R_0^k = 1 \quad (k > 0) \quad (3.40)$$

and the initial condition

$$R_j^0 = 0 \quad \text{all } j \quad (3.41)$$

where the subscript denotes node point and the superscript denotes the time level. Through a similar procedure as in the solution of the direct problem, the resulting equation can be solved numerically and the solution R can be obtained.

In Eq. (3.37), R is a matrix and every node point has a corresponding R value. It can be solved numerically in the same manner as a direct problem. The boundary conditions and initial conditions are known (Eq.(3.40) and Eq.(3.41)). By solving Eq.(3.37), we obtain the sensitivity coefficients for every node point. While solving this equation, iterations are needed. When the difference between values at two iterations is smaller than the specified tolerance, then, the calculation is terminated.

When we update the boundary condition during the iterations in the inverse method, only the sensitivity coefficients on the interface points are used. Since the interface is moving at each time step, the sensitivity coefficient at a different location is adopted. When the controlling boundary is parallel with the y -axis, the sensitivity coefficients vary only with x . For two dimensional solidification problems, the interface profile is not a vertical line and the sensitivity coefficients on the interface have different values. In this case, each point would use its own corresponding value.

In the inverse problem, the governing equations are the same as in direct problem. The difference is that the interface position is given and the temperatures at

the controlling boundary are unknown. The other boundary conditions are given as in the direct problem. During each time step, initially, a guessed temperature at the controlling boundary is used and the equations are solved as a direct problem. The temperatures at the controlling boundary are updated continuously through iterations until the predicted movement of the interface comes within a given tolerance of the desired path. Then the solution is terminated.

In the inverse method, the interface is moving; it moves across one grid spacing at each time step during the calculation. Given the movement of the interface (i.e. prescribed velocity of interface), we then select a constant time step. During the calculations, in order to ensure that at any time step the interface moves from one node point to the next adjacent point, a fixed numerical grid is specified. The grid can be uniform or non-uniform according to the different interface velocities in a particular problem.

3.4 Entropy Based Method for Numerical Stability

Due to the diffusive nature of heat flow, the inverse heat transfer problem is ill-posed in the sense that arbitrarily small errors in the temperature measurements or interface position can lead to arbitrarily large errors in the estimated values of the boundary temperature. Numerical simulations have shown that numerical instability often occurs in the inverse heat transfer problem. During inverse calculations, when the interface moves farther and farther from the boundary, it becomes more and more difficult to control the interface by adjusting the boundary temperature. Then, numerical oscillations are likely to occur. In order to enhance solution performance and improve the stability of the algorithm, an entropy based method will be proposed here.

The scheme currently used in the numerical solution of heat and fluid flow is conservative. This means that it conserves mass, momentum and energy in any control volume. The First Law of Thermodynamics is satisfied for each control volume. However, the Second Law has not received the same treatment. This is unfortunate because,

without an entropy condition, the equations have many solutions, some of which are reasonable and smooth, whereas other possible results may seem non-physical (i.e. oscillations). Furthermore, it has been shown that satisfaction of a cell entropy inequality is sufficient, in some cases, to guarantee non-linear stability in numerical computations.

A few research studies have investigated the Second Law of Thermodynamics for numerical stability in solidification problems. Naterer [27, 28] used an entropy based method successfully in direct heat transfer problems. Entropy may become a mechanism for providing convergence enhancement in phase change computations.

3.4.1 The Second Law and Generalized Entropy

Although the First Law of Thermodynamics allows for conversion from one form of energy to another, as long as the overall quantity is conserved, experimental evidence indicates that in certain types of energy conversion, a restriction must be placed in terms of the direction and extent of transformation. For example, it is a fact that although work can be completely converted into heat, heat cannot be completely converted into work in a continuous manner. It is the Second Law of Thermodynamics that imposes restrictions on the direction and extent of energy transformation processes.

The Second Law is a fundamental law of nature. It can be stated in many different forms, such as the Clausius statement of the Second Law, as follows: it is impossible for any device to operate in a cycle in such a manner that the sole effect is the transfer of heat from one body to another body at a higher temperature. In the case of a heat pump, or refrigerator, heat does flow from a region of low temperature to high temperature, but only when work is added to the machine from an outside source. The Second Law of Thermodynamics, through the property called entropy, permits a quantitative evaluation of thermodynamic systems and predicts the directional nature of physical processes. It provides information that complements an analysis based on the conservation of mass and energy.

Also, recent research on computational interpretations of entropy and entropy pro-

duction has shown that entropy is related to discretization errors, artificial dissipation and non-physical numerical results. This means that in addition to physical mechanisms producing entropy, we find that computational procedures can also produce entropy. We can refer to “generalized entropy” as the combined computational and physical entropy. Then, the Second Law of Thermodynamics in a numerical scheme can be expressed in terms of “generalized” (computational plus physical) entropy as follows,

$$\dot{P}_s \equiv \frac{\partial S_k}{\partial t} + \nabla \cdot F_k \geq 0 \quad (3.42)$$

where \dot{P}_s refers to the entropy production rate and S_k and F_k represent the thermodynamic entropy and entropy flux in phase k, respectively. The equality refers to reversible processes whereas the inequality refers to irreversible processes. We will use the following variables,

$$S_k = \rho_k s_k \quad (3.43)$$

$$F_k = \rho_k \vec{V}_k s_k - \vec{q}/T \quad (3.44)$$

In these equations, s_k represents the specific entropy in phase k,

$$s_k = s_{r,k} + c_v \log \left[\frac{T}{T_r} \right] \quad (3.45)$$

where the additional subscript r refers to values at a reference state.

In a similar way as the construction of the enthalpy equation of state, we must include an entropy of fusion term in the above reference entropy in the liquid phase. Since entropy is calculated after the solution of the conservation equations is obtained, then iterations will not be required in this case. The elements are re-assembled for the computation of volumetric entropy production rate following the solution of the conservation equations. This assembly includes the transient entropy term (using entropy equation of state and backward difference in time), advective term (using pressure-weighted upwinding) and diffusive term (using bilinear interpolation with shape functions). At the boundaries, a positive-definite expression for entropy production (see below), rather than completion of the transport equation, is given as the boundary

condition.

Another expression for the entropy production rate involving heat conduction is given as follows[27],

$$\dot{P}_s = \frac{k \nabla T \cdot \nabla T}{T^2} \quad (3.46)$$

where k denotes the thermal conductivity. The temperature is measured on an absolute temperature scale. Since $\nabla T \cdot \nabla T$, k and T^2 are all greater than or equal to zero, then,

$$\dot{P}_s \geq 0 \quad (3.47)$$

This result is the Second Law of Thermodynamics. It applies to all physical processes such as heat conduction. However, non-physical behaviour in the numerical solution, arising from discretization errors (i.e. coarse grid refinement, lack of solution convergence, large time steps) may violate the Second Law in a discrete control volume.

The difference between the two forms of the expressions for \dot{P}_s is that in Eq. (3.46), the local entropy production rate will be greater than or equal to zero, both analytically and numerically. On the other hand, Eq. (3.42) is a transport equation and non-physical numerical results may lead to negative entropy production rates in the numerical formulation.

3.4.2 Improving Stability of Inverse Method

The principle of increasing entropy during an irreversible process provides a means for analyzing processes on the basis of the Second Law of Thermodynamics. The processes that result in a total positive entropy production are possible according to the Second Law. Any processes that suggest a decrease in the total entropy for an isolated system are impossible, and only totally reversible processes can produce a zero change in the total entropy. Numerical studies suggest that satisfaction of the Second Law is important because a sufficient condition for stability of a scheme can be given in terms of a positive entropy production rate in the computations, i.e. non-physical solutions are accompanied by a negative entropy production. To exclude non-physical solutions

and to seek a physically relevant solution, additional modifications may be required in the numerical formulation. In the current model, this modification will be based on the Second Law of Thermodynamics, which states that the entropy production must remain positive. The method will be called an entropy based method.

For the inverse problem with solidification from the left boundary, experience has shown that as time increases, the interface moves further away from the left boundary and the effect of the boundary temperature on the interface movement becomes weaker and weaker. The thermal information cannot be effectively carried from the boundary to the interface. As a result, it will be more and more difficult to control the interface movement by the temperature at the left boundary. Then, oscillations in the inverse solution may start to occur. Also, from the calculation of the sensitivity coefficient, it can be seen that as the interface moves further and further away from the left boundary, the sensitivity coefficients become smaller and smaller. When the values of the sensitivity coefficient are very small, the resulting roundoff error may lead to numerical instability. Meanwhile, from the updating equation for the boundary temperature, Eq.(2.42), during the iterations, it can be shown that when the sensitivity coefficient becomes very small, the updated boundary temperature will exhibit big changes from one iteration to the next iteration. The solution will become unstable as the iterative values change drastically.

