To the Graduate Council:

I am submitting herewith a thesis written by Bahram Zandi entitled "A Numerical Investigation of Phase Change Problems in Axi-Symmetric Geometry." I have examined the final copy of this thesis for form and content and recommend it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Mechanical Engineering.

J. W. Hodgson, Major Professor

We have read this thesis and recommend its acceptance:

Accepted for the Council:

Vice Provest
and Dean of The Graduate School

A NUMERICAL INVESTIGATION OF PHASE-CHANGE IN AXI-SYMMETRIC GEOMETRY

A Thesis

Presented for the

Masters of Science

Degree

The University of Tennessee, Knoxville

Bahram Zandi

May 1990

ACKNOWLEDGMENTS

The auther wishes to acknowledge with appreciation Dr. J. W. Hodgson for his valuable comments, suggestions, criticism, and understanding. Thanks are also extended to professor R. J. Krane, professor and department head D. R. Pitts, and professor W. S. Johnson for their assistance, encouragements and guidance. Whatever errors remain in this work, however, are solely the authers responsibility.

ABSTRACT

This study is a numerical investigation of a heat transfer problem involving a change of phase. The enthalpy model for phase-change is employed to obtain solutions for two sets of problems in an axi-symmetric geometry.

In both problems, it is assumed that the phase change occurs at a discrete temperature and that there is no change in density upon melting or freezing. The enthalpy model is a mathematical model based on global energy conservation. Enthalpy and temperature are both used as dependent variables, and the relationship between enthalpy and temperature for the phase-change material is required to carry out the solution. This model can be used to solve heat transfer problems with or without phase change; however, it is ideal for problems involving a change of phase because it is not necessary to track the interface.

In other words, the location of the solid-liquid interface can be obtained after the solution is complete by knowing the location of two-phase elements. In most other numerical techniques it is necessary to know the exact location of the interface to carry out the computations at any instant of time, making the numerical method impractical for many problems.

The general form of the enthalpy equation, along with its non-dimensionalized form and its finite difference form are presented. A discussion of various numerical procedures used for solving phase-change problems is presented which includes a summary of the advantages and disadvantages of various schemes such as the explicit scheme, the fully implicit scheme and the Crank-Nicolson scheme. In addition, a survey of methods available for solving the resulting set of simultaneous equations is also presented.

The enthalpy algorithm is developed and applied to two sample problems. In these problems, the freezing of a solid cylinder with either an imposed boundary temperature or a convective boundary heat flux is considered. The results obtained for the imposed boundary temperature case are checked against the known solution with excellent

agreement resulting.

The problem of freezing in a more complex axi-symmetric geometry (a tear-drop shaped container) is also studied. the effects of the curved boundary on the finite difference form of the enthalpy equation is considered, and a general algorithm is devised to employ the appropriate form of the enthalpy equation for a given case, considering various situations that may exist for nodes near the curved boundary.

Limited experimental data are obtained and compared with the numerical results. The comparison showed agreement between the experimental and the numerical results.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF VARIABLES

Variable Name	Variable Description
Bi.....	Biot Number
CF.....	Coefficient in the Enthalpy Equation
D.....	Set of Continuous Linear Functionals
e.....	Specific Internal Energy
E.....	Total Energy
h.....	Heat Transfer Coefficient
k.....	Thermal Conductivity
Fo.....	Fourier Number
i.....	Specific Enthalpy
LEN.....	Length of Spherical Section in Tear-Drop Geometry
n.....	Normal Direction
NVI.....	Number of Interior Nodes in r Direction
q.....	Heat Flux
Rcy.....	Radius of Cylindrical Section
RL.....	Local Radius
Rmax.....	Maximum Radius for Given Height

R..... Nondimensional Radial Coordinate

Ste..... Stefan Number

T..... Temperature

t..... Time

V..... Volume

v..... Velocity

Xi..... Interface Location

Z..... Axial Direction

Greek Variables:

α..... Thermal Diffusivity

Δ..... Finite Change in a Quantity

ε..... Maximum Allowable Deviation of Enthalpy

ϕ..... Nondimensional Temperature

η..... Ration of Distance to Curved Boundary to Distance to Neighboring Grid-Point in Z Direction

λ..... Latent Heat of Material

λ_{tr}..... Root of the Transcendental Equation

θ..... Nondimensional Enthalpy

Θ..... Angle in Cylindrical Coordinate System

ρ..... Density of the Material

Σ..... Solid-Liquid Interface Equation

ξ..... Ratio of Distance to Curved Boundary to Distance to Neighboring Grid-Point in R Direction

CHAPTER 1

INTRODUCTION

Transient heat conduction involving a change of phase has received a great deal of attention because of its importance in many engineering applications such as melting and solidification of metals in casting, freezing of food, etc.

The solution of phase-change problems, formulated with temperature as the independent variable, is difficult due to the motion of the interface between the solid and liquid phases as the latent heat is absorbed or released at the interface. Consequently, the location of the interface is unknown and must be obtained as part of the solution. For pure substances, the solidification takes place at a fixed temperature, and the two phases are separated by a sharp interface. The thickness of this

boundary is a few molecular diameters and is generally regarded as a surface. On the other hand, mixtures and alloys change phase over a range of temperatures, and the two phases are separated by a two-phase moving boundary. In this boundary zone, however, the temperature gradient is much higher in the direction normal to the boundary than it is in other directions.

CLASSICAL STATEMENT OF THE PROBLEM :

At some instant of time t, there are two regions Vs and Vl, each containing one phase of a pure substance adjoining each other. The interface moves as time proceeds, but it is assumed that the total volume V (Vs plus Vl) is not changing with time. It is also assumed that there is no convective motion within V.

The motion of the interface and the changes in temperature are governed by boundary conditions on V, the initial condition, and the properties of the material. The heat conduction equation can be written for each of the two phases separately. In the solid phase the equation is:

$$\rho_s c_s \frac{\partial T}{\partial t} = \nabla \cdot (k_s \nabla T) \qquad (1.1)$$

Similarly in the liquid phase,

$$\rho_l c_l \frac{\partial T}{\partial t} = \nabla \cdot (k_l \nabla T) \qquad (1.2)$$

The temperature at the interface is given as:

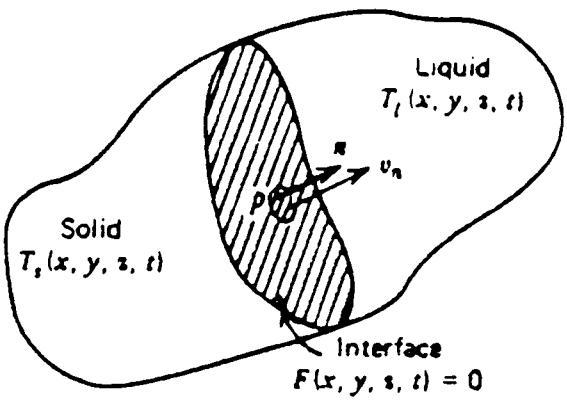
$$T_l = T_s = T_{int} \qquad (1.3)$$

Also, at the interface energy conservation requires:

$$(k \frac{\partial T}{\partial n})_s - (k \frac{\partial T}{\partial n})_l = \rho_s \lambda v_n \qquad (1.4)$$

Where $\frac{\partial}{\partial n}$ represents the derivative along the direction normal to the interface, and Vn is the interface velocity in the normal direction. Equation (1.4) is sometimes referred to as the "Stefan condition"(see Figure 1.1).

In the following paragraphs various methods of solution for phase-change problems (also called moving boundary problems, or Stefan problems, named after J. Stefan) are analyzed, and a literature survey is presented. The solution methods for phase change problems are usually divided into three classifications: exact solutions, approximate solutions using analytical and semi-analytical methods, and numerical methods.



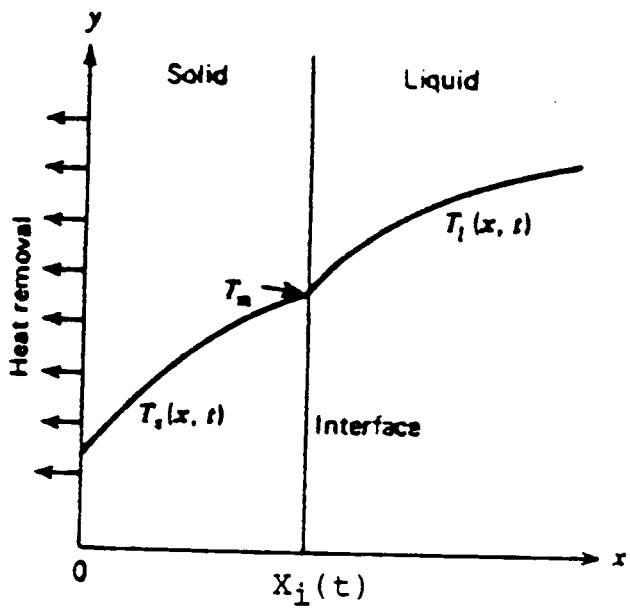


Figure 1.1 Nomenclature and Coordinates for Derivation of Boundary Condition at Interface

EXACT SOLUTIONS:

Early analytic studies of moving boundary problems date back to the seventeenth century [1]1. The first important exact solution for phase-change was given by F. Neumann in the 1860's. In 1891, J. Stefan published a paper in which he developed exact as well as approximate solutions for a one-dimensional solidification problem [2]. In this paper he also discussed the physical significance of a non-dimensional parameter (now known as the Stefan number, Ste = $c\Delta T/\lambda$), and studied the effect of periodically varying boundary conditions.

Even today, exact solutions (all of which are similarity solutions) for phase change problems are limited to one-dimensional cases involving semi-infinite or infinite regions with constant initial and boundary conditions and constant physical properties in each phase. These problems form the backbone of our understanding of phase-change processes. They are frequently used to validate numerical methods, and also to check the accuracy of approximate solutions.

---

1 Numbers in [ ] refer to entries in the Bibliography

As an example of similarity solutions for Stefan problems, consider the problem of melting a semi-infinite slab which is initially solid and at the melt temperature Tm by imposing a constant temperature at the face x=0. For this problem the governing equation and boundary and initial conditions can be written as:

$$T_t = \alpha_l \cdot T_{xx} \quad ; \quad 0 < x < X_i(t) \quad , \quad t > 0 \qquad (1.5)$$

$$T(x, t) = T_m \quad ; \quad x > X_i(t) \quad , \quad t > 0 \qquad (1.6)$$

initial conditions:

$$T(x, 0) = T_m \qquad (1.7)$$

$$X_i(0) = 0. \qquad (1.8)$$

boundary condition:

$$T(0, t) = T_m \quad ; \quad t > 0 \qquad (1.9)$$

interface condition:

$$T(X_i(t), t) = T_m \quad ; \quad t > 0 \qquad (1.10)$$

$$\rho\lambda X_i(t) = -k_l \cdot T_x(X_i(t), t) \quad ; \quad t > 0 \qquad (1.11)$$

where $X_i(t)$ represents the location of the solid liquid interface. Note that T(x,t) and X(t) are the two unknowns.

This problem admits a similarity solution called the "Neumann solution" with the similarity variable defined by $\zeta = \frac{x}{\sqrt{t}}$. The Neumann solution for this problem is:

$$X_i(t) = 2\lambda_{tr}\sqrt{\alpha_l \cdot t} \tag{1.12}$$

$$T(x, t) = T_i - (T_i - T_m)\frac{erf\left(\frac{x}{2\sqrt{\alpha_l t}}\right)}{erf(\lambda_{tr})} \tag{1.13}$$

where λ_{tr} is the root of the following transcendental equation:

$$\lambda_{tr} e^{\lambda_{tr}^2} \cdot erf(\lambda_{tr}) = \frac{Ste_l}{\sqrt{\pi}} \tag{1.14}$$

Similarity solutions also exist for two-phase semi-infinite Stefan problems (i.e., for melting, problems in which the solid is initially at a uniform temperature lower than the melt temperature, so that the heat equation must be solved in two phases). Similarity solutions also exist in cylindrical geometries for the case of melting of an infinite region by means of a line source (located on the axis of the cylinder).

APPROXIMATE METHODS:

In this section, a short literature review of analytical and semi-analytical methods is presented. The majority of the references, especially those of purely

analytical nature, deal with one-dimensional problems.

The analytical approximation methods provide qualitative information for many phase-change problems. These methods are particularly useful for obtaining quick and approximate information about a problem (such as approximate melt time). The analytical methods can be divided into the following categories:

1) Quasi-stationary approximation.
2) Perturbation expansions.
3) Megerlin method.
4) Goodman's heat balance integral method.

The quasi-stationary approximation is ideal for problems in which the value of the sensible heat is negligible compared to the latent heat of the material (low Stefan number problems). For these problems the governing equation reduces to the Laplace equation [3]. Solomon, Alexiades, and Wilson [4] have shown that the solution of the Stefan problem with a convective boundary condition approaches the quasi-stationary approximation solution as the specific heat of the liquid phase tends to zero. The quasi-stationary approximation can be used to obtain solution to moving boundary problems with time variant boundary temperatures, boundary fluxes and

convective boundary conditions.This method has also been successfully applied to phase-change problems in cylindrical and spherical coordinates.

The perturbation method is another analytical approach used by several researchers to obtain approximate solutions for moving boundary problems. This approach utilizes series expansions to reduce the problem into a number of simpler problems which, hopefully, can be solved explicitly. Typically, only the first few terms in the expansions can be obtained due the to increasing complexity of the resulting problems as one takes into account higher order solutions. It is also very difficult to apply this method to problems involving more than one physical dimension. The Stefan number ($Ste = c \cdot \Delta T / \lambda$) is usually taken to be the small parameter in the perturbation expansion. Expansions in Stefan number can be considered as corrections to quasi-stationary approximations because, as mentioned earlier, the quasi-stationary approximation is based on the assumption that the Stefan number is negligibly small.

Jiji and Weinbaum [5] used the perturbation technique to study the freezing of a finite slab which is initially at a uniform temperature above the melting point. Using

the technique of matched asymptotic expansions, Jiji and Weinbaum came up with expressions for the interfacial motion and the temperature distribution. Spaid and coworkers' paper on sublimation from a surface subjected to unsteady heat flux [6] is another example of an application of the perturbation method to phase change problems. This method was applied by Alexiades [7] to predict the freezing of dilute alloys. One of the characteristics of alloy solidification is that the phase change temperature is not constant; it depends on the alloy concentration. In these freezing processes solidification is the result of removal of the latent heat as well as solute redistribution.

The Megerlin method [3] and the integral heat balance method are other examples of analytical approximation methods. In both of these methods, a parabolic temperature profile is assumed, and an attempt is made to find the coefficients of the parabola via the solution of a resulting ordinary differential equation [8].

Goodman [9] used the heat balance integral method to obtain the solution of a one-dimensional transient melting problem. Another version of the heat balance integral method was applied by Poots [10] to investigate the

solidification inside a square prism. Poot's method provides a straightforward approach for obtaining results in simple geometrical configurations.

Several analytical and semi-analytical methods for solving phase change problems are presented in [11]. The variational formulation was used by Biot [12] to obtain solutions to one-dimensional transient phase change problems.

The moving heat source method, first applied by Lightfoot [13], is an example of the semi-analytical approach for moving boundary problems. In this method, the liberation or absorption of the latent heat is represented by a moving plane heat source or sink located at the interface, and the analysis of the phase change problem generally reduces to solving an integral equation [11]. Using the moving heat source method, Jiji and Rathjen [14] developed a similarity solution for freezing or melting in an infinite two-dimensional right-angled corner. The same type of similarity solution was developed by Budhia and Kreith [15] for infinite wedges.