By calculating the entropy production rate, numerical studies in the present work have shown that the entropy production rate becomes negative when the simulation becomes unstable. The scheme may violate the Second Law of Thermodynamics. Therefore, the algorithm will be modified in order to stabilize the calculations. In the current formulation, the entropy production will be employed as a criterion, i.e. if the value of entropy production becomes negative, it suggests that an instability occurs and thus a correction of the solution needs to be used. The correcting procedure should only be applied when the non-physical solution behavior arises. In this way, we reduce computational time and avoid corrections of the solution at every time step.

During the simulation, the entropy production rate is computed based on Eq.(3.42) following each time step. If the nodal value of \dot{P}_s is negative, then the local solution is not physically correct. Therefore, instead of proceeding to the next time step, a correction is performed based on the magnitude of entropy production within the control volume.

From Eq.(3.46), the conductivity can be expressed in terms of entropy production \dot{P}_s as follows.

$$k = \frac{T^2 |\dot{P}_s|}{\nabla T \cdot \nabla T} \quad (3.48)$$

Thus, k is related to the entropy production rate and the local temperature gradient.

In order to calculate the above expression, we need to estimate the value of temperature gradient corresponding to the related entropy production rate. Recall that \dot{P}_s was obtained for a specific control volume and it was based on the transient entropy term and the net entropy flux across the boundaries of the control volume. As a result, it was comprised of a volumetric summation of terms in sub-control-volumes corresponding to a given node in the mesh. On the other hand, the temperature gradient in Eq. (3.48) is a pointwise value based on the temperature distribution obtained from the solution of the energy equation. In order to match representations, the area-weighted value of temperature gradient is also calculated and assembled with the calculation of \dot{P}_s such that an overall or characteristic ∇T is employed for the volumetric value in the calculation of k . This approximation becomes more accurate as the grid is refined.

If the Second Law is violated locally, then the conductivity can be computed in Eq.(3.48) using the entropy production \dot{P}_s from Eq.(3.42). Then this entropy based conductivity is used to calculate the sensitivity coefficients again and modified sensitivity coefficients are obtained. These new sensitivity coefficients can be employed to update the boundary temperature during the iteration. It will be shown that this method prevents potentially non-physical solution behaviour in numerical calculations in inverse problems.

3.5 Solution Procedure

The numerical procedure for the inverse solution can now be summarized as follows.

1. Specify the movement of the phase interface (i.e. interface velocity) and choose a constant time step based on this interface velocity.
2. Based on the velocity of the interface and the chosen time step, then a fixed numerical grid is specified such that at any time step the interface moves from one grid point to the next grid point. If a variable interface velocity is specified, then either a variable time step or non-uniform grid is selected. In this work, the latter approach is adopted.
3. Within each time step, an estimate of the unknown controlling boundary temperature is given and then the energy conservation equation is solved in a direct manner for the temperature, T .
4. The sensitivity coefficients are obtained and the unknown (controlling) boundary temperature is updated. Then, the energy equation is solved again. An entropy based correction of conductivity is provided for the sensitivity coefficients whenever the Second Law is locally violated.
5. Repeat steps 3-4 for each iteration in solving the energy conservation equation. The solution is terminated when the predicted movement of the interface agrees, to a given tolerance, with the specified (desired) interface movement.
6. Assemble the internal mass-momentum conservation equations. The current temperature solution is used for construction of buoyancy terms in the y-momentum equation.
7. Apply the velocity and pressure boundary conditions.

8. Solve the simultaneous mass-momentum equation set for the fluid velocity, \vec{v} , and pressure, p , in the liquid region. The velocity in the solid phase is taken as zero.

9. Repeat steps 6-8 for iterations of the mass-momentum conservation equations until solution convergence involving the $p - \vec{v}$ coupling is obtained.

10. Return to item 3 for solution convergence in the $T-\vec{v}$ coupling. After the inter-equation residuals decrease below a specified small tolerance, the calculation is terminated for this time step. Then we begin the next time step.

For the inverse problem without the effects of fluid flow, the numerical procedure follows step 1 to step 5. If the effects of fluid flow are considered, the numerical procedure follows step 1 to step 10.

Chapter 4

Applications and Results

In this chapter, numerical results are presented to indicate some general features, performance and accuracy of the proposed methodology. The goal of the study involves determining the boundary temperature, which produces a prescribed moving phase interface shape and motion. The following problems are considered: (1) Stefan problem with variable interface velocity, (2) Stefan problem with constant interface velocity, (3) two-dimensional solidification process involving pure aluminum, (4) solidification process with pure gallium and the effects of fluid flow. The finite element computer code was verified through comparisons with results obtained from a direct problem and by comparisons with available exact solutions. Calculations were performed with a Pentium 233 MHz processor and the solution at each time step generally required only a few seconds.

4.1 Stefan Problem with Variable Interface Velocity

One of the classical direct problems reported by Carslaw and Jaeger[29] is the solidification of a pure material initially at the phase-change temperature ($T_0 = 0$) in a semi-infinite region. All parameters in this example are reported in dimensionless form. The temperature at the surface $x=0$ (left boundary) is maintained at the temperature $T(0, t) = -1$. The solution of this direct problem describes how the position of the phase interface moves with time, i.e.

$$x = 0.314\sqrt{t} \tag{4.1}$$

where the interface velocity, V , is variable, with

$$V = \frac{0.157}{\sqrt{t}} \quad (4.2)$$

Let us now reverse this direct problem and use it as an example of an inverse problem to verify the performance of the present method. We have a one dimensional problem. However, the computer program is written for two-dimensional problems. So, for this example, the variables in the y direction will be uniform. Given the interface position, Eq.(4.1), and the variable interface velocity, Eq.(4.2), the phase change will be controlled by conduction in the solid behind the phase interface. We require the estimation of the boundary temperature, $T(0, t)$, at $x=0$. The following problem parameters are adopted,

$$\begin{aligned} \alpha &= 1 \\ c &= 1 \\ Ste &= 0.05 \end{aligned}$$

where α , c and Ste are thermal diffusivity, specific heat and Stefan number in dimensionless form, respectively. Using a constant time step in the numerical solution, it is determined that

$$\Delta t = \frac{1}{0.0986n}$$

where n is the number of nodal points in the grid. This time step refers to a dimensionless time step (i.e. Fourier number increment).

In order to ensure that the interface moves one grid point in one time step in the x direction, a non-uniform grid in the x direction is used. The grid in the y direction is uniform. The x grid coordinate is expressed as follows.

$$x = \sqrt{\frac{P}{n}}$$

where P refers to node point.

time step	Boundary Temperature
1	-1.0311662
2	-0.9931377
3	-1.0247968
4	-0.9912896
5	-1.0215482
6	-0.9911610
7	-1.0192622
8	-0.9916910
9	-1.0173930
10	-0.9925229
11	-1.0157434
12	-0.9935080
13	-1.0142196
14	-1.0142196
15	-0.9945795
16	-1.0127866
17	-0.9956929
18	-1.0114311

Table 4.1: Boundary temperature at $x=0$ for Stefan problem with variable interface velocity (with 40×5 element mesh)

The analytical solution for this problem shows that the boundary temperature remains constant, i.e.

$$T(0, t) = -1.0$$

The results from the current inverse algorithm are shown in Table 4.1. It appears that the results are in good agreement with the analytical solution. This one-dimensional problem is solved in a two-dimensional domain with a 5×40 element mesh (Table 4.1). In the heat flow direction (x direction), 40 elements are employed, whereas 5 elements are sufficient in the y direction since the problem is one-dimensional. The temperature predictions are within the range -0.9911610 to -1.0311662. These results indicate that the algorithm remains stable and it exhibits good accuracy.

Since a specified interface velocity is used, the problem remains identical upon refinement of the grid and refinement of the time step. Considering different refinements of the grid, the results for a 5×20 and 5×10 element mesh are shown in table 4.2-4.3.

time step	Boundary Temperature
1	-1.0314784
2	-0.9907735
3	-1.0250678
4	-0.9891495
5	-1.0219077
6	-0.9892465
7	-1.0196872
8	-0.9899748
9	-1.0178328
10	-0.9910083
11	-1.0161370
12	-0.9922191
13	-1.0145171
14	-0.9935507
15	-1.0129205
16	-0.9949734

Table 4.2: Boundary temperature at $x=0$ for Stefan problem with variable interface velocity (with 20×5 element mesh)

time step	Boundary Temperature
1	-1.0626889
2	-0.9956866
3	-1.0336510
4	-0.9864186
5	-1.0280416
6	-0.9844545
7	-1.0255967
8	-0.9840683

Table 4.3: Boundary temperature at $x=0$ for Stefan problem with variable interface velocity (with 10×5 element mesh)

The temperature predictions are now within the range -0.9891495 to -1.0314784 and -0.9840683 to -1.0626889. Thus, improved accuracy is obtained for the refined grid. A balance between the refinement of the grid and the length of computing time should be considered since more grid points are taken in the calculations, but more calculations are needed and the computing time will be longer.

4.2 Stefan Problem with Constant Interface Velocity

The problem involves the solidification of a pure material initially at the phase-change temperature ($T_0 = 0$) as described in section 2.1.1. It requires the estimation of the boundary temperature, $T(0, t)$, at $x=0$, such that this boundary temperature provides a constant phase interface velocity, V . Since the top, bottom and right boundaries of the cavity are insulated, no temperature gradients will occur in the liquid region and the phase change will be essentially controlled by conduction in the solid. In physical terms, this case represents solidification of a liquid initially at the melting temperature, which is cooled at the left boundary ($x=0$). Solidification starts at $x=0$ and proceeds rightward. The liquid portion of the mold remains at the melting temperature during the process. This problem was chosen as a test problem because the exact solution is available in the technical literature[29]. The following problem parameters (dimensionless) are adopted here,

$$\begin{aligned}\alpha &= 1 \\ c &= 1 \\ Ste &= 2 \\ V &= 2.0\end{aligned}$$

where Ste , α and c are Stefan number, thermal diffusivity and specific heat, respectively.

The exact analytic solution for this situation involving a constant-velocity interface motion, $V = \text{constant}$, was first obtained by Stefan and later reported by Carslaw and Jaeger [29] as

$$T(x, t) = T_0 + \frac{L}{c} \left[1 - \exp\left(\frac{V^2}{\alpha}t - \frac{V}{\alpha}x\right) \right], x \leq Vt \quad (4.3)$$

and

$$T(x, t) = T_0, x \geq Vt \quad (4.4)$$

At $x = 0$, the surface temperature required to produce the prescribed motion becomes

$$T(0, t) = T_0 + \frac{L}{c} \left[1 - \exp\left(\frac{V^2}{\alpha} t\right) \right], t > 0 \quad (4.5)$$

The exact solution for our example can be obtained by substitution of the parameters into the above general solution. Then it becomes

$$T(0, t) = \frac{1}{2} [1 - \exp(4t)] \quad (4.6)$$

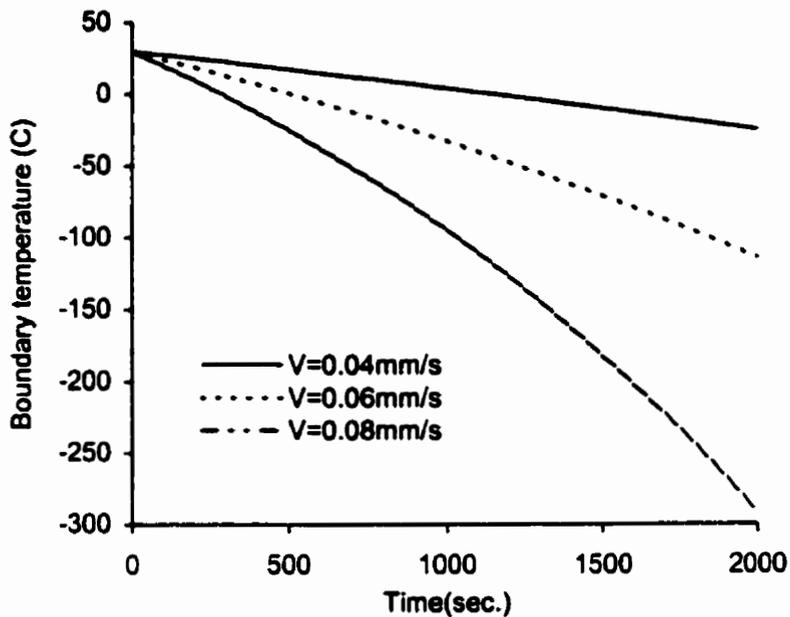


Figure 4.1: $T(0, t)$ for pure gallium as a function of time for three different interface velocities ($r = 1.3762 * 10^{-5} m^2/s$, $L = 80160 J/kg$, $c_p = 381.5 J/kg^{\circ}C$, $T_0 = 29.8^{\circ}C$)

Carslaw and Jaeger[29] commented that this is an impractical case at large t due to the exponential nature of the time-varying boundary condition. The above example is difficult to achieve in a large casting for long time periods since it is required eventually to have an infinitely low temperature at $x=0$ to maintain a constant velocity at the interface. But, this example is a good way to examine the effectiveness of the present algorithm. Also, it does represent many practical circumstances during early stages of

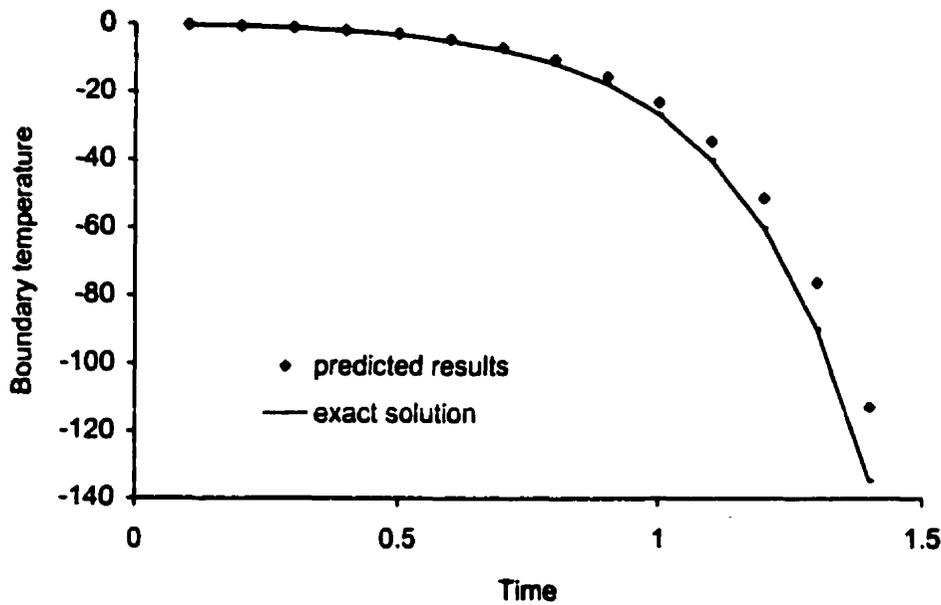


Figure 4.2: Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.1)

solidification. Figure 4.1 displays the boundary temperature $T(0, t)$ as a function of time resulting from three specified interface velocities for pure gallium. From the figure, the interface position can be determined based on the given interface velocity and time level. Additionally, although the exact solution displayed in Eq.(4.5) is defined in terms of an exponential function, it is evident that low-order polynomials could be used to describe $T(0, t)$.

The numerical solution uses a constant time step and a uniform grid. The results are shown in Figs. 4.2-4.4 together with the exact solutions. The results are compared with exact solutions given by Eq.(4.6) for three different time steps: $\Delta t = 0.1, 0.05$ and 0.005 .

From Figs. 4.2-4.4. it can be observed that very good accuracy is achieved at early time stages where the distance between the interface and the left boundary is small. Also, with a smaller time step, a more accurate numerical solution is obtained. For example, when the time step is 0.005 , the accuracy of the solution is the best and

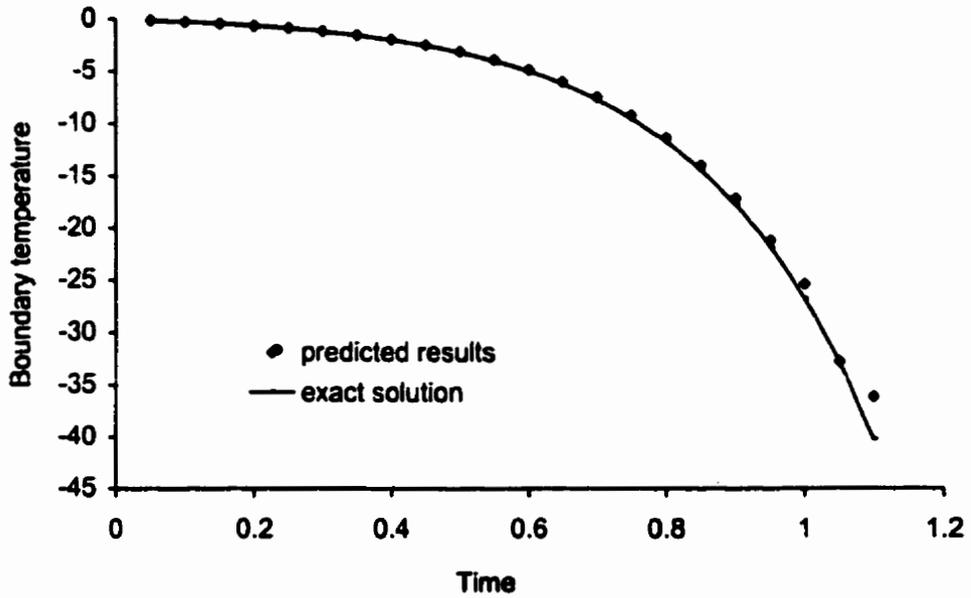


Figure 4.3: Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.05)