The embedding technique, first used by Boley [16] to solve a one-dimensional melting problem, has been applied

to a number of phase change problems. This semi-analytical method seems to be versatile enough to obtain approximate solutions to multi-dimensional problems. Sikarskie and Boley [17] and Boley and Yogoda [18] applied this method to obtain series solutions for some phase change problems. In the above papers, it was assumed that the molten liquid is removed instantaneously. An important feature of these solutions is that they apply for only a short time after the start of the phase change process.

In some papers a mapping technique is used to solve moving boundary problems. This method is applicable to problems in which heat capacity effects are much smaller than latent heat effects. When this holds, the temperature distribution in each of the two phases obeys the Laplace equation [3]. One method used to solve the Laplace equation in irregular regions is called "conformal mapping." Siegel, Goldstein and Sovino [19] used conformal mapping to solve some two-dimensional phase change problems.

From the forgoing paragraphs on analytical and semi-analytical methods it is clear that although many methods are available for phase change problems, each one is suitable for only a certain class of problems, and a

small variation in a given problem may result in failure of a previously working method. Therefore, the significant weakness of the analytical and semi-analytical methods is their lack of generality, which is caused by the presence of the moving interface between the two single-phase regions. As mentioned earlier, the position and shape of the interface are both unknown functions of time. Moreover, the interface condition which links the temperature distributions in the solid and the liquid phases is a non-linear partial differential equation. The final drawback of most analytical approximation methods is the inability to assess 'a priori' the accuracy of the method; thus, for each problem, a validation must be performed by comparison with other methods and experiments.

NUMERICAL METHODS:

The third main category of solution methods for phase change problems involves approximating the equations that represent the physical processes and usually takes advantage of the power of the digital computer to perform the required calculations. Since digital computers are capable of representing only a finite number of rational numbers, they can deal only with discrete approximations

of the continuum concepts. Moreover, since computer memory sizes and computational time are limited the amount of information that can be processed is also limited.

The limited capabilities of computers impose certain restrictions on numerical solutions of physical processes. The physical domain must be approximated by a "finite" number of control volumes. Time may vary only in discrete steps. In addition, mathematical concepts (such as derivatives) are approximated by finite differences.

Some of the main advantages of numerical methods are discussed next. They usually have wider ranges of applicability, the results are limited only by the grid-size, and high degrees of accuracy can be achieved by reducing the grid sizes and time-steps (as long as the computation time remains reasonable and "round off" errors do not accumulate to unacceptable levels). Furthermore, numerical methods are often easier to apply, and a vast knowledge of mathematical skills (such as may be required to apply to some of the semi-analytical techniques) is not needed. The main disadvantage of the numerical techniques is that generalizing the physical meaning of the results may not be easy.

Numerical methods for solving phase-change problems can be divided into two groups, based on the choice of the dependent variable.

In the first group, temperature is taken to be the only dependent variable. The governing differential equations for the solid and the liquid regions and the interface condition are then solved numerically. Such approaches are collectively referred to as "front tracking" schemes because they attempt to track the interface using the Stefan condition. There exist a large number of papers in the literature using such methods to obtain numerical solutions to phase-change problems. Some of these papers are reviewed in the next few paragraphs.

Tao [20] used a modified formulation of Binder-Schmidt along with a finite difference scheme to solve the set of coupled differential equations to describing the freezing of a saturated liquid in cylinders and spheres. Springer and Ollson [21] proposed a standard explicit finite difference approach to study freezing in a cylindrical annulus. The movement of the interface within the annulus was illustrated, but due to the form of the dimensionless variables used, the final solution for the case of a solid cylinder could not be obtained by letting

the inside radius approach to zero [20]. Rathjen and Jiji [14] used the explicit finite difference method to verify the results of their semi-analytical solution in a corner region.

To overcome the difficulty of handling the normal derivatives of temperature that exists in the interface condition of multi-dimensional problems, Lazaridis [22] expressed $\frac{\partial T}{\partial n}$ in terms of $\frac{\partial T}{\partial x}$ and $\frac{\partial T}{\partial y}$ along the coordinate axis. He then was able to obtain solutions to several two-dimensional problems involving finite and infinite corners.

The problem of phase-change in the presence of fluid motion has also been studied. Bilenas and Jiji [23] employed the explicit difference formulation to investigate flow in a pipe with surface solidification. In all the papers mentioned so far, the explicit difference technique was used together with a fixed grid. The main drawback of the explicit difference method, however, is the necessity to restrict the maximum time-step to avoid instability. For example, any attempt to improve the accuracy by reducing the space step-size is offset by the increased effort resulting from the need for a much smaller time-step. For this reason, the explicit

technique, in spite of its relative simplicity, is suited only for cases in which the Stefan number is high and the time-span of the transient is small (such as casting of metals).

The combination of the implicit method and a fixed grid discretization has not been much used. This is due to the effect of the interface condition on the iterative scheme.

In one class of the temperature-based methods, the spatial variables are transformed in such a way that the interface is fixed independently of time. This procedure, which is sometimes called the immobilization technique, makes applications of the implicit difference method to phase change problems easier. The transformed plane is discretized and consequently the shape of the grid in the physical plane changes with time. In the transformed space the governing differential equations contain parameters involving interface location. The finite difference equations are easily written near the interface, but the resulting equations are very complicated and must be solved iteratively [24]. After the procedure is completed, it is necessary to transform the results back to the physical domain.

Duda and coworkers [25] used the immobilization method in their analysis of phase change in a cylindrical container. The authors used a fully implicit difference method, and employed an iterative scheme to solve the resulting non-linear algebraic equations. Spaid and coworkers [6] also applied the immobilization technique to verify their perturbation solution of the sublimation problem mentioned earlier. In their treatment, they made a transformation from Cartesian coordinate (X,Y) to the transformed coordinates (X,Z) where Z=Y-Y*, and Y* is the Y coordinate of the interface. They then applied a modified explicit difference scheme.

Isotherm migration is another class of temperature-based numerical techniques for solving phase change problems. This approach is suitable for problems with boundaries on which the temperature is specified. The method consist of making one of the spatial coordinates the dependent variable and time, temperature and the other spatial coordinates the independent variables. Consequently, if a uniform mesh is employed in the new variables, in the physical plane the corresponding mesh will be fine in regions with large temperature gradients, and coarse in regions with small temperature gradients (a

desirable characteristic). Crank and Gupta [26] used this approach with an explicit finite difference method to solve a number of two-dimensional problems. The authors mentioned that since the transformed equations are non-linear, it would be very difficult to use the implicit difference scheme.

Now, consider the second main class of numerical methods for solving phase change problems. In this class of numerical methods, the enthalpy is used as the dependent variable along with temperature. The enthalpy method (also called the "single region method" or the "weak formulation") is based on the fact that the temperature gradient causes an enthalpy gradient. The conduction heat transfer is therefore related to the enthalpy gradient.

The main attractive feature of this method is the elimination of the interface from the calculations. In other words, no modification of the numerical scheme is necessary in order to satisfy the conditions at the moving interface, and therefore, no tracking of the interface is required. The location of the interface, however, can easily be found by knowing the location of the two-phase elements [24]. Since the method used in the present study

falls in this group, a detailed description of the method and the governing equations will be presented in the next chapter.

In most of the existing work based on the enthalpy method, the phase change is assumed to occur diffusively over a range of temperatures, and the curve of enthalpy versus temperature is assumed to have a finite slope at the phase change temperature. While this is a valid assumption for impure substances and alloys, the approach has also been applied to cases involving a single phase change temperature [24]. Hashemi and Sliepcevich [27] employed this approximation to solve a two-dimensional problem. For pure substances, the authors suggested using a range of temperatures large enough for the numerical method to work, but small enough to avoid large errors. However, their results are very sensitive to the magnitude of the assumed range. Meyer [28] used the diffuse enthalpy method along with a fully implicit scheme to solve a two-dimensional phase change problem. His technique is unconditionally stable and the computational effort is independent of the range assumed, but the range must be nonzero. Shamsundar and Sparrow [29] modified the implicit scheme of Meyer and constructed a discrete version of the enthalpy method. Comini and coworkers [30] employed the

enthalpy technique with a diffuse range using a finite element method. Their results also show dependence on the range, and numerical instability appears as the range is reduced.

In reference [24], following a comparison between the diffuse enthalpy method and the discrete enthalpy method, Shamsundar concluded that for pure substances the diffuse model is not a good approximation and that the discrete enthalpy method is just as easy to use and closer to reality. He reworked the problem introduced in [28] using the discrete model, and obtained high degrees of accuracy in reference [29].

CHAPTER 2

THE ENTHALPY MODEL

As introduced in Chapter 1, the enthalpy method consists of a single differential equation with two dependent variables (temperature and enthalpy) which is applicable to the entire region comprising single-phase zones and two-phase zones.

The main attractive feature of this model is its relative simplicity resulting from the elimination of the moving boundary from the calculations. In this approach the "jump" condition is not forced on the solution, but is obeyed by the solution as a natural boundary condition. This makes the problem equivalent to a non-linear heat conduction problem with no change in phase. Another advantage of this model is its wider range of applicability. The same mathematical formulation applies

to the case involving phase change at a discrete temperature, cases involving phase change over a range of temperatures, and also cases involving no change of phase. The location of the interface can be recovered a posteriori as the set of points on which the energy lies between zero and $\rho\lambda$. It may happen to be a sharp surface, or it may be an extended zone.

The enthalpy model was described in one dimension by the Soviet mathematician Kamin [31] .It was applied to one dimensional problems in which phase change occure over a range of temperatures by Byresan [32]. The enthalpy model is obtained by applying the local energy conservation equation throughout the material.

$$\rho . e_t(\bar{x}, t) + div\vec{q}(\bar{x}, t) = 0 \qquad (2.1)$$

The phases will be distinguished only by the relationship between enthalpy and temperature for the material of interest.

For years the application of the enthalpy technique was restricted to problems involving substances that change phase diffusively over a range of temperatures, ie, the impure substances. This was due to the fact that the

governing differential equation contained terms such as $\frac{\partial e}{\partial t}$ and $\nabla(k\nabla T)$ and since (e) and (k.gradT) are both discontinuous at the interface, the above terms are undefined and the equation is not applicable at the interface.

However, since the introduction of the Dirac Delta function and the concept of the "distributional derivative" and "weak derivative", full mathematical meaning has been given to such terms. In 1948, L. Schwarts published the theory of distributions, which is the foundation of the concept of the weak derivative. This was followed by a number of studies in which the theory of partial differential equations was recast into the frame work of the weak derivative. Today's enthalpy method is a direct consequence of those developments.

The classical derivative is a "local" concept requiring pointwise smoothness of the function. In the concept of weak derivative, however, the requirement of continuity at a point is replaced by integrability over a set, which is a "global" concept. Also, it should be noted that the weak derivative is merely a more general concept of derivative, and it coincides with the classical derivative whenever the latter exists. In order to

understand the concept of the weak derivative, it is necessary to introduce the idea of a distribution and distributional derivatives.

The space $D'(\Omega)$ of distributions on Ω (Ω is an open bounded set in RN) consists of all continuous linear functionals on a given set of test functions, $D(\Omega)$. Note that a functional is a special kind of function whose domain contains functions. The value of a distribution, T, on a test function, f, is denoted by <T,f>, as opposed to T(f).

$$T: f \rightarrow \langle T, f \rangle \in \Re$$

As an example of a distribution on Ω consider any ordinary function, u which is integrable on Ω by means of:

$$\langle u, f \rangle = \int_{\Omega} u(x).f(x).dx \quad , \quad f \in D(\Omega) \qquad (2.2)$$

A distribution can have derivatives of all orders. The distributional derivative of T with respect to xj is also a distribution, $\frac{\partial T}{\partial x_j}$ such that:

$$\left\langle \frac{\partial T}{\partial x_j}, f \right\rangle = -\left\langle T, \frac{\partial f}{\partial x_j} \right\rangle \quad ; \quad f \in D(\Omega) \qquad (2.3)$$

Weak derivatives are special kind of distributional derivatives. Consider u, an integrable function on the open bounded set Ω . The weak derivative of u on Ω with respect to x is an integrable function,v ,such that the following integral equality holds:

$$\int_{\Omega} u \frac{\partial f}{\partial x_j} \cdot dx = -\int_{\Omega} (v \cdot f)\, dx \quad ; \quad f \in D(\Omega) \tag{2.4}$$

It can easily be shown that this is, in fact, a generalization of the classical concept of a derivative by showing that the weak - derivative coincides with the classical one whenever the latter exists.

To illustrate the concept of the weak solution of a partial differential equation, consider the one-dimensional heat conduction equation:

$$T_t = \alpha \cdot T_{xx} \quad ; \quad 0 < x < 1 \quad , \quad t > 0 \tag{2.6}$$

with initial and boundary conditions given as:

$$T(x, 0) = T_i(x) \quad ; \quad 0 \le x \le 1 \tag{2.7}$$

$$-k \cdot T_x(0, t) = h.[T - T(0, t)] \quad ; \quad t > 0 \tag{2.8}$$

$$T(l, t) = T_l(t) \quad ; t > 0 \tag{2.9}$$

A classical solution of the above problem in D (D = (0,1) x (0,tf)) is a function T(x,t) such that:

1) T(x,t) is continuous in the closure of D.

2) Tx(x,t) is continuous in D1; D1 = [0,1) x (0,tf)

3) Tt(x,t) and Txx(x,t) are continuous in D and they satisfy the equation pointwise.

Now, in order to admit discontinuities in T, one has to relax the requirements that T and its derivatives be continuous in D, and seek the weak solution of the above problem.

The weak solution can be formulated as:

$$\langle (T_t - \alpha T_{xx}), f \rangle = \iint_D (T_t - \alpha T_{xx}).\, f\, .\, dxdt = 0 \qquad (2.10)$$

where f is a test function with continuous first and second derivatives.

The above expression can be integrated by parts. After some modification and employing the initial and boundary

conditions the following equation is obtained:

$$\iint_D T\,.(f_t + \alpha f_{xx})\,.\,dxdt + \int_0^l f(x,0)T_t(x)\,.\,dx =$$

$$\int_0^{t_f}\left\{\left(\frac{h}{\rho c}\right)[T(0,t) - T_\infty(t)]\,.\,f(0,t) + \alpha T_t(t) f_x(l,t)\right\}\cdot dt \qquad (2.11)$$

At this point we can define the weak solution.

T is a weak solution of the above problem if:

1) Ti is twice integrable in (0,1) and Tl and T_∞ are twice integrable in (0,tf).

2) T is twice integrable in the interval (0,tf).

3) T and Tx are twice integrable in interval (0,1).

4) Tt is twice integrable in D.

5) Equation (2.11) holds for every test function f such that f has continuous first and second derivatives in closure of D, and fx(0,t)=0, f(1,t)=0, f(x,tf)=0 .

Note that here we only need to require integrabilty over a region; hence, a weak solution can be very continuous indeed.