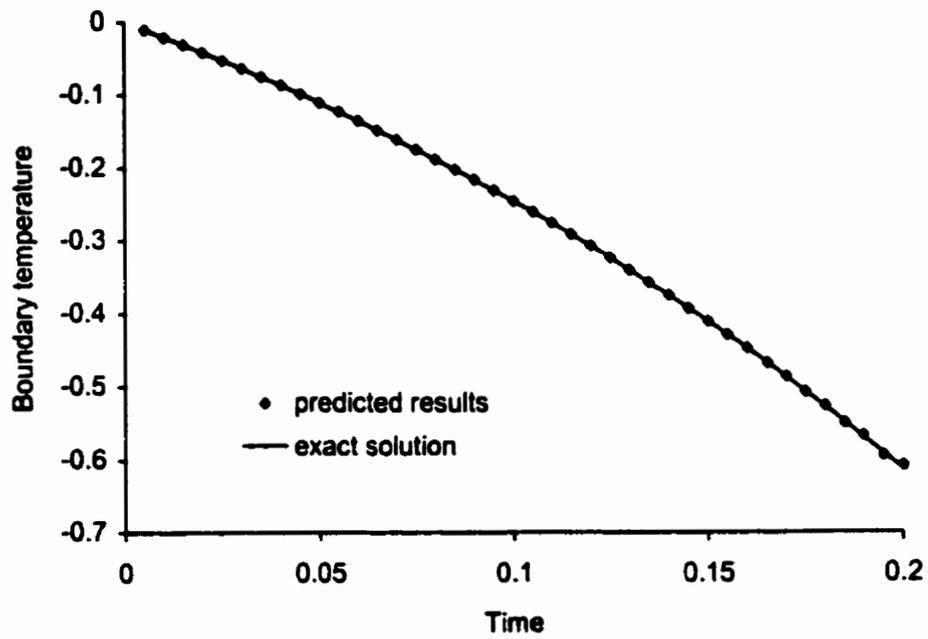


Figure 4.4: Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.005)

Δt	$\frac{1}{\Delta t}$	error
0.1	10	0.0459123
0.05	20	0.0059122
0.01	100	0.0002432
0.005	200	0.0000649

Table 4.4: Time step sensitivity (with $t=0.1$)

Δt	$\frac{1}{\Delta t}$	error
0.1	10	1.3439865
0.05	20	0.3701153

Table 4.5: Time step sensitivity (with $t=0.8$)

when the time step is 0.1, less accuracy is obtained. Tables 4.4-4.5 show the time step sensitivity, i.e. the relation between time step (Δt) and solution error at the given moment in time ($t=0.1$ and $t=0.8$). It is apparent that the numerical error decreases when a smaller time step, Δt , is taken.

Also, as time passes, the interface moves further away from the left boundary and the solution becomes more unstable and oscillations (or divergence) in the analytical solution occur (in Figs. 4.2-4.4, only the stable region has been plotted). The algorithm loses recognition of the correct boundary condition. From Figs. 4.2-4.4, it can be seen that at a smaller time step, oscillations occur earlier. In other words, the calculated boundary temperature becomes unstable at an earlier time in the case of small time steps. Thus, the calculation eventually does not predict the desired boundary temperature. For example, when $\Delta t=0.005$, the numerical solution rapidly diverges from the exact solution at about $t=0.225$; while using $\Delta t=0.05$, the oscillations start only after $t=1.1$.

Figures 4.5-4.6 show the results for different Stefan number ($Ste=0.2$ and $Ste=0.05$). The dimensionless time step is 0.05. It is shown that accurate results are achieved over a fairly wide range of Stefan numbers.

In the above numerical simulations, we find that when the distance between the interface and the left boundary becomes large, it is more difficult to control the interface

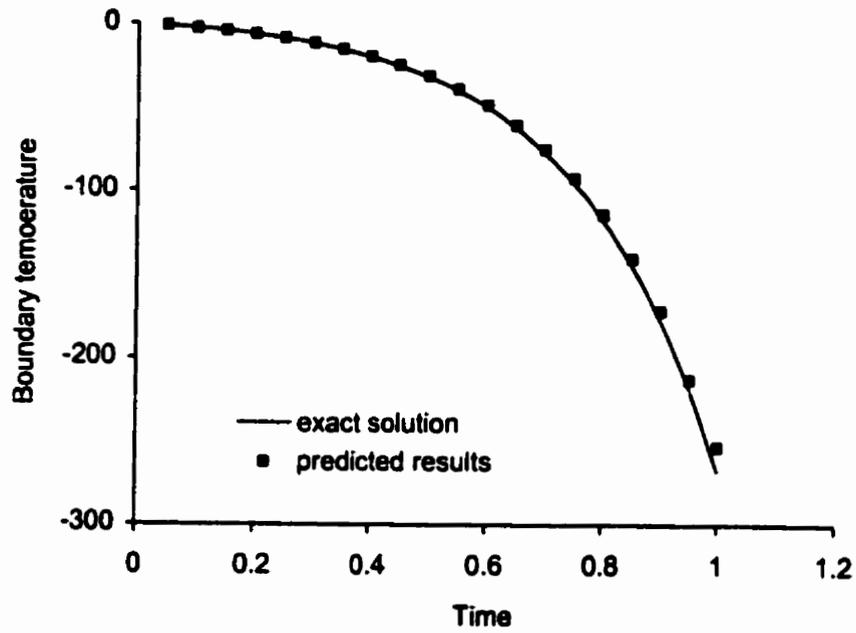


Figure 4.5: Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.05, $Ste=0.2$)

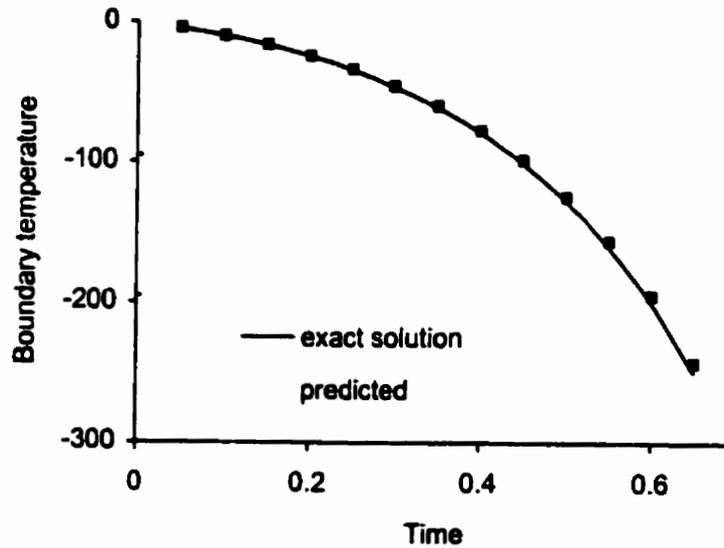


Figure 4.6: Boundary temperature at $x=0$ for Stefan problem with constant interface velocity (dimensionless time step=0.05, $Ste=0.05$)

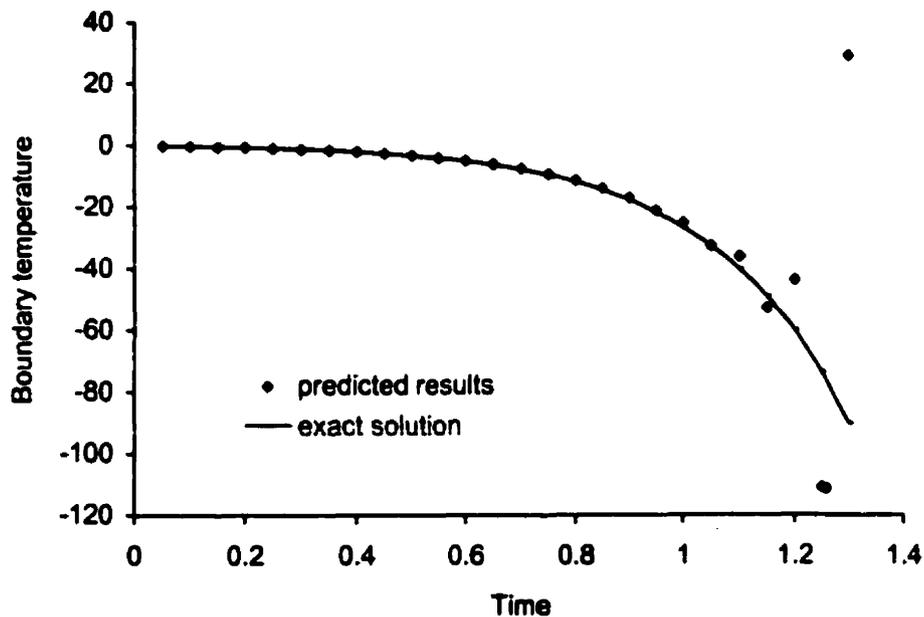


Figure 4.7: Boundary temperature at $x=0$ with $\Delta t = 0.05$ (no entropy based modification)

movement with the left boundary temperature. An instability will occur. This is illustrated in Fig. 4.7 (time step $\Delta t=0.05$). From Fig. 4.7, the results indicate that before $t= 1.1$, the simulation performs well in predicting the required boundary temperature; after $t= 1.1$, the oscillations start and they become larger and larger as time passes. The calculation no longer predicts the required boundary temperature after large t .

An entropy based method is used here to improve the numerical stability. The rationale for this approach has been described in chapter 3.

The Second Law is employed to provide a criterion for the application of an entropy based modification. During each time step, the entropy production rate is computed and it should be positive within a control volume. Otherwise, it means that the Second Law is violated and the process is not physically plausible (i.e. the numerical instability will happen). Thus, an entropy based method can be used. The thermal conductivity is corrected based on the local calculation of the entropy production rate. Then it is used to compute and modify the sensitivity coefficient prior to the next update of the boundary temperature in the iterative procedure.

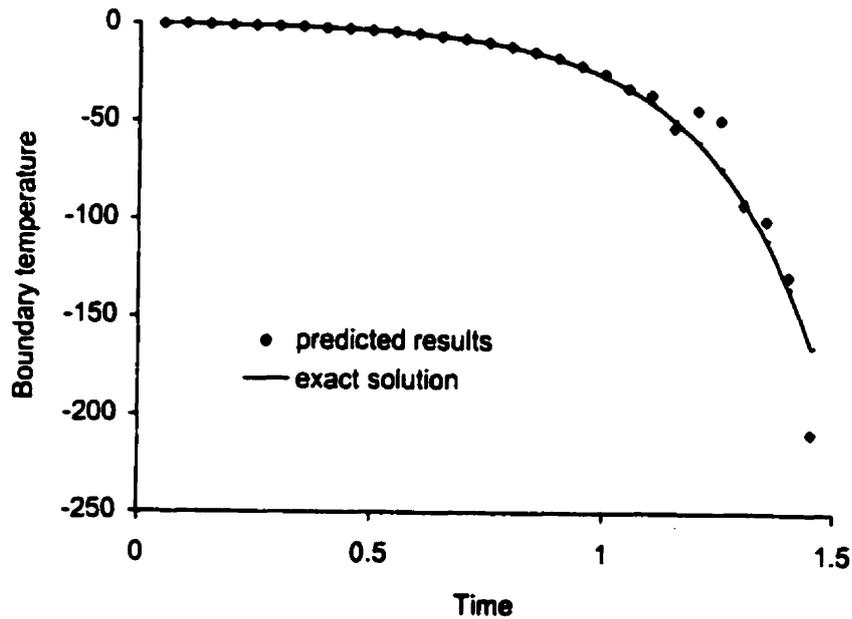


Figure 4.8: Boundary temperature at $x=0$ with $\Delta t = 0.05$ (with entropy based modification)

The results shown in Fig. 4.8 illustrate that the entropy based method performs well in improving the numerical stability of the computations. Notice that the oscillations are reduced for the next few time steps. The entropy based modification permits stable computations of the controlling boundary temperature for additional time steps. Since the entropy based modification is only applied after local violations of the Second Law, it does not affect accuracy during early stages of the results since it is not applied at those times. After it is applied for the boundary temperature, it reduces the oscillations in the boundary temperature (i.e. improves accuracy). Also, since an improved boundary estimate is obtained in this manner, it is anticipated that a more accurate temperature distribution is obtained in permitting proper conduction transport back to the interface through the solid. It is worthwhile comparing this approach to another conventional technique for stabilizing inverse computations, namely future time stepping. In future time stepping, the boundary temperature is assumed as fixed for r future time steps, and the system of equations is solved over this time range. Then the boundary temperature is updated and iterations continue until a sum of squares difference (involving interface temperature at future time and phase

change temperature) is minimized. In addition to the potential cost savings in terms of computational time, the current entropy based approach provides a physically based alternative to future time stepping. It is anticipated that entropy is a natural mechanism (often overlooked) which can stabilize results exhibiting non-physical behaviour (i.e. numerical oscillations).

4.3 Solidification of Pure Aluminum

4.3.1 One-dimensional Problem

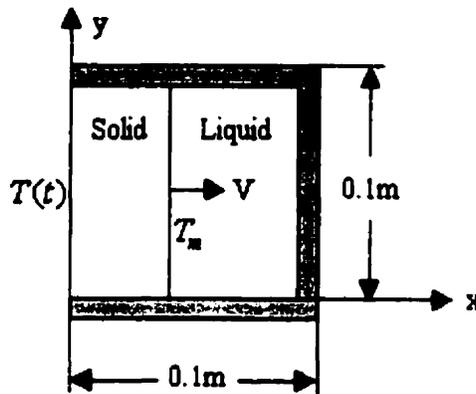


Figure 4.9: One-dimensional solidification of pure aluminum

	Symbol	Value	Units
Specific heat in liquid	c_l	1056.88	$J/kg^{\circ}C$
Specific heat in solid	c_s	1056.88	$J/kg^{\circ}C$
Heat conductivity in liquid	k_l	229.28	$W/m^{\circ}C$
Heat conductivity in solid	k_s	229.28	$W/m^{\circ}C$
Diffusivity in solid	α_s	$8.1865 * 10^{-5}$	m^2/s
Diffusivity in liquid	α_l	$8.1865 * 10^{-5}$	m^2/s
Density	ρ	2650	kg/m^3
Latent heat	L	397480	J/kg
Reference length	X_0	0.1	m
Time step	Δt	125.0	s
Melting temperature	T_m	660	$^{\circ}C$

Table 4.6: Thermal properties of pure aluminum

By considering the following solidification process involving pure aluminum, we are able to understand further the possibility of implementing the method in a practical application. The enclosure (Fig. 4.9) is initially occupied by liquid aluminum with properties as given in Table 4.6. The top, bottom and right walls are insulated and no temperature gradients are initially present in the liquid region. The liquid is initially at the melting temperature. The interface is then prescribed to move uniformly at a constant velocity ($V = 0.04\text{mm/s}$) as a vertical straight line (i.e. one dimensional phase change). The temperature at the left boundary will be predicted in order to control the interface shape and motion.

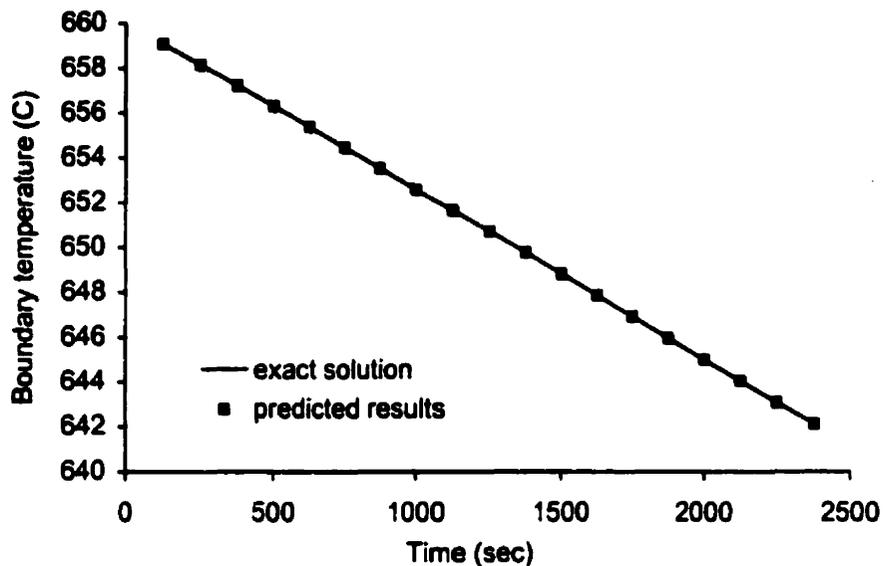


Figure 4.10: Boundary temperature for solidification process of pure aluminum (time step=125s)

The predicted boundary temperature at $x=0$ is a function of time. The results are shown in Fig. 4.10. These results agree well with the exact solution. In this example, the velocity of the interface is 0.04mm/s . The dimensionless interface velocity equals 0.0488 . Comparing this example with the last example (dimensionless interface velocity = 2), this result is like the early part of the solution. As a result, the solutions have the good agreement. The entire solidification process takes 41.6 min (time step = 125 seconds). The program is executed on a personal computer and computations require a few seconds for each time step. If we compare execution time in the program with an

actual time step in practice, then the physical time scale is substantially larger than the computational time scale. Thus, it is feasible to apply the program in a practical control setting for solidification processes. In other words, since the computational time required to advance the phase interface a given distance is more than an order of magnitude less than the physical time in the interface advance, then it appears that the algorithm may be suitable for a real-time control setting. The results from the predictions can allow the control engineer to modify the boundary temperature in time to have the desired interface movement in a given time increment. In more complicated problems, such as three dimensional predictions with fluid flow, it is likely that the computational time may exceed the corresponding physical time. However, with the rapid advance of computing technology and processor speed, it is anticipated that control engineering, merged with CFD (Computational Fluid Dynamics), will become a viable alternative to conventional control techniques. For example, recent studies involving the NASA Lewis Research Center [30] have facilitated the use of CFD by control engineers designing propulsion controls.