REFORMULATION OF STEFAN PROBLEM:

Having shown what is meant by a weak solution in general, we proceed to reformulate equations (1.1) through

(1.4) in terms of the enthalpy on a fixed domain. As the reformulation equation we consider the energy equation as valid throughout the region:

$$\rho\left(\frac{De}{Dt}\right)+\nabla(\vec{q})-\frac{Dp}{Dt}-\Phi=0. \qquad (2.12)$$

The last two terms in equation (2.12) represent the flow work and the viscous "dissipation". For the problem of interest in this study these terms are zero. Furthermore, if one assumes constant density for each phase, or that the fluid is initially at its saturation temperature (for solidification problems) the terms that involve fluid velocities may be dropped from the equation; otherwise, the momentum equation must be solved in the liquid region.

Assuming conduction as the only means of heat transfer, equation (2.12) can be rewritten as:

$$E_t+\nabla q=0. \qquad (2.13)$$

where q is the rate of heat transfer given by:

$$q=-k.\nabla T \qquad (2.14)$$

and E is the enthalpy, which is related to the temperature through the following expressions:

$$E(x,t)=\begin{cases}\displaystyle\int_{T_m}^{T(x,t)}\rho c_s\cdot dt & ; \quad T(x,t)\le T_m \quad (solid)\\ \rho\lambda+\displaystyle\int_{T_m}^{T(x,t)}\rho c_l\cdot dt & ; \quad T(x,t)\ge T_m \quad (liquid)\end{cases} \tag{2.15}$$

where λ is the latent heat of the material. Note that the enthalpy E is the sum of the sensible heat and the latent heat in the liquid region. Equation (2.15) can be simplified if one assumes the specific heat is constant within each phase:

$$E(x,t)=\begin{cases}\rho c_s(T-T_m) & ; \quad T(x,t)\le T_m \quad (solid)\\ \rho\lambda+\rho c_l(T-T_l) & ; \quad T(x,t)\ge T_m \quad (liquid)\end{cases} \tag{2.16}$$

It can be seen that with the temperature - enthalpy relationship defined as above, equation (2.13) can be reduced to equations (1.1) and (1.2) in the solid and the liquid regions, respectively. Also, across the interface (where the enthalpy experiences a discontinuity) the conservation form of equation (2.13) reduces to the Stefan

condition, equation (1.4). Therefore, since the interface conditions given by equations (1.3) and (1.4) are automatically satisfied by any solution of equation (2.13), equation (2.13) (along with relations (2.15)) is equivalent to the classical set of equations for phase-change problems. Equation (2.13) will be frequently referred to as the "enthalpy equation" in the reminder of this thesis.

DIMENSIONLESS VARIABLES:

It is helpful to non-dimensionalize the enthalpy equation to improve the generality and usefulness of the solutions. The spatial coordinates are non-dimensionalized by dividing by a characteristic length, l. The time coordinate is made dimensionless by introducing the Fourier number, Fo, where the Fourier number is defined as

$$Fo = \left[\frac{k}{\rho_s c_s L^2}\right] \cdot t \qquad (2.17)$$

Next the dependent variables are made dimensionless. For single-phase elements, the enthalpy variable is non-dimensionalized by making use of the latent heat, λ, as the characteristic enthalpy difference:

$$\theta = \frac{e - e_{sat}}{\lambda} \qquad (2.18)$$

Note that the above definition implies that for saturated liquid θ is equal to one, and for saturated solid θ is equal to zero. For two-phase elements the dimensionless enthalpy is:

$$\theta = \frac{\left(\frac{V_s e_{s,sat} + V_l e_{l,sat}}{V}\right)}{\lambda}$$

$$\theta = \frac{V_l}{(V_l + V_s)} \qquad (2.19)$$

In the above definition the volume based average is used. Alternatively, a mass based average can be used. However, if one assumes no density change, the two formulations will be identical. The dimensionless temperature can be defined as:

$$\phi = \frac{c_s}{\lambda} \cdot \int_{T_m}^{T} \left(\frac{k}{k_{sat}}\right) \cdot dT \qquad (2.20)$$

For problems in which the change in temperature is not very large, the thermal conductivity, k, can be

assumed constant. Hence, equation (2.20) reduces to:

$$\phi = \frac{c_s(T - T_{sat})}{\lambda} \qquad (2.21)$$

From the above definitions it can be seen that the dimensionless temperature is equal to zero at saturation. For subcooled solids, both θ and Φ are negative, and for superheated liquid θ is greater than one while Φ is greater than zero.

by substituting the dimensionless variables in equation (2.13) and relation (2.16) the non-dimensional form of the enthalpy equation can be obtained:

$$\frac{\partial \theta}{\partial Fo} = \nabla^2 \phi \qquad (2.22)$$

$$\theta = \begin{cases} \phi & (solid) \\ 1. + \phi & (liquid) \end{cases} \qquad (2.23)$$

WEAK FORMULATION OF THE STEFAN PROBLEM:

The existence of the weak solution for phase-change problems was first established by the Soviet mathematician

Kamin for imposed boundary temperature problems. This was followed by existence proofs for more general non-linear equations and boundary conditions by Friedman in 1968 and Damlanian in 1977 .In the next few paragraphs, the weak formulation of a typical phase-change problem is illustrated.

Consider the non-dimensional phase-change problem described by equation (2.22) and relations (2.23) with the initial condition given by:

$$\Theta(x,0) = \Theta_o \quad ;$$

is the initial enthalpy distribution with boundary conditions:

$$\Phi(x,t) = g(x,t) \qquad \text{on boundary (1) of } \Omega$$

$$\frac{\partial \Phi}{\partial n}(x,t) = h(x,t) \qquad \text{on boundary (2) of } \Omega$$

Θ and Φ constitute a weak solution of the previous Stefan problem if:

1) Θ and Φ are bounded integrable functions in D.

2) The following integral equation

$$\int_0^{t_f}\int_\Omega \left[\theta \cdot f_t + \phi \cdot \nabla^2 f\right] . \, dx \, . \, dt + \int_\Omega \theta_0(\bar{x}) \cdot f(\bar{x}, 0) . \, dx =$$

$$\int_0^{t_f}\left(\int_{\delta_t \, \Omega} g \cdot \frac{\partial f}{\partial n} ds - \int_{\delta_u \, \Omega} h . f . ds \right) . \, dt \qquad (2.24)$$

holds for every test function f given by:

$$f \in F\left\{ f \in C_2(\bar{D}) : f(\bar{x}, t_f) = 0 \; f\,|_{\delta_t \, D} = 0 \, , \left.\frac{\partial f}{\partial n}\right|_{\delta_u \, D} = 0 \right\} \qquad (2.25)$$

Some important characteristics of the weak solutions of phase-change problems can best be described through the following two theorems.

Theorem 1:

a) Any classical solution of the Stefan problem is also a weak solution.

b) If (Θ , Ψ) constitute a weak solution of the enthalpy equation, and if there exists a function Σ (x,t) (equation for the smooth interface) such that $\frac{\partial \Sigma}{\partial t}$ and $\nabla\Sigma$ exist and are continuous in the domain of interest, and also the Stefan condition (i.e., equation (1.4)) is

satisfied across the interface, then $\{ \theta , \Phi \}$ is also a classical solution.

The above theorem establishes that the weak solution concept is indeed a generalized solution concept; moreover, a weak solution is capable of existing when a classical solution may not exist.

Theorem 2: (Uniqueness Theorem)
There exists at most one weak solution for any phase-change problem.

NUMERICAL SCHEME:

For most cases, analytical solutions of the enthalpy equation are difficult to obtain. This leads to an examination of numerical schemes. The two most widely used numerical methods are finite element methods and finite difference methods. In both of these schemes, continuous space is replaced by a number of grid points, and the values of the unknown quantity at these points are chosen such that the governing equations are satisfied approximately.

Finite element methods are more appropriate for

problems involving complicated geometries. In addition, these methods can handle non-linear boundary conditions and variable thermal properties with little difficulty. However, finite element methods have not been used often to solve the enthalpy equation due to difficulties in selection of an appropriate basis function.

Finite difference methods, on the other hand, are ideal for phase-change problems. It has been shown that both explicit and implicit finite difference forms of the enthalpy equation converge to the weak solution [33]. Also, The properties of finite difference schemes have been investigated thoroughly, and there are several efficient algorithms available for solving the resulting simultaneous equations. Their main drawback, however, is their adaptation to complicated geometries. Nevertheless, the method used in this study is a finite difference method.

The first step in obtaining a finite difference representation of the problem is to divide the region into a number of small elements. At the center of each element a nodal point is placed. The value of the enthalpy and the temperature at these points are investigated as functions of time.

Now, thanks to the enthalpy reformulation, the problem appears identical to the plain heat conduction problem (with similar discretization equations). However, the relationship between temperature and enthalpy is now different, and it depends on the phase. The phase of a particular element is described by the value of the enthalpy assigned to the element. Thus, the elements for which the enthalpy is negative are solid; the elements for which the value of enthalpy lies between zero and λ are two-phase elements (i.e., the interface passes through such elements), and the elements for which the enthalpy is greater than λ are in the liquid phase. Once the phase of an element is determined, its temperature is set to:

1) $T = T_m + E/\rho C_s$ for solid elements

2) $T = T_m$ for two-phase elements

3) $T = T_m + (E-\rho\lambda)/\rho C_l$ for liquid elements

using relations (2.16).

A detailed analysis of the discretized form of the enthalpy equation is presented in Chapter 4 of this thesis.

STEFAN NUMBER AND INFLUENCE OF BOUNDARY CONDITIONS:

Next consider the boundary conditions and their effects on the phase change process. If T_{∞} is a characteristic external temperature, one driving force influencing the phase change could be the difference between this external temperature and the saturation temperature of the material. Therefore, the enthalpy difference $i_{ext} - i_{sat}$ may be considered as a characteristic enthalpy difference (as an alternative to the latent heat, λ).

The ratio of this characteristic enthalpy difference to the latent heat is known as the Stefan number . The Stefan number signifies the importance of the sensible heat relative to that of the latent heat. In low Stefan number problems, the heat liberated or absorbed at the moving interface during the phase change process is affected very little by changes in the sensible heat. For these problems, quasi-stationary approximations can be used to reduce the governing equations to the Laplace equation [3].

In high Stefan number problems, however, the sensible

heat plays an important role. Large Stefan number problems are usually encountered in the casting of metals with large temperature changes. In these problems the solidification time is only a fraction of the total casting time.

The value of the Stefan number can also influence the choice of a finite difference scheme, as will be seen in the next chapter.

CHAPTER 3

NUMERICAL PROCEDURE

As pointed out in Chapter 1 both explicit and implicit method have been used to obtain solutions for phase-change problems.

The explicit formulation of the finite difference representation provides relatively straightforward expressions for the determination of the unknown temperatures at given time intervals. In this method, fluxes are evaluated at the previous time step, so they are known. In other words, it is assumed that the values of the fluxes do not change appreciably during the time interval (tm,tm+1), and consequently the process up to time tm+1 is driven by the fluxes at time tm. The disadvantage of this method is that once the material properties and the step-size are fixed, there is a maximum

time-step beyond which the procedure becomes unstable. This maximum time step size makes the explicit method inappropriate for many phase-change problems.

In some cases the rate at which the material changes phase decreases as time goes on. For these cases, it might be desirable to gradually increase the time step-size in order to reduce the computational time. However, this may not be possible due to restrictions imposed by the numerical stability criterion.

In some thermal storage problems the process may require several hours to complete. Again, the use of the explicit scheme for these problems may result in a prohibitively large number of time steps.

Another class of problems for which use of the explicit representation may be not suitable is "low Stefan number problems". In non-dimensional form the stability criterion for an interior node is $\Delta\tau \leq Ste\, \frac{\Delta^2}{S}$ where $\Delta\tau$ and Δ represent the non-dimensional time step and the non-dimensional spatial step, respectively; and S = 2 ,4 and 8 for one, two, and three dimensional problems, respectively. Now, for two "low Stefan number" problems which are different only with respect to the Stefan number

(say Ste=0.1 and Ste=0.01), the above criterion requires time step values for the second problem that are one tenth of those for the first problem (the two problems may have nearly identical solutions[24]). In other words, the second problem has to be solved with ten times as many time steps as the first problem, and therefore, for these two similar problems the computational efforts differ by a factor of ten.

There are a number of unconditionally stable implicit methods suitable for solving the heat conduction equation. A common feature in implicit methods is the requirement to solve a set of simultaneous equations in order to obtain the results for each time step. Fortunately, the augmented matrices corresponding to these sets of equations have a large number of zero entries, and solution methods are available that take advantage of this fact.

For one-dimensional problems the tridiagonal procedure can be used to solve the set of equations efficiently, but this procedure is not applicable to multi-dimensional problems if a fully implicit difference scheme is used. According to Shamsundar [3] most of the "splitting" methods (such as the ADI method and the line-by-line method) make use of the tridiagonal equations

for multi-dimensional problems; however, the asymmetry resulting from the splitting of the Laplace difference operator (which is a characteristic of these methods) leads to poor accuracy for problems involving phase-change at a discrete temperature.

Another scheme that is used frequently to obtain numerical solutions for heat transfer problems is the Crank-Nicolson method. This method is also unconditionally stable, but if large values of time step are used, the results could show oscillatory behavior. The advantage of the Crank-Nicolson scheme over the explicit and fully implicit methods is that the resulting solution involves less truncation error for a given Δt than the other two methods. In fact, in the explicit and fully implicit methods the timewise truncation error is proportional to Δt, whereas the Crank-Nicolson method has an error proportional to $(\Delta t)^2$. On the other hand, the Crank-Nicolson scheme involves nearly twice as many operations as the fully implicit method, and hence, additional storage space and computer time are required.

There are several methods available for solving the simultaneous equations resulting from the application of the fully implicit method. Some of these techniques

include direct methods, iterative methods and semi-iterative methods.

Direct elimination methods, such as Gaussian elimination, are among the most widely used methods for numerically solving systems of simultaneous equations. However, these methods make no distinction between the zero and the non-zero coefficients in the solution procedure, and this leads to a large number of unnecessary operations when dealing with a set of equations obtained by the application of the fully implicit method. Another undesirable characteristic of these methods is the vast amount of storage space required. This is particularly important when a very large set of equations are to be solved simultaneously.

Iterative and semi-iterative techniques are usually the most efficient methods for solving a set of equations arising from a heat transfer problem. These methods have a very reasonable storage space requirement as they require storing of the unknown quantities for only two successive time levels.

The iterative methods (such as the Gauss-Seidel iterative technique) use the approximation for the unknown

quantity after n levels of iterations to come up with a better approximation for the quantity after n+1 iterations. In other words, the algorithm is independent of the number of iterations.

In semi-iterative methods, however, the value of the unknown quantity after n number of iterations is found by making use of the results of the previous iterations. Therefore, the algorithm varies from one iteration to next. Semi-iterative methods may lead to fewer numbers of iterations and faster convergence, but iterative schemes involve fewer calculations, and their algorithms are easier to apply. Furthermore, iterative methods have been used for a long time, and their theoretical aspects have been investigated thoroughly.

The numerical method used in this study consists of the fully implicit technique along with the Gauss-Seidel iterative scheme.

Details of the iterative scheme are illustrated in the next chapter, as it is applied to a sample problem.

The central difference scheme is employed for the spatial discretization . Since this representation

involves one node neighboring in each direction, a five-point difference scheme is required for two-dimensional problems.

CHAPTER 4

APPLICATION OF THE ENTHALPY METHOD TO AXI-SYMMETRIC SOLIDIFICATION PROBLEMS

In this chapter, the method developed in Chapter 2 and Chapter 3 is applied to one-dimensional and two-dimensional sample problems. The objective of this chapter is to check the validity of the enthalpy method by comparing the results to those obtained by a different numerical approach, and at the same time, illustrate the calculation procedure and the iterative scheme.

The first problem involves a circular cylindical container of radius R which is filled with the phase-change material initially in the liquid state and at its saturation temperature Tsat . At time t=0 the surface temperature is dropped to T_{∞} . The sudden drop in the

surface temperature causes a frozen layer to grow inwardly from the surfaces. The location of the interface and the temperature distribution in the resulting transient are to be computed at various time intervals until the solidification process is completed. This problem will also be solved using a convective boundary condition. The second problem is the extension of the first problem in two dimensions. Here,it should be mentioned that all the equations in this chapter are derived for the more general case of having two spatial dimensions, but they can easily be modified for one-dimensional problems.

DERIVATION OF GOVERNING FINITE DIFFERENCE EQUATIONS:

Since there is no variation of temperature with θ , one can make use of the axial symmetry and reduce the problem to a two-dimensional one.

The region needs to be divided into number of elements with a node placed at the center of each element. The scheme chosen is illustrated in Fig. 4.1 where the sides OA and OB are divided into n and m parts, and a finite difference net of rectangular mesh of size ($\Delta r, \Delta z$) is constructed over the region. Elements having nodes at their centers are constructed .

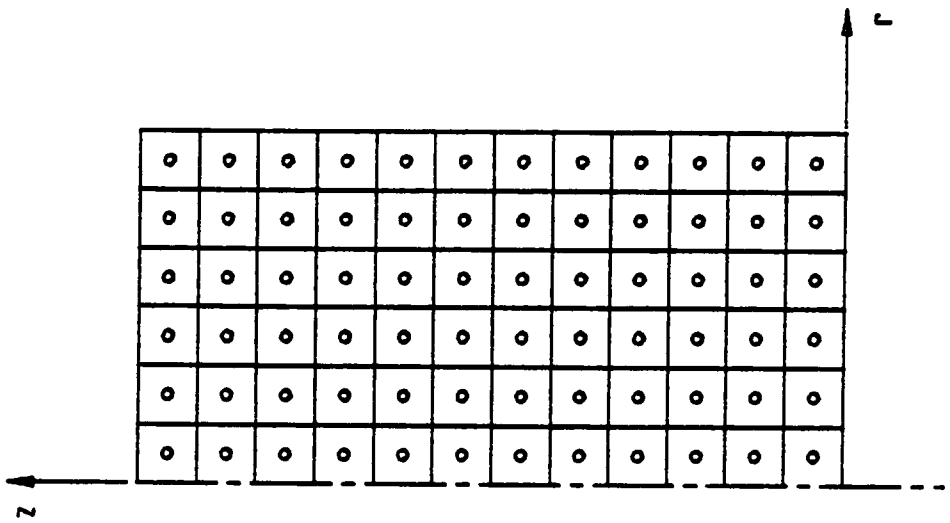


Figure 4.1 Rectangular Network of Mesh Dr, Dz

FINITE DIFFERENCE EQUATIONS:

The geometry of the problem suggests the use of a cylindrical coordinate system. In cylindrical coordinates the Laplacian $\nabla^2\Phi$ can be expressed as:

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial R^2} + \frac{1}{R}\left(\frac{\partial\phi}{\partial R}\right) + \left(\frac{1}{R^2}\right)\left(\frac{\partial^2\phi}{\partial\Theta^2}\right) + \frac{\partial^2\phi}{\partial Z^2} \qquad (4.1)$$

The terms in expression (4.1) are now replaced by their finite difference counterparts. In the following paragraphs the coordinates (R,Z) are replaced by:

$$R = i*\Delta R, \quad Z = j*\Delta Z \quad ,$$

where i and j are integers. Subscript pairs are also used to denote location. Hence: $T(R,Z) = T(i\Delta R, j\Delta Z) = T_{i,j}$.

Various derivatives in $\nabla^2\Phi$ at node (i,j) are represented in the finite difference form using central difference formulae:

$$\left.\frac{\partial^2\phi}{\partial R^2}\right|_{i,j} = \frac{\phi_{i-1,j} - 2\phi_{i,j} + \phi_{i+1,j}}{(\Delta R)^2} + O(\Delta R^2)$$

$$\left.\frac{\partial\phi}{\partial R}\right|_{i,j} = \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta R} + O(\Delta R^2) \qquad (4.2)$$

$$\left.\frac{\partial^2\phi}{\partial Z^2}\right|_{i,j} = \frac{\phi_{i,j-1} - 2\phi_{i,j} + \phi_{i,j+1}}{(\Delta Z^2)} + O(\Delta Z^2)$$

Substituting for various terms in expression (4.1) yields:

$$\nabla^2\phi\Big|_{i,j} = \frac{1}{\Delta R^2}\left[\left(1-\frac{1}{2i}\right)\phi_{i-1,j} - 2\phi_{i,j} + \left(1+\frac{1}{2i}\right)\phi_{i+1,j}\right] + \frac{1}{\Delta Z^2}\left(\phi_{i,j-1} - 2\phi_{i,j} + \phi_{i,j+1}\right) \tag{4.3}$$

In this sample problem the same spatial step size is used for both r and z. Therefore, since $\Delta R=\Delta Z$

$$\nabla^2\phi\Big|_{i,j} = \frac{1}{\Delta R^2}\left[\left(1-\frac{1}{2i}\right)\phi_{i-1,j} - 4\phi_{i,j} + \left(1+\frac{1}{2i}\right)\phi_{i+1,j}\right] + \phi_{i,j-1} + \phi_{i,j+1} \tag{4.4}$$

Next, the finite difference form corresponding to the left hand side of the enthalpy equation is obtained. Using the fully implicit technique (as described in Chapter 3) for a node (i,j) the time derivative is:

$$\frac{\partial\phi}{\partial Fo}\Big|_{i,j} = \frac{\theta^m - \theta^{m-1}}{\Delta Fo} \tag{4.5}$$

Note that in the above expression superscripts are used to denote time levels. Now, the finite difference representation of the enthalpy equation for an interior node can be written as

$$\frac{\theta^{m}-\theta^{m-1}}{\Delta Fo}=\frac{1}{\Delta R^{2}}\left[\left(1-\frac{1}{2i}\right)\phi^{m}_{i-1,j}-4\phi^{m}_{i,j}+\left(1+\frac{1}{2i}\right)\phi^{m}_{i+1}\right]+\frac{\phi^{m}_{i,j-1}+\phi^{m}_{i,j+1}}{\Delta R} \qquad (4.6)$$

This is the finite difference form of the enthalpy equation for non-zero values of R. As R approaches zero we have:

$$\lim_{R\to 0}\left[\frac{1}{R}\left(\frac{\partial\phi}{\partial R}\right)\right]=\frac{\partial^{2}\phi}{\partial R^{2}} \qquad (4.7)$$

So, at R=0 the Laplacian becomes:

$$\nabla^{2}\phi\Big|_{i,j}=2\left(\frac{\partial^{2}\phi}{\partial R^{2}}\right)+\left(\frac{\partial^{2}\phi}{\partial Z^{2}}\right) \qquad (4.8)$$

Thus, for a node (0,j) we have:

$$\nabla^{2}\phi\Big|_{0,j}=2\frac{\phi_{1,j}-2\phi_{0,j}+\phi_{1,j}}{(\Delta R)^{2}}+\frac{\phi_{0,j-1}-2\phi_{0,j}+\phi_{0,j+1}}{(\Delta Z)^{2}} \qquad (4.9)$$

and since $\Delta R=\Delta Z$ we get:

$$\nabla^{2}\phi\Big|_{0,j}=\frac{(4\phi_{1,j}+\phi_{0,j-1}+\phi_{0,j+1}-6\phi_{0,j})}{(\Delta R)^{2}} \qquad (4.10)$$

Thus, the finite difference form of the enthalpy equation at R=0 is:

$$\frac{\theta^{m} - \theta^{m-1}}{\Delta F 0} = \frac{\left(4\phi^{m}_{1,j} + \phi^{m}_{0,j-1} + \phi^{m}_{0,j+1} - 6\phi^{m}_{0,j}\right)}{(\Delta R)^{2}} \qquad (4.11)$$

The above difference equations along the with the initial condition and the θ vs. Ψ relationship (explained in Chapter 2) form a set of simultaneous algebraic equations for the unknown enthalpies and temperatures at a given time level.

THE METHOD OF CALCULATION:

The Gauss-Seidel iterative scheme is to carry out the computational procedure is described in the following paragraphs.

Suppose that the solution has been performed for m-1 time levels, and it is desired to obtain the solution for Fo=m* Δ Fo. The first step is to assign the initial approximations for the enthalpy variable, $\theta^{m}_{i,j}$. For this purpose, the value of $\theta^{m-1}_{i,j}$ obtained at the previous time level is set equal to $\theta^{m}_{i,j}$ (for all i,j). Then i and j are varied in a definite order to perform the first iteration

for the mth time level. To do this, equation (4.4) is rewritten in the following manner (note that for R=0 equation (4.11) should be used, but the tratment is similar.

$$\frac{(\Delta R)^2}{\Delta Fo}\theta^m_{i,j} + 4\phi^m_{i,j} = \frac{(\Delta R)^2}{\Delta Fo}\theta^{m-1}_{i,j} + \left(1-\frac{1}{2i}\right)\phi^m_{i-1,j} + \left(1+\frac{1}{2i}\right)\phi^m_{i+1,j} + \phi^m_{i,j-1} + \phi^m_{i,j+1} \qquad (4.6)'$$

There are two possibilities for $\theta^m_{i,j}$. If $\theta^m_{i,j}$ is negative, the element is in the solid phase. Subsequently, we have:

$$\theta^m_{i,j} = \phi^m_{i,j}$$

Hence, the left hand side of equation (4.6) becomes:

$$\left[\frac{(\Delta R)^2}{\Delta Fo}\right]\cdot\ \theta^m_{i,j}$$

On the other hand, if $0 < \theta^m_{i,j} \leq 1$, the element is a two-phase element. Therefore, for this element $\Phi^m_{i,j} = 0$, and the left hand side of equation (4.6) becomes:

$$\left[\frac{(\Delta R)^2}{\Delta Fo} + 4\right]\cdot\ \theta^m_{i,j}$$

Thus, for both cases the coefficient of $\theta^m_{i,j}$ is a positive

real number, and the sign of $\vartheta^{m}_{i,j}$ is the same as that of the right hand side of equation (4.6).

The right hand side of equation (4.6) is evaluated by knowing the most recent values of $\theta^{m}_{i,j}$ at the four nodes surrounding the node (i,j) and the appropriate θ vs. Φ relationship (note that $\theta^{m-1}_{i,j}$ is known from the previous time level).

Now, if the right hand side is negative, it means that $\theta^{m}_{i,j}$ is negative; hence, $\theta^{m}_{i,j}$ can be obtained by dividing the right hand side by $\frac{\Delta R^2}{\Delta Fo} + 4$. Similarly, if the right hand side is positive, it can be concluded that $\theta^{m}_{i,j}$ is positive , and it can be calculated by dividing the right hand side by $\Delta R^2/\Delta Fo$. Immediately the newly calculated value of $\theta^{m}_{i,j}$ is compared with the old value, and the deviation is recorded. Then, the new value of $\theta^{m}_{i,j}$ replaces the old value.

This procedure is carried out for all nodes, and deviations from old values are checked to see if a convergence criterion is met.

The foregoing method of solution is illustrated in figure 4.2.