4.3.2 Two-dimensional Problem

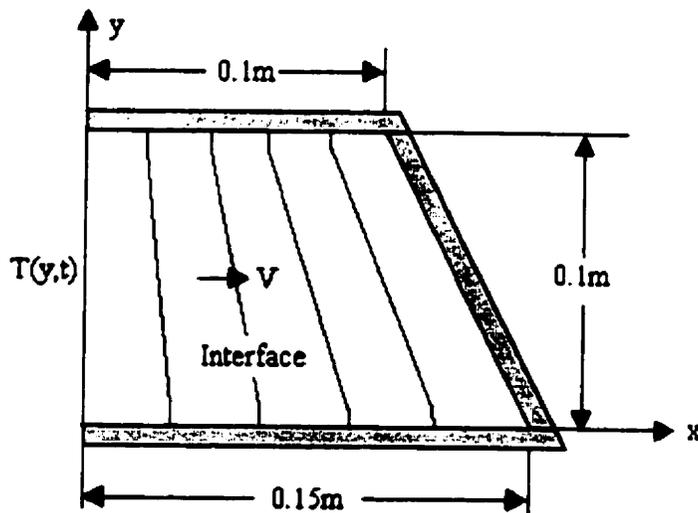


Figure 4.11: Two-dimensional solidification of pure aluminum

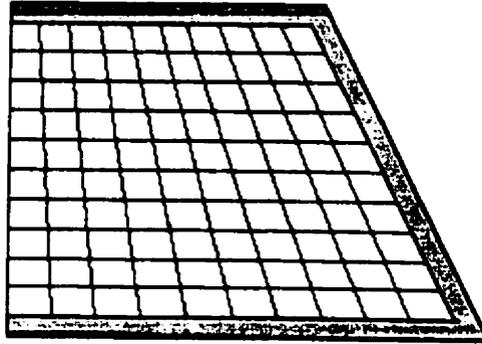


Figure 4.12: Mesh structure

In this problem, we consider solidification in a region (Fig. 4.11) occupied initially by pure aluminum at the melting temperature T_m . The top, bottom and right boundaries of the cavity are insulated. We require that the interface velocity in the x direction varies with y , while the shape of the interface remains as a straight line and all points on the interface reach the right boundary at the same time. The position of the interface will be controlled by the temperature of the left boundary of the cavity. We will solve this problem to obtain the manner in which the temperature of the left boundary should vary with time and position y , i.e. $T_0(t, y)$. This example is a 2-D problem. The dimensional parameters in this simulation are the same as in Table 4.6.

Figure 4.12 shows the mesh structure for the finite element discretization. The simulation is performed on a 20×20 mesh. The results for midpoint boundary temperature are shown in Fig. 4.13. Although no exact solution is available for comparisons, we can still compare the temperature at the mid-point of the left boundary to the average of an equivalent 1-D problem. Meanwhile, Fig. 4.14 shows the results for the distribution of temperature at the left boundary at different times and the trend of temperature is correct, i.e. the average left boundary temperature decreases with time and the temperatures along the left boundary increase from the bottom to the top boundary. Also, comparing the temperature for a point at the left boundary with the exact solution of an equivalent one-dimensional problem, at the top, we find that the temperature is higher than the exact solution; at the bottom, the temperature is lower

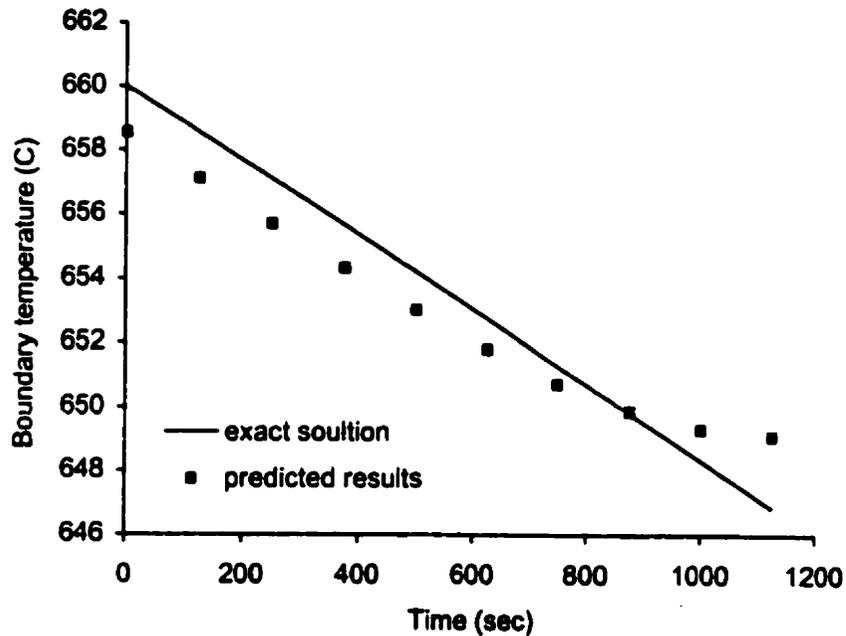


Figure 4.13: Boundary temperature at mid-point for solidification of pure aluminum

than the exact solution. It reveals the correct trends of the two dimensional effects in this problem. Heat transfer occurs from the top to the bottom regions. So, at the top, the boundary temperature does not need such a cold value as the one-dimensional problem. At the bottom, the boundary temperature should be colder than the temperature in the equivalent one-dimensional problem. Thus, in overall terms, the current inverse algorithm appears suitable for applications to multi-dimensional problems.

4.4 Solidification Problem for Pure Gallium with the Effects of Fluid Flow

Consider a solidification process with pure gallium in a two-dimensional rectangular enclosure of aspect ratio=2 (height=0.045m, width=0.09m). The top, bottom and right walls are insulated. The liquid is initially superheated, i.e. the initial temperature is larger than the melting temperature. The boundary conditions and geometry for this example are shown in Fig. 4.15. The interface is prescribed to move at a constant velocity ($V = 0.076\text{mm/s}$) and we require it to move as a vertical straight line. The dimensional parameters used in calculation are listed as follows (Table 4.7).