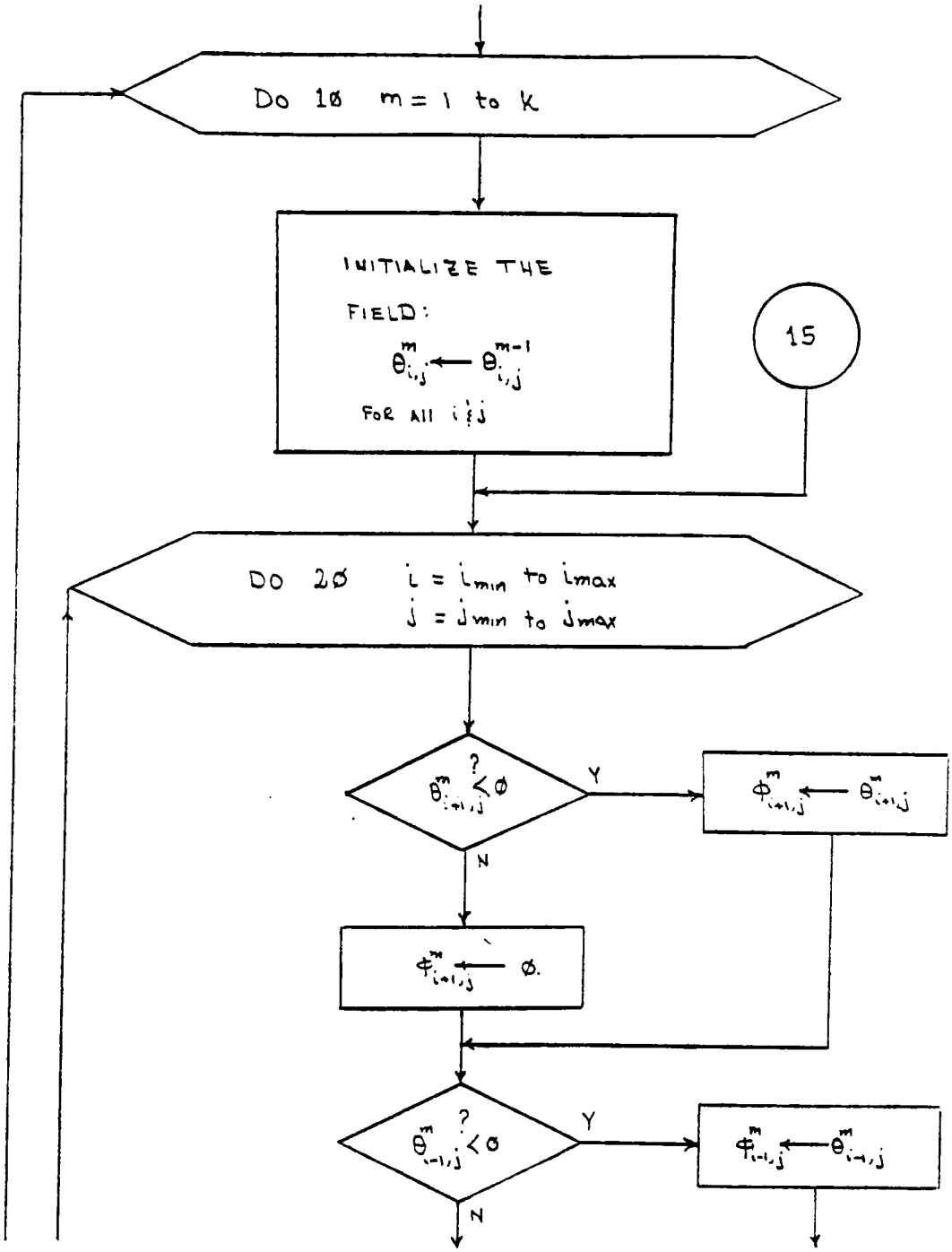


Figure 4.2 Flowchart of Calculation Procedure

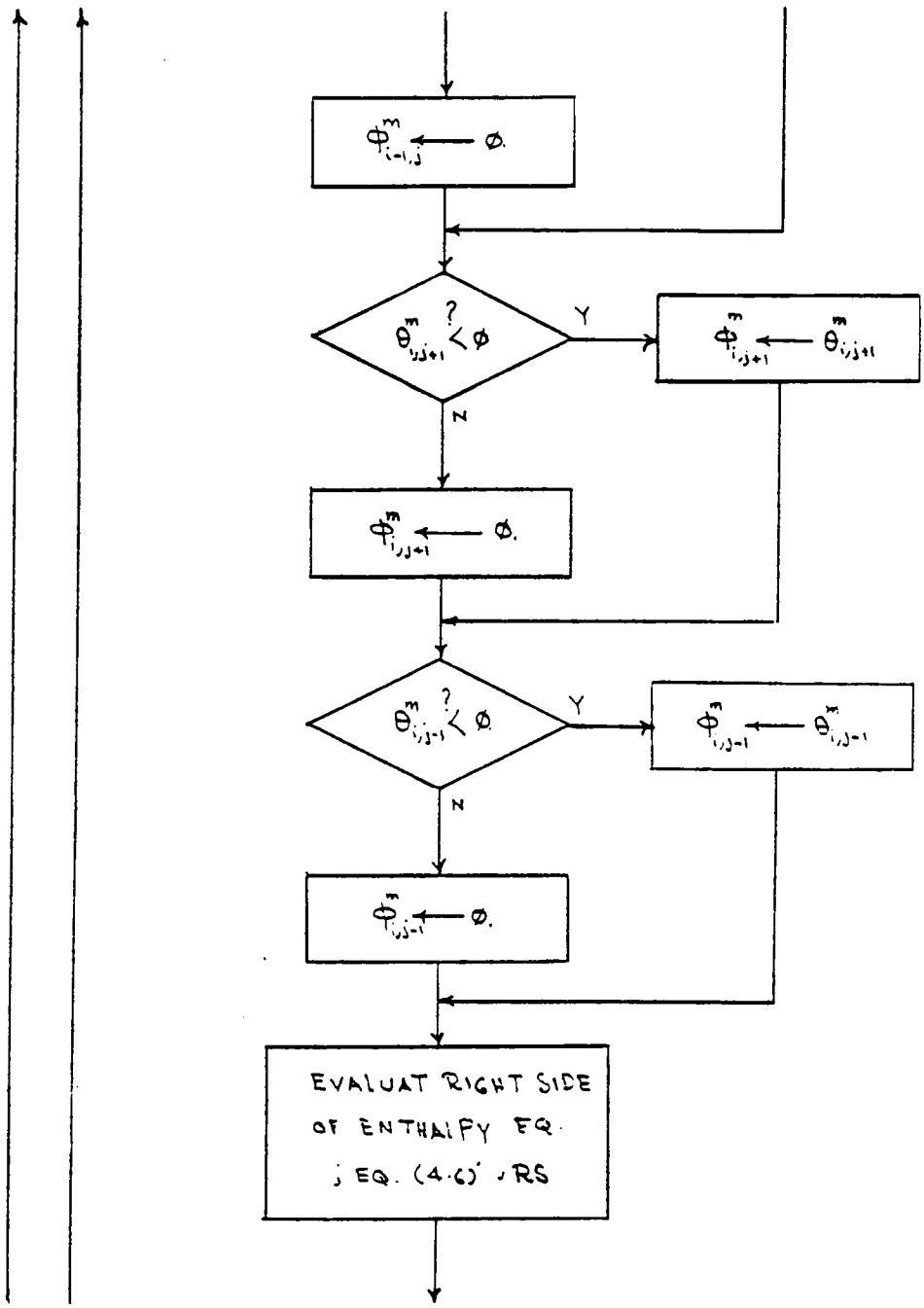


Figure 4.2 (Continued)

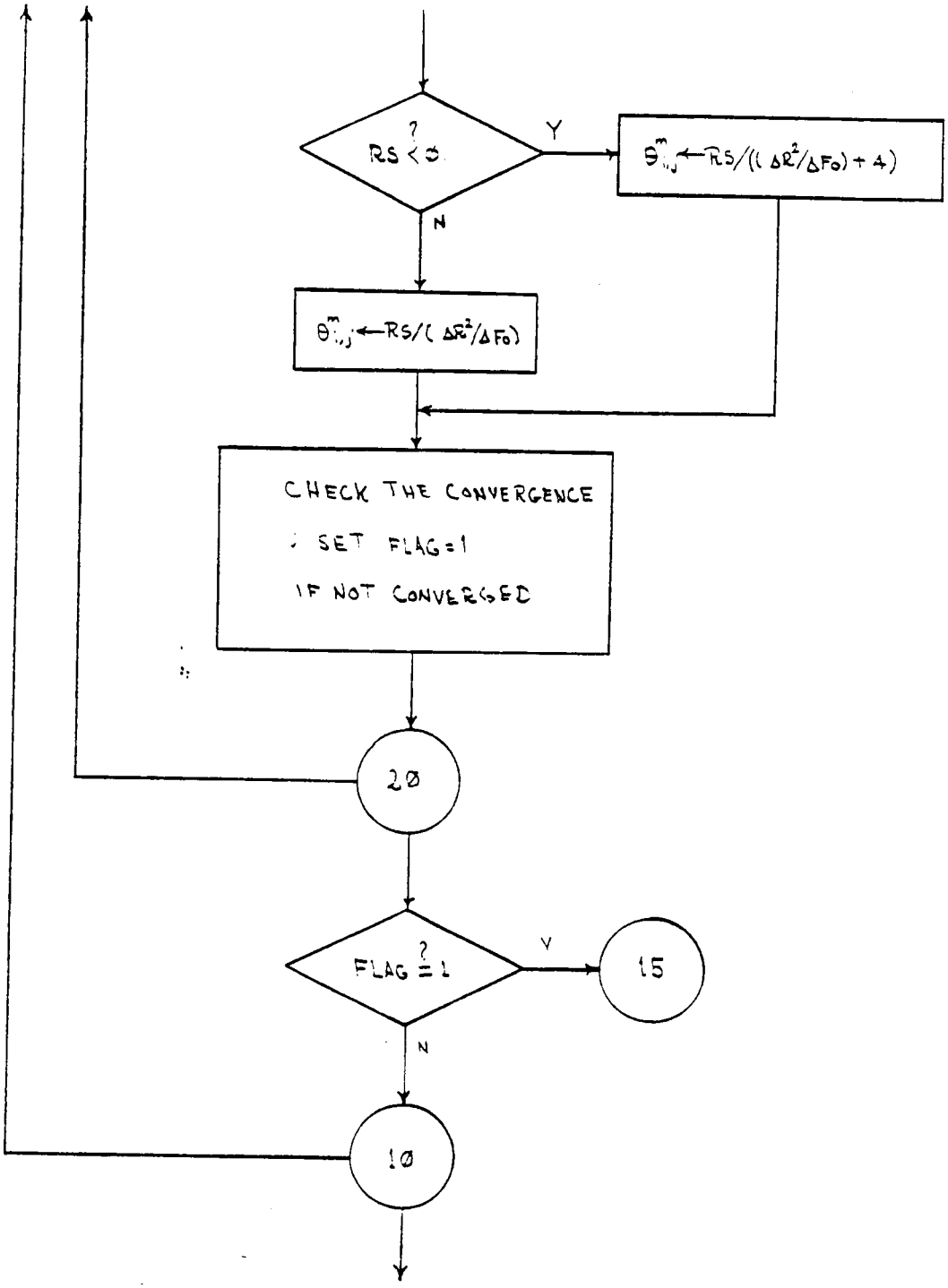


Figure 4.2 (Continued)

CALCULATION OF THE TEMPERATURE DISTRIBUTION AND THE INTERFACE LOCATION:

Using the iterative procedure described in the previous section, the values of the nondimesional enthalpy , θ ,can be obtained as a function of time for all nodes. In this section calculations of other important results, the temperature distribution, frozen fraction and the interface location are presented.

Temperature Distribution

The values of the temperatures at the nodal points can be readily obtained from the appropriate temperature-enthalpy relationship and the definition of θ as given in equation (2.21). Therefore, if Θ is negative (the element is in the solid phase) the nodal temperature is given by: $T = T_{sat} + \frac{\lambda\theta}{C_s}$ (note: for solid elements $\phi = \theta$). If $0 \leq \theta \leq 1$,it means that the element is a two-phase element, and the nodal temperature is given by $T = T_{sat}$.

Frozen Fraction

One of the indicators of the extent of the solidification of the phase-change material at a given

time is the frozen fraction defined as the ratio of the mass of the solid to the total mass of the material at any time.

The frozen fraction can be obtained by dividing the total mass of the solid elements plus the mass of the solid part of the two-phase elements by the total mass of the phase change material. The mass of the solid portion of the two-phase elements is given as: $Ms=1-\theta$.

Interface Location

Information that is often required in phase change problems is the location of the interface as a function of time. This result can be obtained by knowing the location of the two-phase elements (the elements for which $0<\theta<1$) and by knowing the value of the dimensionless enthalpy, θ , for these elements. Note that for two-phase elements θ is given by equation (2.19) and it represents the percentage of the material in the liquid state for that particular element (for example, $\theta=0.623$ means that 62.3 percent of this element is in the liquid phase). Therefore, an algorithm can be developed to provide the r and z coordinates of several points on the interface for each time step.

RESULTS AND DISCUSSION:

The calculation procedure described earlier was applied to the two sample problems. Some of the results are presented in the following paragraphs.

First, to compare results of the enthalpy method with known solutions, the method is applied to the one-dimensional problem involving the inward solidification of a long unit circular cylinder (0<r<1) containing a phase-change material initially in the saturated liquid state. The thermal conditions in this problem are T(r,0)=1., T(1,t)=0., $K_s=K_\ell=1.$, $C_s=C_\ell=1.$, $\lambda=1.$ and $\rho=1.$ Therefore, the boundary and initial conditions for the enthalpy method are :

$$I.C: \quad \begin{cases} \theta(R,0)=1. \\ \phi(R,0)=0. \end{cases} \qquad 0<R<1$$

$$B.C: \quad \phi(1,Fo)=-1. \qquad Fo>0$$

The calculations were performed using ΔFo=0.0025 and ΔR=0.1.

The results of interface position from this test problem are compared with those from Tao [20] in Figure 4.3. Tao's method is a temperature-based numerical scheme, requires a starting solution, and is only applicable to problems having initially saturated condition. The two solutions agree very closely, with a maximum difference of 7.0% occurring at Fo=0.35. Also, one can see that the speed of the solid-liquid interface decreases after the initial portion of the freezing period and stays almost constant for most of the solidification. However, the interface speed increases again shortly before the freezing process is complete.

Figure 4.4 shows the temperature as a function of radial position, R, at the instant the center freezes. The variation of temperature is almost linear with R (with the exception of the regions very close to the center and the external boundary).

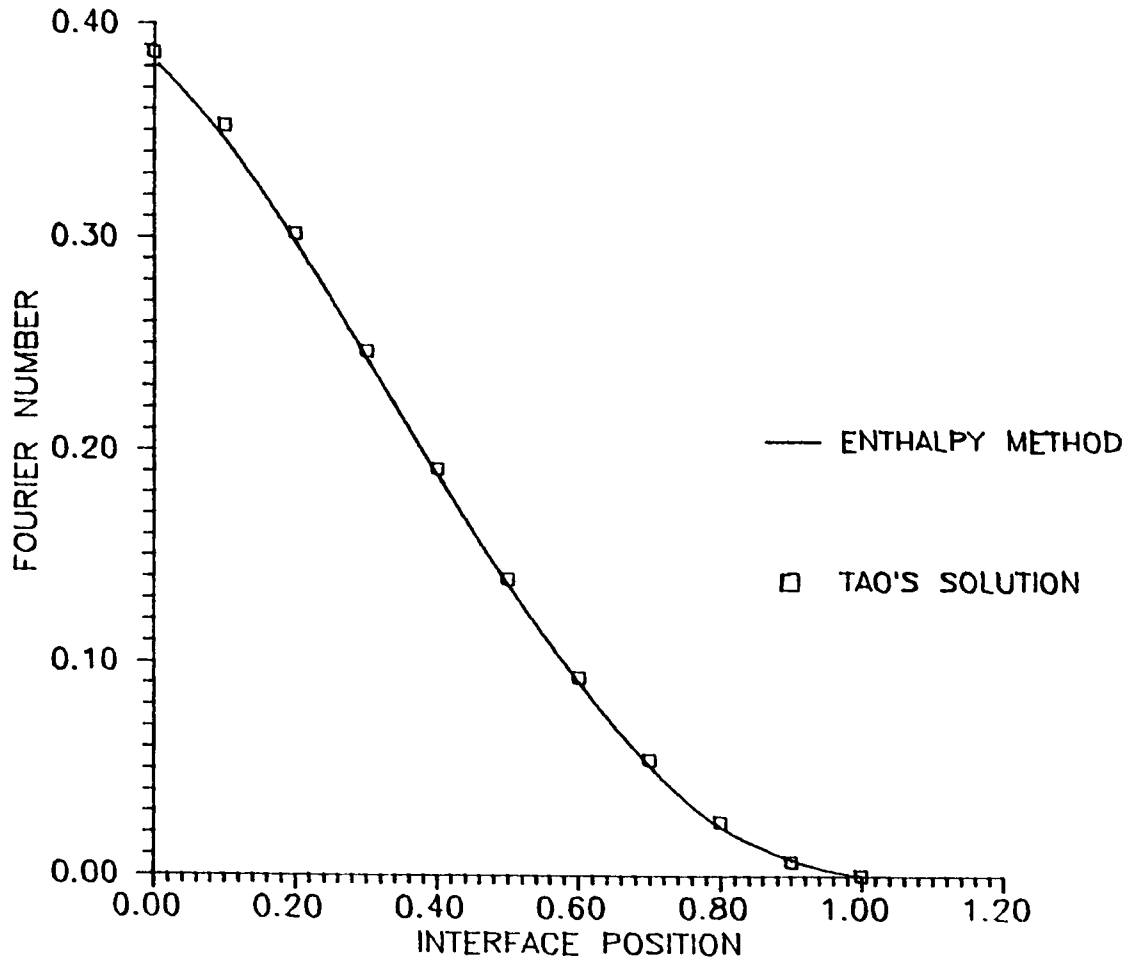


Figure 4.3 Interface Position for Infinitely Long Cylinder with Specified Boundary Temperature

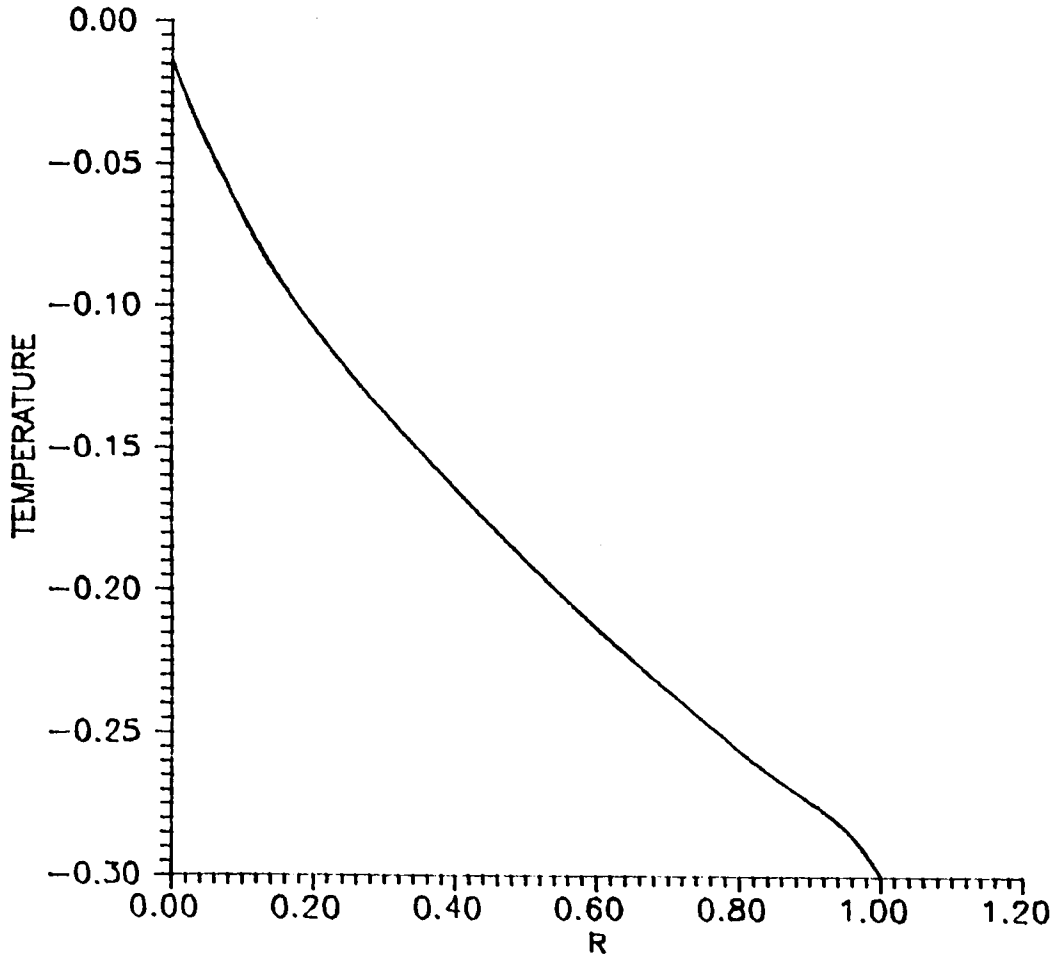


Figure 4.4 Temperature Distribution for Cylinder with Specified Boundary Temperature

THE EFFECTS OF CONVECTIVE BOUNDARY CONDITION:

In this section, results are presented for the same one-dimensional cylindrical problem already considered except that the outer surface is subjected to a convective boundary condition. The convective heat transfer is characterized by the surrounding fluid convective heat transfer coefficient, h. It is assumed that the heat transfer coefficient and the surrounding temperature, T_∞, are constant and uniform over the surface of the container. The results are shown for different values of the Biot number (Bi), hR/K .

In Figure 4.5 the radial position of the solid-liquid interface is plotted against the non-dimensional time for various values of Bi. It is evident from this figure that as the Biot number increases the curves approach the constant boundary temperature case. For very large Biot numbers (i.e. Bi=10000) the results are indistinguishable from those obtained using the constant temperature boundary condition. This behavior is expected since at a such high Biot numbers the external convective resistance is negligibly small when compared to the internal resistance to heat flow, and hence, the boundary

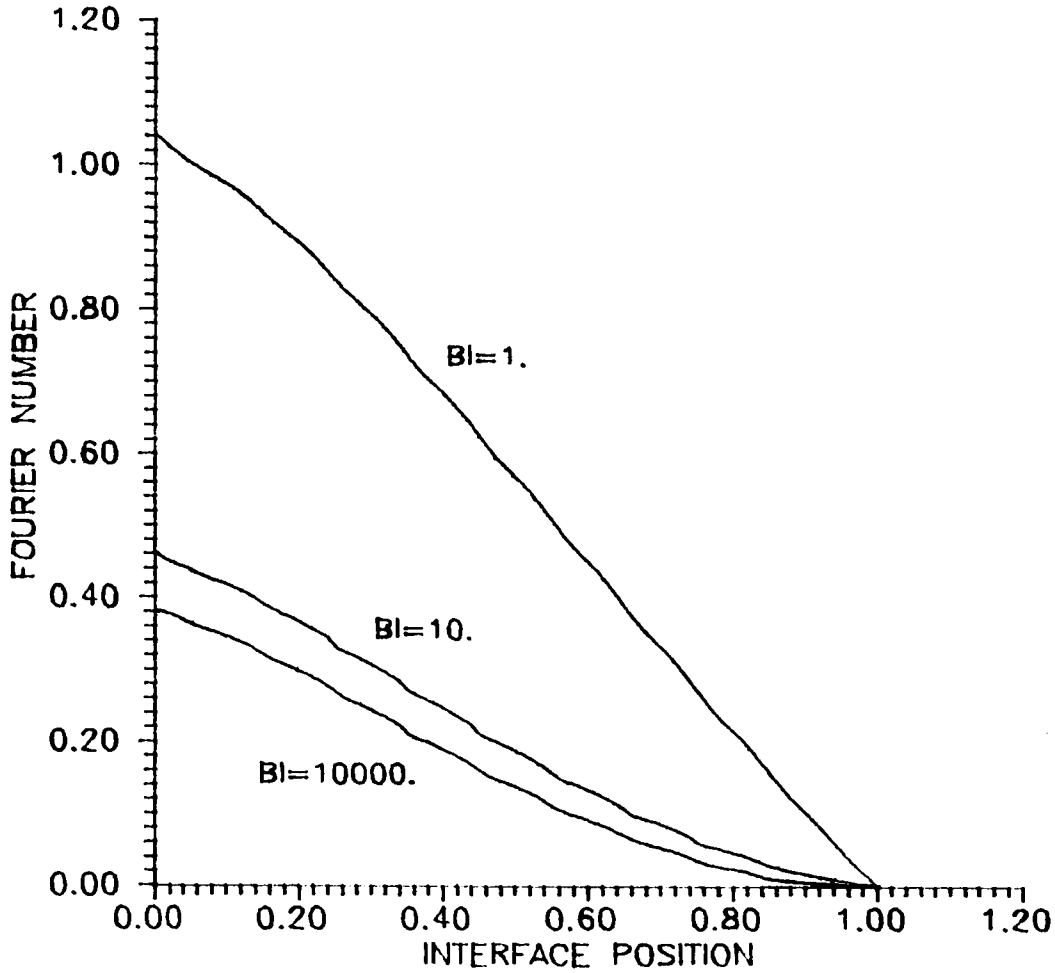


Figure 4.5 Interface Position for Infinitely Long Cylinder with Convective Boundary Condition

temperature will be essentially equal to the fluid temperature.

Figure 4.6 contains the temperature distribution curves for various values of the Fourier number. Again, in the solid region, the linear variation of temperature with the radial position can be seen from these curves.

RESULTS OF THE TWO-DIMENSIONAL PROBLEM:

The enthalpy method was also applied to a two-dimensional cylindrical problem in order to obtain a preliminary computer code for the axi-symmetric problem to be introduced in Chapter 5. The two-dimensional case differs from the previously-presented one-dimensional cases in that the ends of the cylinder are subjected to the given boundary conditions, and the cylinder is of finite length.

The interface locations for this problem are shown in Figure 4.7 at various times. The results are presented for the case of a specified boundary temperature. The thermal properties used here are identical to those used in the previous one-dimensional problem.

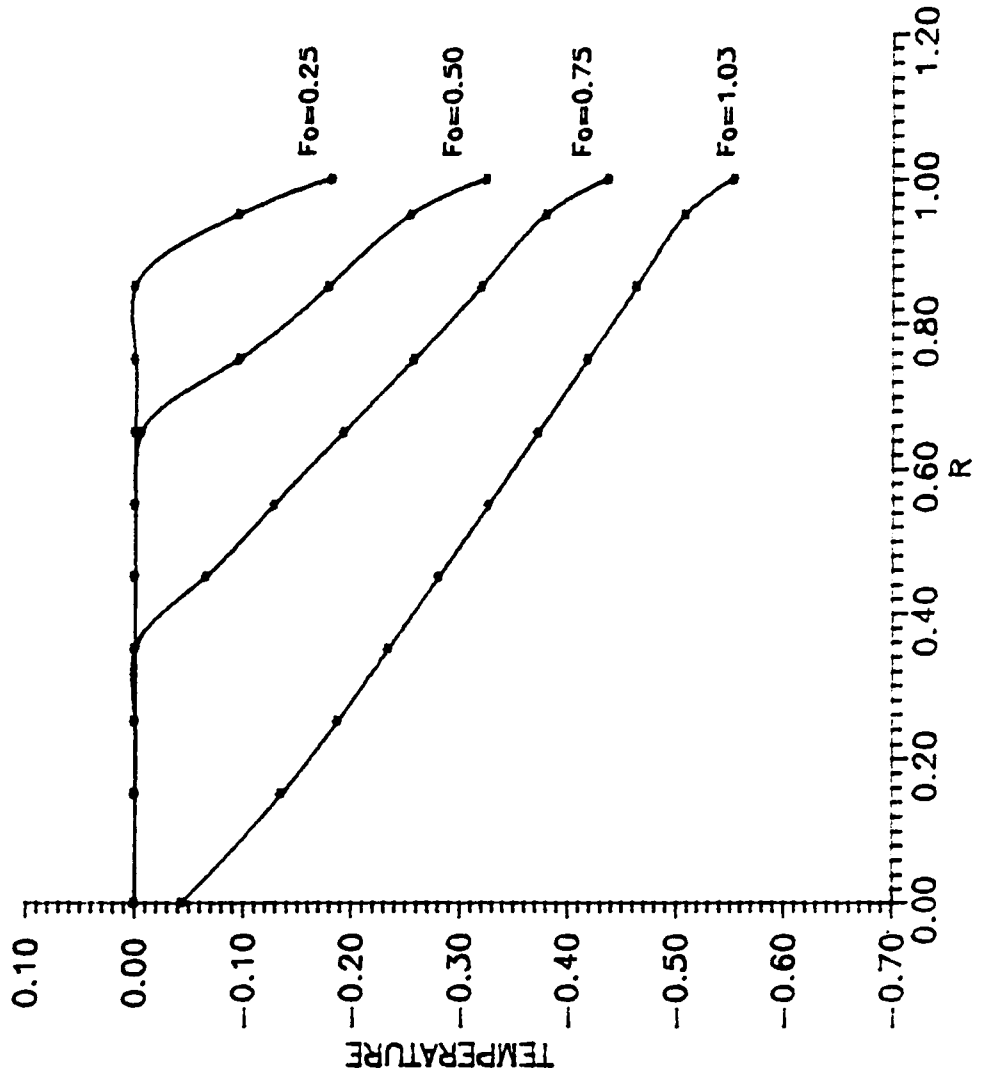


Figure 4.6 Temperature Distribution for Cylinder with Convective Boundary Condition

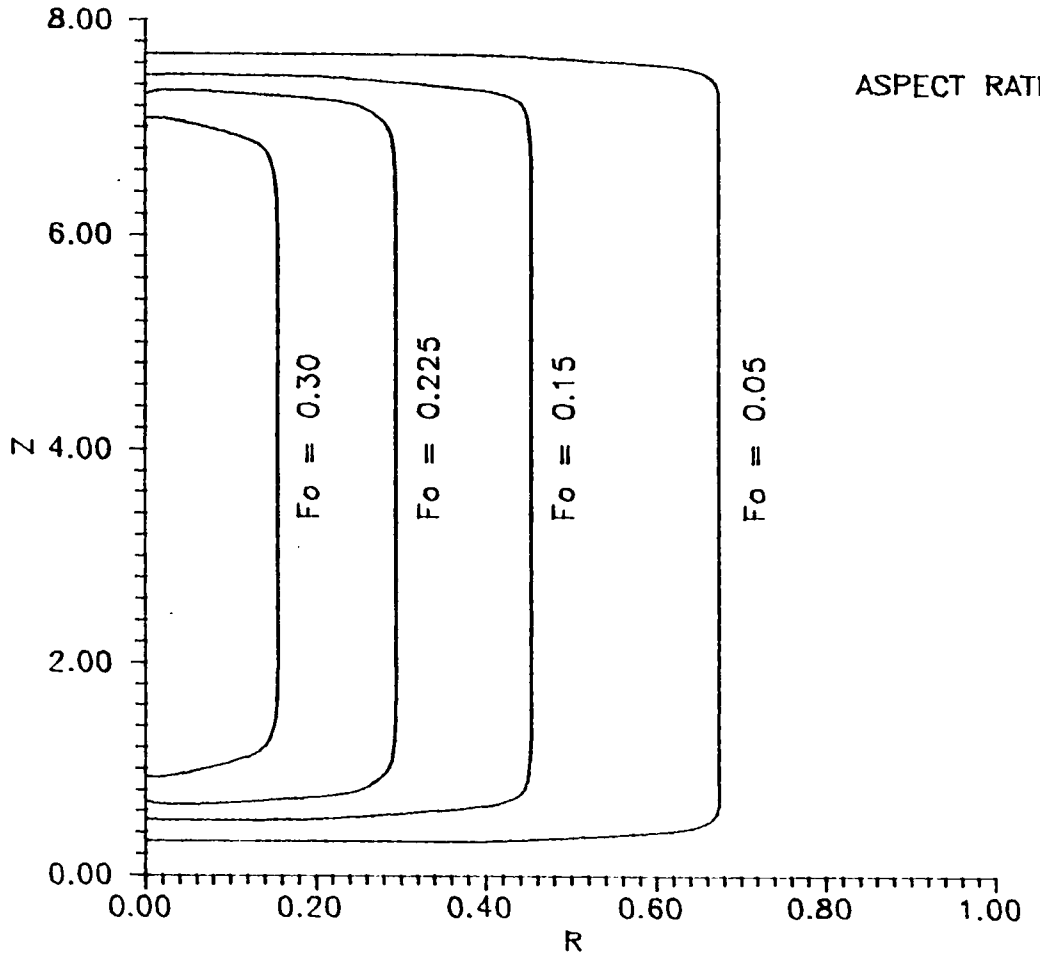


Figure 4.7 Interface Position for Phase-Change in a Circular Cylinder

In Figure 4.8 the curve of frozen fraction, FFCT , is plotted against time. This figure indicates that the freezing rate is highest at the start of the process, and then decreases as time goes on. The time corresponding to FFCT=1.0 is the total freezing time. Note that the total freezing time for this problem is very close to that of the first problem (at Fo=0.38). This result is to be expected since the ratio of the length to the radius used in this particular problem is eight, and therefore, the ends of the cylinder have only a limited effect on the total freezing time .