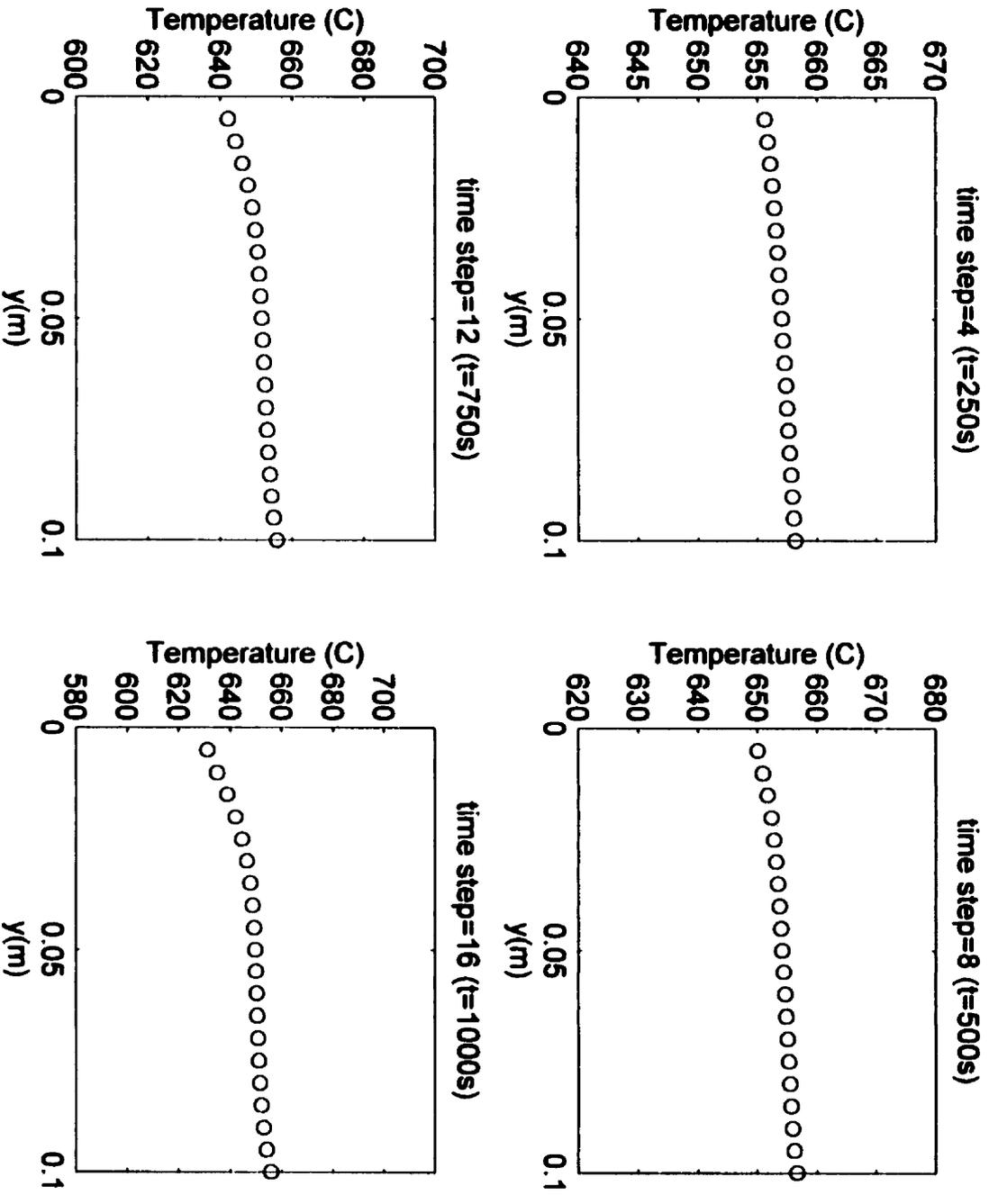


Figure 4.14: Boundary temperature for solidification of pure aluminum

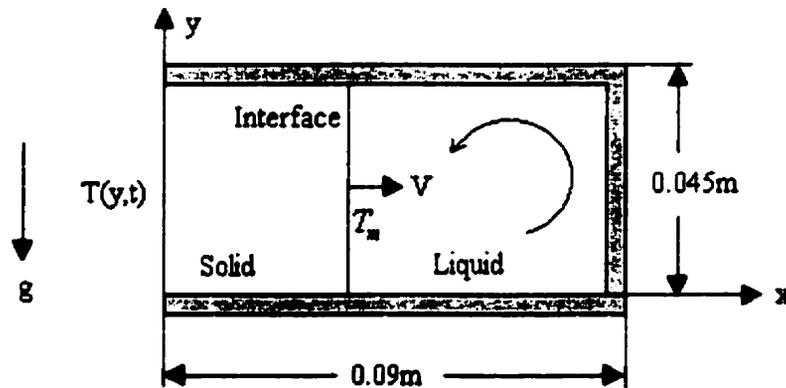


Figure 4.15: Solidification of pure gallium in a rectangular enclosure

The prediction of the boundary temperature is required.

	Symbol	Value	Units
Specific heat in liquid	c_l	381.5	$J/kg^{\circ}C$
Specific heat in solid	c_s	381.5	$J/kg^{\circ}C$
Heat conductivity in liquid	k_l	32.0	$W/m^{\circ}C$
Heat conductivity in solid	k_s	32.0	$W/m^{\circ}C$
Kinematic viscosity	ν	0.00181	m^2/s
Diffusivity in solid	α_s	$1.3762 * 10^{-5}$	m^2/s
Diffusivity in liquid	α_l	$1.3762 * 10^{-5}$	m^2/s
Density	ρ	6095	kg/m^3
Latent heat	L	80160	J/kg
Reference length	X_0	0.0045	m
Time step	Δt	58.86	s
The interface velocity	V	$7.6455 * 10^{-3}$	m/s
Melting temperature	T_m	29.8	$^{\circ}C$

Table 4.7: Thermal properties of pure gallium

We can observe some important practical benefits in performing this particular simulation. We are effectively trying to reduce the interface distortion due to natural convection arising at the interface. In this example, if a uniform left boundary temperature was specified, then cold liquid would descend along the phase interface, and warmer liquid would rise in the bulk liquid region as a result of buoyancy forces. In practical circumstances, such as materials processing and casting, this situation leads

to undesirable defects in the final material, since the convection process often leads to non-uniform dispersion of alloy constituents (i.e. defect called macro segregation) or an undesirable interface structure. This partly explains recent interests in microgravity materials testing in space since these effects of buoyant-induced flow are reduced or eliminated.

In our example, we will modify the boundary temperature in an attempt to retain a uniform interface shape rather than a profile distorted by the effects of natural convection. From a practical perspective, it means that the procedure can outline a manner by which the control engineer should orient cold regions along the boundary to produce desirable material characteristics. For example, once the boundary conditions are outlined, the grain boundaries may grow into the liquid, during solidification, in a manner such that the final material can effectively resist conditions of maximum stress.

So, in this example, if the temperatures at the left boundary remain uniform, then the shape of the phase interface would not be a vertical straight line as a result of fluid flow. Thus, the temperatures at the left boundary should be a function of time, t , and position, y , in order to obtain the desired shape of the interface (vertical straight line).

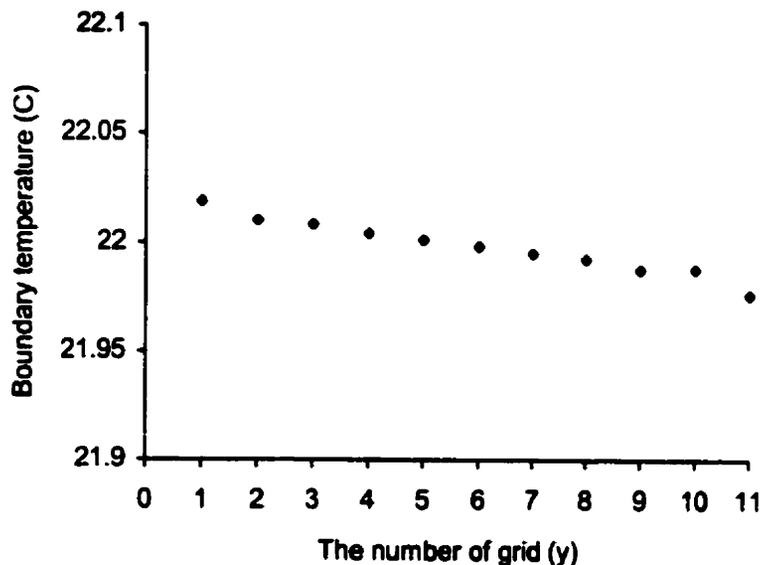


Figure 4.16: Boundary temperature for solidification with fluid flow

Figure 4.16 shows the results of the temperatures at the left boundary at early stages of solidification. It can be seen that the boundary temperature is not uniform and it decreases from the bottom to the top in order to eliminate the effect of convection on shape distortion in the interface. The distribution of velocity (direction) in the liquid region is shown in Fig. 4.17.

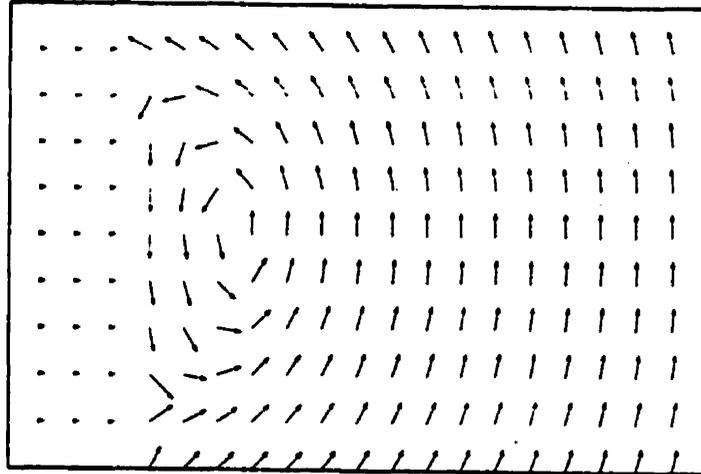


Figure 4.17: Distribution of velocity in liquid region (same scale for magnitude)

The results illustrate that a somewhat steeper temperature change is required at the two ends (upper and lower ends) of the boundary to retain the linear interface shape. This can be explained partly by the larger lateral pressure gradients in these sections resulting in larger u -velocity magnitudes there. For example, the enhanced rate of convective heat transfer near the top section of the phase interface from the bulk liquid implies that a colder temperature is required there in order to maintain a similar freezing rate as a lower section with less interfacial heat transfer. In the middle section of the boundary, the temperature change is more uniform. In any case, the main trend, involving the vertical temperature change along the boundary, due to the effect of natural convection in the liquid, can be observed in the present results. As a result, it is anticipated that the current model is suitable for applications involving phase change together with fluid flow.