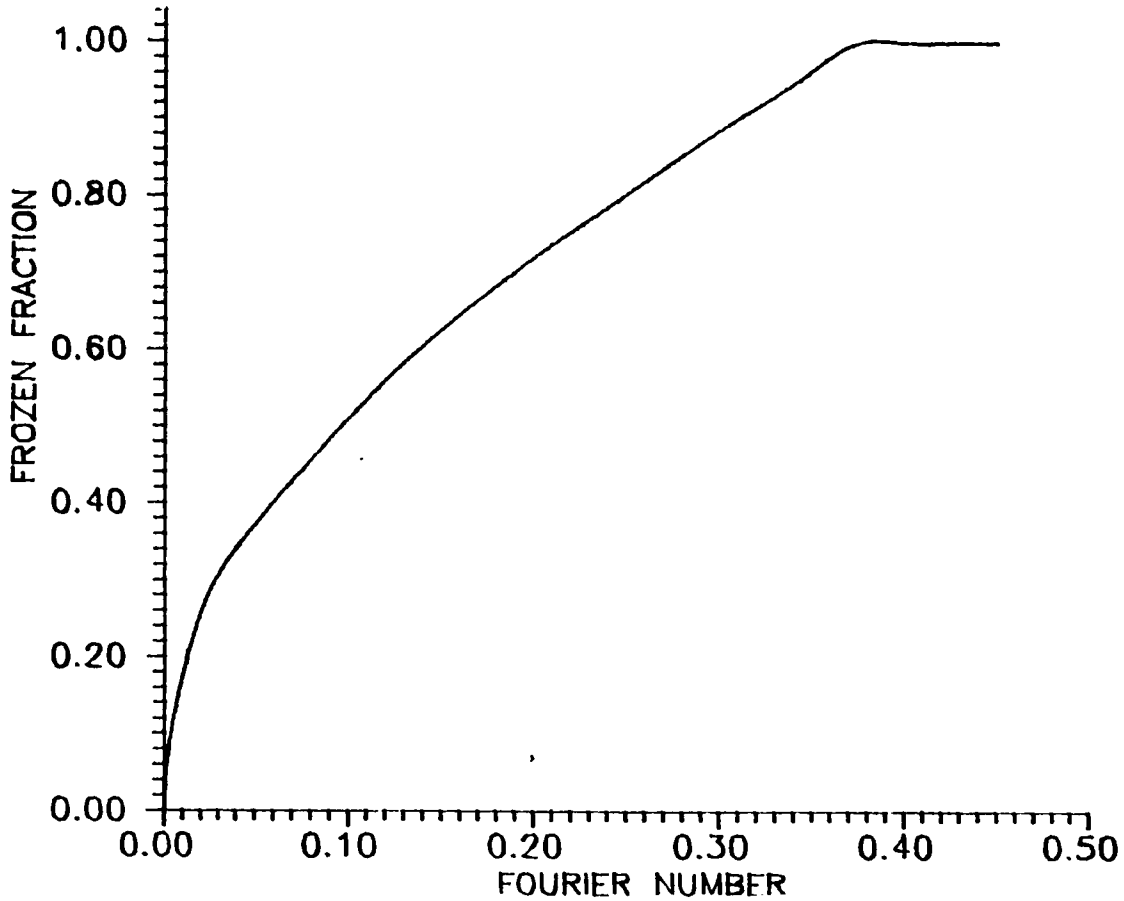


Figure 4.8 Frozen Fraction for a Cylinder with Specified Boundary Temperature

CHAPTER 5

AN AXI-SYMMETRIC FREEZING PROBLEM IN APPROXIMATE TEAR-DROP SHAPE GEOMETRY

In this chapter the method developed in Chapter 4 is applied to a multi-dimensional solidification problem in which there is a special interest in the position of the interface as a function of time. The impetus for this study resulted from the practical problem of mass producing cast candles.

The problem involves a tear-drop shaped container filled with the phase change material (wax) initially in the liquid state. The tear-drop geometry is approximated by a sphere of radius R attached to a cylinder of radius Rcy and height H (see Figure 5.1). At time t=0 the walls of the container experience a sudden drop of the temperature. As a result of this a frozen layer is formed, and it grows as time passes. The temperature distribution

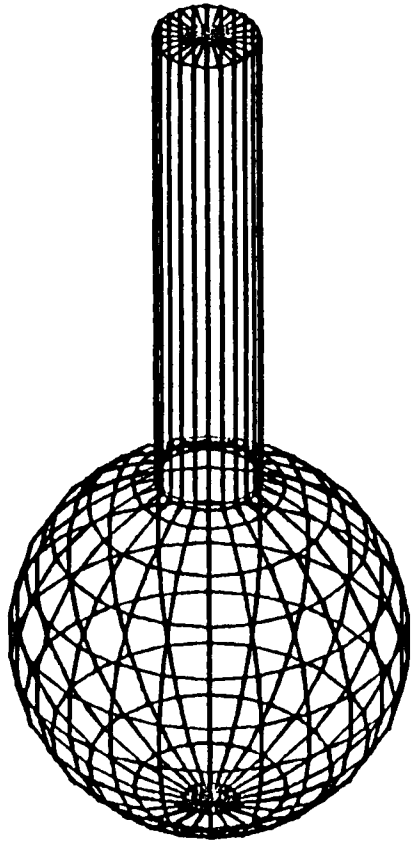


Figure 5.1 Approximate Tear-Drop Container

and the interface location are to be evaluated at various time intervals for the duration of the solidification process.

GRID AND CONTROL VOLUMES:

A rectangular mesh of size ($\Delta r, \Delta z$) is constructed over the region. The calculation domain is divided into a number of control volumes (the dashed lines denote the control volume boundaries), and grid points are placed at the centers of the control volumes . Grid points are placed on the external boundaries (with the exception of the curved boundary); consequently, the boundary control volumes are half control volumes.

DEFINITION OF SOME TERMS:

Before the actual calculation procedure is presented, it is helpful to define some terms encountered frequently in the next few sections.

Symbols Rcy and R are used to denote the radius of the cylindrical section and the radius of the sphere, respectively.

Rmax(j) is the maximum r for a given z. Rmax(j) is constant in the cylinder, but in the sphere it is a function of z and is given by:

$$R_{Max}(j) = \sqrt{R^2 - (R - Z)^2} \tag{5.1}$$

Rl is the local radius. For a given z (or j), Rl varies from zero to Rmax(j). It is evaluated by:

$$R_L = i \times \Delta r \tag{5.2}$$

LEN is the height of the spherical section. It can be obtained from:

$$LEN = R + \sqrt{R^2 + R_{Cy}^2} \tag{5.3}$$

NVI(j) denotes the number of interior nodes in the r direction for a given z. NVI(j) is a function of z and is calculated as:

$$NVI(j) = INT\left(\frac{R_{Max}(j)}{\Delta y}\right) \tag{5.4}$$

Prior to the start of the computation, the values of Rmax(j) and NVI(j) are calculated for all values of z=j z (j=0,nn) and stored in two arrays for later use.

FINITE DIFFERENCE EQUATIONS:

Even though the calculation procedure is similar to the one described in Chapter 4; here, several equations must be employed to handle different possible situations. The finite difference equations used in this chapter are derived for the general case of Rn (the non-dimensional ΔR) not being equal to Zn (the non-dimensional Δ Z). All of these equations along with their regions of applicability are introduced in the next few paragraphs. The derivation of these equations are shown in Appendix A.

For nodes located on the axis of symmetry (r=0) the governing finite difference equation is:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^{m}_{0,j} + \left[4. + 2.\left(\frac{RN}{ZN}\right)^2\right].\ \phi^{m}_{0,j} = \left(\frac{RN^2}{\Delta Fo}\right)\theta^{m-1}_{0,j} + 4.\ \phi^{m}_{1,j} + \left(\frac{RN}{ZN}\right)^2\left(\phi^{m}_{0,j-1} + \phi^{m}_{0,j+1}\right) \qquad (5.5)$$

The equation used for the regular interior nodes (nodes that have four neighboring grid points with

distances Δr and Δz apart from them) is:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^{m}_{i,j} + \left[2. + 2.\left(\frac{RN}{ZN}\right)^2\right].\ \phi^{m}_{i,j} = \left(\frac{RN^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} + \left(1 - \frac{1}{2i}\right)\phi^{m}_{i,j-1} + \left(1 + \frac{1}{2i}\right)\phi^{m}_{i+1,j} + \left(\frac{RN}{ZN}\right)^2\left(\phi^{m}_{i,j-1} + \phi^{m}_{0ij+1}\right) \qquad (5.6)$$

Note that this equation reduces to equation (4.6) if we let Rn be equal to Zn.

For nodes near the curved boundary it is possible that one, two or three strings are crossed by the curved boundary;therefore, equation (5.6) should be modified correspondingly.

First, consider the case in which the Δr string is intersected by the curved boundary. Here, ξr ($0<\xi<1$) is the distance between the grid point and the curved boundary, and ϕ_B represents the non-dimensional temperature at the boundary. The finite difference form of the enthalpy equation appropriate for this case is:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^{m}_{i,j} + \left[\frac{2.}{\xi}\left(1 + \frac{1-\xi}{2i}\right) + 2.\left(\frac{RN}{ZN}\right)^2\right].\ \phi^{m}_{i,j} = \left(\frac{RN}{\Delta Fo}\right)^2\theta^{m-1}_{i,j} + \left[\frac{2}{\xi(1+\xi)}\right]\left(1 + \frac{1}{2i}\right)\phi^{m}_{B} + \left(\frac{2}{1+\xi}\right)\left(1 - \frac{\xi}{2i}\right)\phi^{m}_{i-1,j} + \left(\frac{RN}{ZN}\right)^2\left(\phi^{m}_{i,j-1} + \phi^{m}_{i,j+1}\right) \qquad (5.7)$$

Another possibility is that the lower Δz string is intersected by the curved boundary. Using $\eta_1 \Delta z$ $(0<\eta_1<1)$ as the distance between the node and the boundary, the equation suitable for this case can be written as:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^m_{i,j} + 2.\left[1+\frac{1}{\eta_1}\left(\frac{RN}{ZN}\right)^2\right].\ \phi^m_{i,j} = \left(\frac{RN^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} + \left(1-\frac{1}{2i}\right)\phi^m_{i-1,j} + \left(1+\frac{1}{2i}\right)\phi^m_{i+1,j} + 2.\left(\frac{RN}{ZN}\right)^2\left[\frac{\phi^m_B}{\eta_1(1+\eta_1)} + \frac{\phi^m_{i+1,j}}{(1+\eta_1)}\right] \qquad (5.8)$$

If the upper Δz is intersected by the curved boundary, the finite difference equation for this node will be:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^m_{i,j} + 2.\left[1+\frac{1}{\eta_2}\left(\frac{RN}{ZN}\right)^2\right].\ \phi^m_{i,j} = \left(\frac{RN^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} + \left(1-\frac{1}{2i}\right)\phi^m_{i-1,j} + \left(1+\frac{1}{2i}\right)\phi^m_{i+1,j} + 2.\left(\frac{RN}{ZN}\right)^2\left[\frac{\phi^m_B}{\eta_2(1+\eta_2)} + \frac{\phi^m_{i-1,j}}{(1+\eta_2)}\right] \qquad (5.9)$$

where $\eta_2 \Delta z$ $(0<\eta_2<1)$ is the distance between the grid point and the point at which the curved boundary crosses the string.

It is also possible that one Δz string and one Δr string are intersected by the curved boundary. For this case, the finite difference equation becomes:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^{m}_{i,j} + 2.\left[\frac{1}{\xi}\left(1+\frac{1-\xi}{2i}\right)+\frac{1}{\eta}\left(\frac{RN}{ZN}\right)^2\right].\ \phi^{m}_{i,j} = \left(\frac{RN^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} + \left(\frac{2}{1+\xi}\right)\left(1-\frac{\xi}{2i}\right)\phi^{m}_{i-1,j} + \left[\frac{2}{\xi(1+\xi)}\right]\left(1+\frac{1}{2i}\right)\phi^{m}_{i+1,j} + 2.\left(\frac{RN}{ZN}\right)^2\left[\frac{\phi^{m}_{B}}{\eta(1+\eta)}+\frac{\phi^{m}_{i+1,j}}{(1+\eta)}\right] \qquad (5.10)$$

The final case that should be considered is when two Δz strings and one r string are intersected by the curved boundary. The appropriate form of the finite difference equation for this situation is:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^{m}_{i,j} + 2.\left[\frac{1}{\xi}\left(1+\frac{1-\xi}{2i}\right)+\frac{1}{\eta_1\eta}\left(\frac{RN}{ZN}\right)^2\right].\ \phi^{m}_{i,j} = \left(\frac{RN^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} + \left(\frac{2}{1+\xi}\right)\left(1-\frac{\xi}{2i}\right)\phi^{m}_{i-1,j} + \left[\frac{2}{\xi(1+\xi)}\right]\left(1+\frac{1}{2i}\right)\phi^{m}_{i+1,j} + 2.\left(\frac{RN}{ZN}\right)^2\left[\frac{\phi^{m}_{B}}{\eta_1\eta_2}\right] \qquad (5.11)$$

Note that in equations (5.7) through (5.11),if we let ξ, η_1 and η_2 be equal to one, the resulting equation would be identical to equation (5.6), as expected.

DETAILS OF CALCULATION:

For every iteration at a given time level, calculations are performed by sweeping first in the r direction (changing i from zero to NVI(j) while keeping j constant) and then in the z direction by incrementing j from zero to nn keeping i constant.

The calculation procedure is similar to the one described in Chapter 4.

Depending on where the main node is located, one of equations (5.5)-(5.11) is chosen. Then, the right hand side is calculated using the most recent information on Φ's. If the right hand side is negative, $\theta^{m}_{i,j}$ is obtained by dividing the right hand side by (Rn**2/ Δ Fo). Alternatively, if the right hand side is positive, $\theta^{m}_{i,j}$ is found by dividing the right hand side by (Rn**2/ΔFo) plus the term that multiplies $\Phi^{m}_{i,j}$. For more information on the treatment of the finite difference equations the reader should refer to the "calculation procedure section" in Chapter 4.

Equation (5.5) is used for all the nodes on the axis of symmetry (nodes for which i=0). For all other grid

points one of equations (5.6) through (5.11) must be used. In order to decide which one of these equations should be used for a given grid point, it is necessary to determine the type of the node (i.e., whether it is a regular interior node or it has one, two or three strings intersected by the curved boundary).

If for a certain node (i,j), the local radius, Rl, is larger than the maximum radius of the previous level, Rmax(j-1), the lower z string of this node is intersected by the curved boundary. In this case, η_1 can be obtained from:

$$\eta_1 = \frac{j.\Delta Z - \left(R - \sqrt{R^2 - R_L^2}\right)}{\Delta Z} \qquad (5.12)$$

Similarly, if for node (i,j) Rl is greater than Rmax(j+1), then the upper Δ z string is crossed by the curved boundary. Here, η_2 is evaluated as:

$$\eta_2 = \frac{\left(R - \sqrt{R^2 - R_L^2}\right) - j.\Delta Z}{\Delta Z} \qquad (5.13)$$

The only nodes that can have their Δ r string intersected by the curved boundary are those for which

i=NVI(j).