Chapter 5

Conclusions and Recommendations

5.1 Conclusions

The Enthalpy Method and Finite Element Method were employed effectively to simulate one and two-dimensional inverse solidification problems where a desired interface velocity was specified. The Enthalpy Method is based on a node-jumping algorithm which calculates the boundary temperature for a given interface velocity. The interface velocity can be constant or variable and the grids could be uniform or non-uniform. The finite element method permits flexibility in inverse problems with irregular geometries. Before oscillations appear in certain examples, the solutions show good agreement with exact solutions. Also, an entropy based method is used to improve the stability of the calculations whenever oscillations appear. The entropy production rate is considered as a quantitative measure to judge when the non-physical solutions arise. By entropy based modifications, the solutions become stable for more time steps. The effects of fluid flow during solidification are also considered. The computational code simulated natural convection in the liquid during solidification. Further studies of inverse solidification problems can lead to automated control of various processes, such as casting, where the boundary temperature can be used to control the interface shape and motion as well as the properties of the solidifying crystals.

5.2 Recommendations for Future Research

Some suggestions for future research will now be provided. Firstly, experimental testing should be pursued for further validation of the procedure and an assessment of its suitability in practical applications involving materials processing. Secondly, current work on the effects of fluid flow give initial developments and further studies should be undertaken for the investigation of the inverse fluid flow algorithm.

Bibliography

- [1] Demirci, H. H., and Coulter, J. P., and Guceri, S. I., "*A Numerical and Experimental Investigation of Neural Network— Based Intelligent Control of Molding Processes*", ASME, J. of Manufacturing Science and Engineering, Vol. 119, pp. 88-94, 1997.
- [2] Stolz, G. J., "*Numerical Solution to an Inverse Problem of Heat Condition for Simple Shapes*", ASME, J. of Heat Transfer, Vol. 82, pp. 20, 1960.
- [3] Burggraf, D. R., "*An Exact Solution of the Inverse Problem in Heat Conduction Theory and Applications*", J. Heat Transfer, Trans. ASME, pp 373-382, 1964.
- [4] Beck, J. V. and Blackwell, B., "*Inverse Problems, in Handbook of Numerical Heat Transfer*", Chap. 19, ed. W.J. Minkowycz et al. Wiley, New York, 1988.
- [5] Tikhonov, A. N. and Arsenin, V. Y., "*Solutions of Ill-posed Problems*", V. H. Winston and Sons. Washington. D. C., 1977.
- [6] Bass, B. R., "*Application of the Finite Element Method to the Nonlinear Inverse Heat Transfer Problem Using Beck's Second Method*", ASME Journal of Engineering for Industry, Vol. 102, pp. 168-176, 1980.
- [7] Osman, A. M., and Beck, J. V., "*Nonlinear Inverse Problem for the Estimation of Time and Space Dependent Heat Transfer Coefficients*", Journal of Thermophysics and Heat Transfer, Vol. 3, No. 2, pp. 146-152. 1989a.
- [8] Murio, D. A., "*On the Characterization of the Solution of the Inverse Heat Conduction Problem*", ASME paper, No. 85-WA/HT-41, 1985.

- [9] Scott, E. P. and Beck, J. V., "Analysis of Order of the Sequential Regularization Solutions of Inverse Heat Conduction Problems", ASME Journal of Heat Transfer, Vol. 111, pp. 218-224, 1989.
- [10] Beck, J. V., and Murio, D. A., "Combined Function Specification-Regularization Procedure for Solution of Inverse Heat Conduction Problem", AIAA Journal, Vol. 24, pp. 180-185, 1986.
- [11] Osman, A. M., Dowding, K. J., and Beck, J. V., "Numerical Solution of the General Two-dimensional Inverse Heat Conduction Problem (IHCP)", ASME, J. of Heat Transfer, Vol. 119, pp. 39-45, 1997.
- [12] Rubinsky, B. and Shitzer, A., "Analytic Solutions to the Heat Equation Involving a Moving Boundary with Applications to the Change of Phase Problem (the Inverse Stefan Problem)", ASME, J. of Heat Transfer, Vol. 100, pp. 300-304, 1978.
- [13] Frederick, D. and Greif, R., "A Method for the Solution of Heat Transfer Problem with a Change of Phase", ASME, J. of Heat Transfer, Vol. 107, pp. 520-526, 1985.
- [14] Zabararas, N., "Inverse Finite Element Techniques for the Analysis of Solidification Processes", International J. For Numerical Method in Engineering, Vol. 29, 1569-1587, 1990.
- [15] Katz, M. A. and Rubinsky, B., "An Inverse Finite Element Technique to Determine the Change of Phase Interface Location in One Dimensional Melting Problems", Numerical Heat Transfer. Vol. 7, pp. 269-283, 1984.
- [16] Hsu, Y. F., Rubinsky, B. and Mahin, K., "An Inverse Finite Element Method for the Analysis of Stationary Arc Welding Process", ASME, J. of Heat Transfer, Vol. 108, pp. 739-741, 1986.
- [17] Zabararas, N. and Mukherjee, S., "An Analysis Solidification Problems by the Boundary Element Method", International J. For Numerical Method in Engineering, Vol. 24, pp. 1879-1900, 1987.

- [18] Zabaras, N. and Mukherjee, S. and Richmond, O., "An Analysis of Inverse Heat Transfer Problems with Phase Changes Using an Integral Method", ASME, J. of Heat Transfer, Vol. 110, pp. 554-561, 1988.
- [19] Voller, V. R., "Enthalpy Method for Inverse Stefan Problem", Numerical Heat Transfer, Part B, Vol. 21, pp. 41-55, 1992.
- [20] Sparrow, E. M., Patankar, S. V. and Ramadhyani, S. "Analysis of Melting in the Presence of Natural Convection in the Melt Region", ASME J. Heat Transfer, Vol. 99, pp. 520-526, 1977.
- [21] Marshall, R. S., Heinrich, J. C. and Zienkiewicz, O. C., "Natural Convection in a Square Enclosure by a Finite-Element, Penalty Function Method Using Primitive Fluid Variables", Numerical Heat Transfer, 1, pp. 315-330, 1978.
- [22] Strada, M. and Heinrich, J. C., "Heat Transfer Rates in Natural Convection at High Rayleigh Number in Rectangular Enclosures: a Numerical Study", Numerical Heat Transfer, Vol. 5, pp. 81-93, 1982.
- [23] McDaniel, D. J. and Zabaras, N., "A Least-Squares Front-Tracking Finite Element Method Analysis of Phase Change with Natural Convection", International Journal for Numerical Methods in Engineering, Vol. 37, pp. 2755-2777, 1994.
- [24] Voller, V. R., "Control of the Solidification and Melting of Metals Using Inverse Methods". Modeling of Casting, Welding and Advanced Solidification Processes. Edited by M. Rappaz, M. R. Ozgu and K. W. Mahin, The Minerals, Metals and Materials Society, 1991.
- [25] Naterer, G. F. , "Numerical and Experimental Techniques for Binary Solid-Liquid Phase Transition" , Ph. D Thesis, Department of Mechanical Engineering, University of Waterloo, Waterloo, Ontario, Canada, 1995.
- [26] Schneider, G. E., Raw, M. J., "Control Volume Finite Element Method for Heat Transfer and Fluid Flow Using Co-located Variables-1. Computational Procedure", Numerical Heat Transfer, Vol.11, pp.363-390, 1987.

- [27] Naterer, G. F. and Roach, D.C., "*Entropy and Numerical Stability in Phase Change Problems*", Joint AIAA/ASME 7th Thermophysics and Heat Transfer Conference, Albuquerque, New Mexico, June 15-18, 1998
- [28] Naterer, G. F. and Schneider, "*Use of the Second Law for Artificial Dissipation in Compressible Flow Discrete Analysis*", AIAA, Journal of Thermophysics and Heat Transfer, Vol. 8, No. 3, pp. 500-506, 1994.
- [29] Carslaw, H. S. and Jaeger, J. C. "*Conduction of Heat in Solids*", 2nd ed., Oxford University Press, Oxford, U. K., 1959.
- [30] Chicatelli, A., Hartley, T., Cole, G., Melcher, K., "*Interdisciplinary Modelling Using Computational Fluid Dynamics and Control Theory*", Proceedings of the American Control Conference, pp. 3438-3443, June, 1994.