Now, a scheme is required to systematically examine all the nodes and to identify the appropriate form of the finite difference equation to be used for each node. To accomplish this, a general governing finite difference equation is introduced for all grid-points (for which i=0) as:

$$\left(\frac{RN}{\Delta Fo}^2\right)\theta_{i,j}^{m} + 2.\left[\frac{1}{\xi}\left(1+\frac{1-\xi}{2i}\right)+ CF_1\left(\frac{RN}{ZN}\right)^2\right]\cdot\ \phi_{i,j}^{m} = \left(\frac{RN^2}{\Delta Fo}\right)\theta_{i,j}^{m-1} +$$
$$\left[\frac{2}{\xi(1+\xi)}\right]\left(1+\frac{1}{2i}\right)\phi_{i+1,j}^{m} + \left(\frac{2}{1+\xi}\right)\left(1-\frac{\xi}{2i}\right)\phi_{i-1,j}^{m} +$$
$$\left(\frac{RN}{ZN}\right)^2\left(CF_2\phi_{i,j-1}^{m}+ CF_3\phi_{i,j+1}^{m}\right) \qquad (5.15)$$

Note that for elements in which the Δr string is not crossed by the curved boundary ($0<i<NVI(j)$), ξ is equal to one, and the general equation becomes:

$$\left(\frac{RN}{\Delta Fo}^2\right)\theta_{i,j}^{m} + 2.\left[1+ CF_1\left(\frac{RN}{ZN}\right)^2\right]\phi_{i,j}^{m} = \left(\frac{RN^2}{\Delta Fo}\right)\theta_{i,j}^{m-1} +$$
$$\left(1+\frac{1}{2i}\right)\phi_{i+1,j}^{m} + \left(1-\frac{1}{2i}\right)\phi_{i-1,j}^{m} +$$
$$\left(\frac{RN}{ZN}\right)^2\left(CF_2\phi_{i,j-1}^{m}+ CF_3\phi_{i,j+1}^{m}\right) \qquad (5.16)$$

In equation (5.15) and (5.16) CF1, CF2 and CF3 are coefficients that determine whether upper, lower, both or none of the Δz string are intersected by the curved boundary. For most of the interior elements, none of the Δz strings are intersected by the curved boundary. In this case

$$\begin{cases} CF_1 = 1. \\ CF_2 = 1. \\ CF_3 = 1. \end{cases} \qquad (a)$$

When the lower Δz string is intersected by the curved boundary, these coefficients are:

$$\begin{cases} CF_1 = \dfrac{1.}{\eta_1} \\ CF_2 = \dfrac{2}{\eta_1(1+\eta_1)} \\ CF_3 = \dfrac{2}{(1+\eta_1)} \end{cases} \qquad (b)$$

Alternatively, when the upper Δz string is crossed by the curved boundary,

$$\begin{cases} CF_1 = \dfrac{1.}{\eta_2} \\ CF_3 = \dfrac{2}{\eta_2(1+\eta_2)} \\ CF_2 = \dfrac{2}{(1+\eta_2)} \end{cases} \qquad (c)$$

Finally, when both of the Δz strings are crossed by the curved boundary, the following relationship must be used:

$$\begin{cases} CF_1 = \dfrac{1}{\eta_1 \eta_2} \\ CF_2 = \dfrac{2}{\eta_1 \eta_2} \\ CF_3 = \dfrac{2}{\eta_1 \eta_2} \end{cases} \qquad (d)$$

Thus, the algorithm should be such that all the possibilities (for nodes(i,j) , i=0) must be checked, and for each element a set of coefficients (CF1, CF2 and CF3) must be evaluated using relations (a) through (d). Next, these sets of coefficients are introduced in the governing equation (eq. (5.15)) and the resulting equation is solved for the non-dimensional enthalpy at the grid-point using the method described earlier. Then, the procedure goes to the next grid-point by incrementing i, and the entire procedure is repeated.

This procedure can be seen best from Figure 5.2. Note that this flowchart is for one iteration in the z direction.

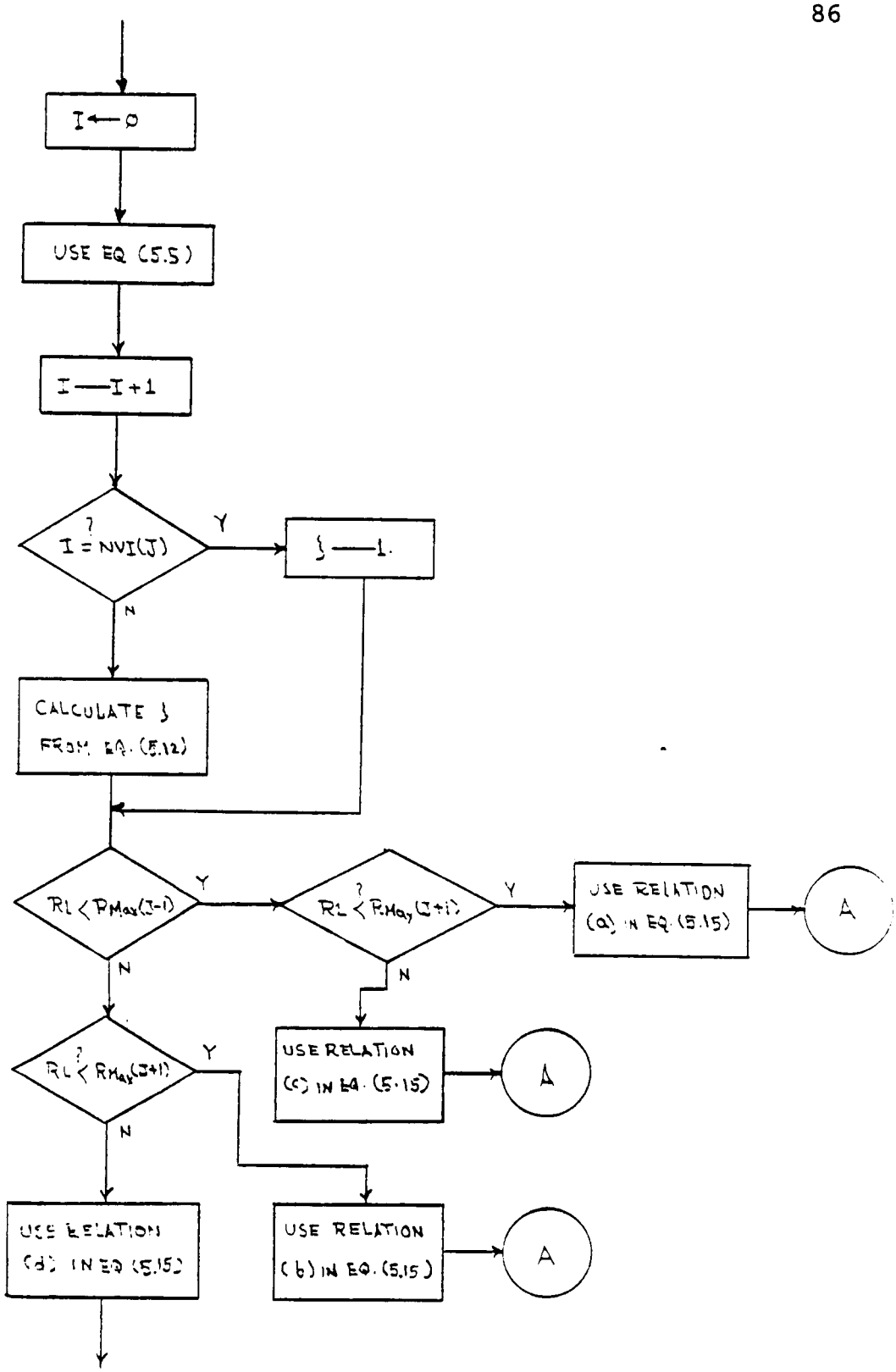


Figure 5.2 Flowchart of Calculations for General Axi-Symmetric Case

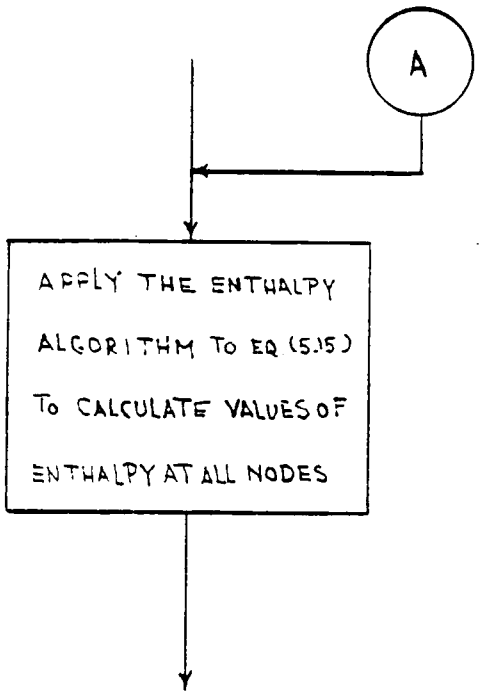


Figure 5.2 (Continued)

RESULTS AND DISCUSSION:

The procedure described in the previous section was applied to the problem to obtain a set of results for the interface location and the temperature distribution.

Prior to running the final cases a number of numerical experiments were performed to examine some aspects of the solution method. The first aspect that was studied was the convergence criterion. The convergence criterion for the set of non-linear algebraic equations was based on the maximum deviation from the previous iteration. The value of 0.0005 was used for the maximum allowable difference between the values of the non-dimensional enthalpy at any node for two successive iterations at a certain time level. Noting that the non-dimesional enthalpy is of the order one, the value of 0.0005 leads to high degrees of accuracy. With this criterion, the convergence was obtained with less than twenty five iterations for any time level (the value of 25 was used as the maximum allowable number of iterations for any time level).

The effect of time step size was another aspect of the solution method that was examined. To insure that the step

size (DFo) was not a factor in the accuracy of the results, runs were repeated with decreasing values of time steps. The value of DFo were considered small enough when the results of the final solidification time for two successive runs were within one percent of each other.

The final aspect that was examined was the effect of the spatial step sizes DR and DZ. This was performed in two stages. In the first stage, the uniform grid case (constant DR and DZ) was studied. Numerical runs were repeated with halved step-sizes until the results of two successive runs agreed to two significant digits. Once the base grid was chosen, the effect of employing a non-uniform grid was examined (stage 2 of these experiments). The results of these experiments were very close to the ones obtained using the base uniform grid (i.e, the final solidification times were identical and the results of each time step agreed to two to three significant digits).

RESULTS OF INTERFACE LOCATION:

Once again, the position of the solid-liquid interface can be found by locating the two-phase elements and employing the procedure described in Chapter 4 (Note: This procedure was performed inside the computer program after the convergence achieved).

In Figure 5.3 the position of the interface on the line corresponding to the maximum radius of the sphere is plotted against the dimensionless time. From this graph it is evident that the speed of the interface decreases after the initial solidification period and stays almost constant for most of the freezing process. However, just before the completion of the solidification the interface speed increases again.

Figure 5.4 contains interface location curves at various values of the Fourier number. Note that these curves are drawn for the vertical plane passing through the center of the sphere; the actual shape of the interface can be imagined by rotating these curves around the axis of symmetry (the Z axis).

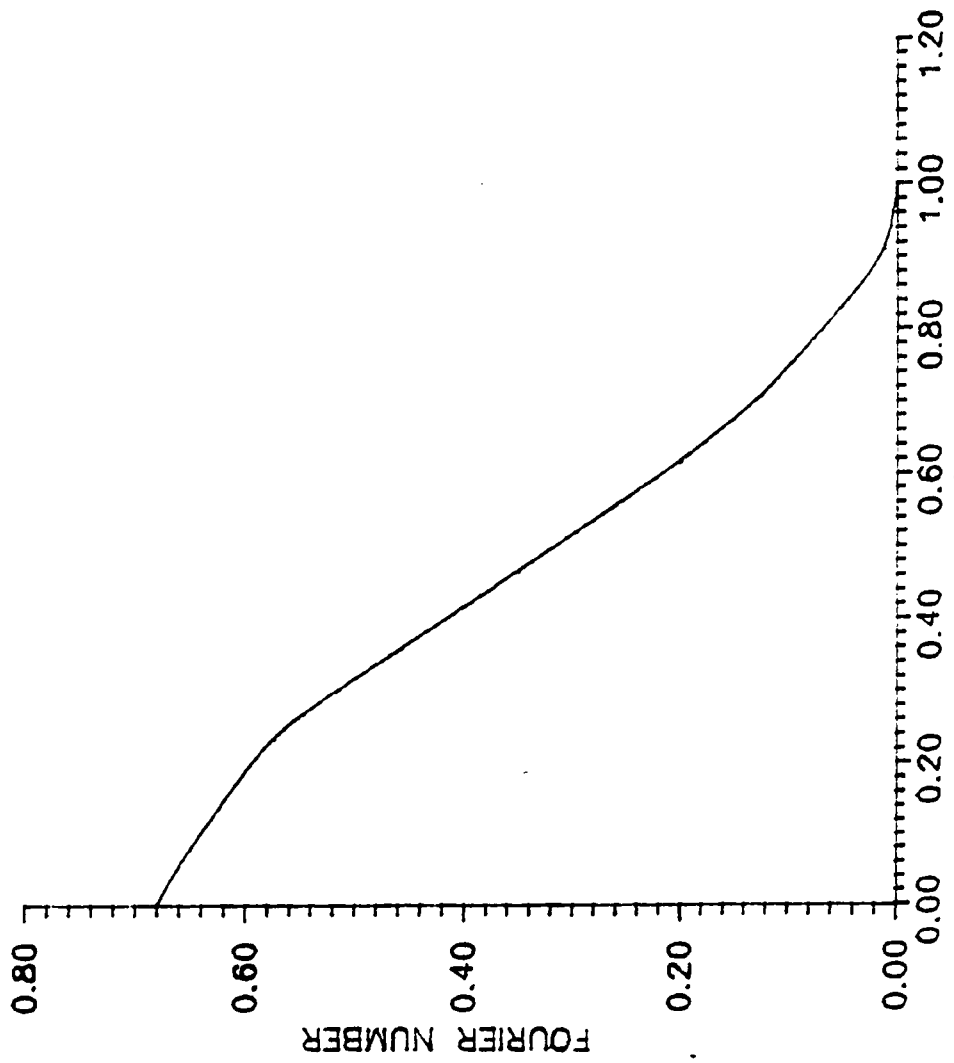


Figure 5.3 Interface Position for Approximate Tear-Drop Geometry Along the Line of Maximum Radius

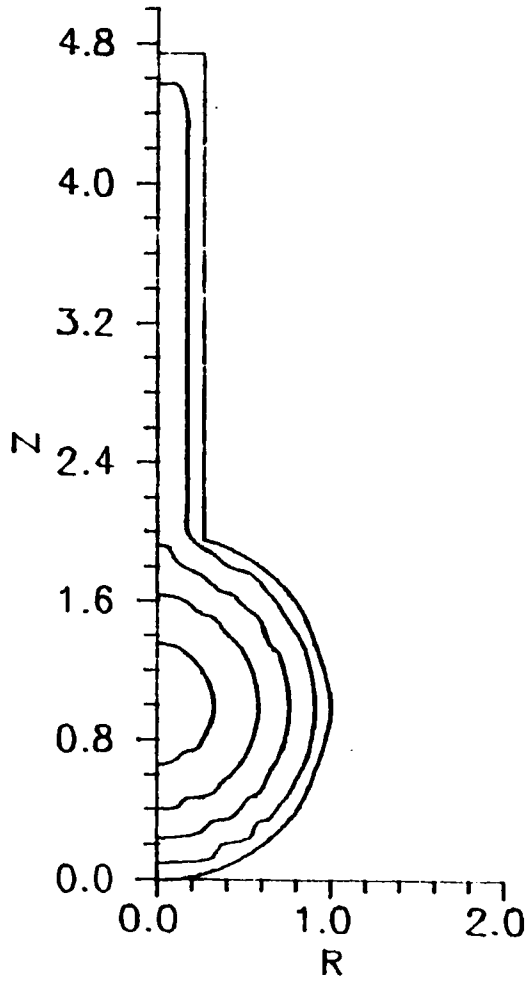


Figure 5.4 Interface Position for Approximate Tear-Drop Geometry

RESULTS OF TEMPERATURE DISTRIBUTION:

The values of temperature for any node and at any time during the solidification can be obtained from the known values of enthalpy and the appropriate temperature vs. enthalpy relationship.

The temperature distribution on the line corresponding to maximum Rmax(j) is shown in Figure (5-5).

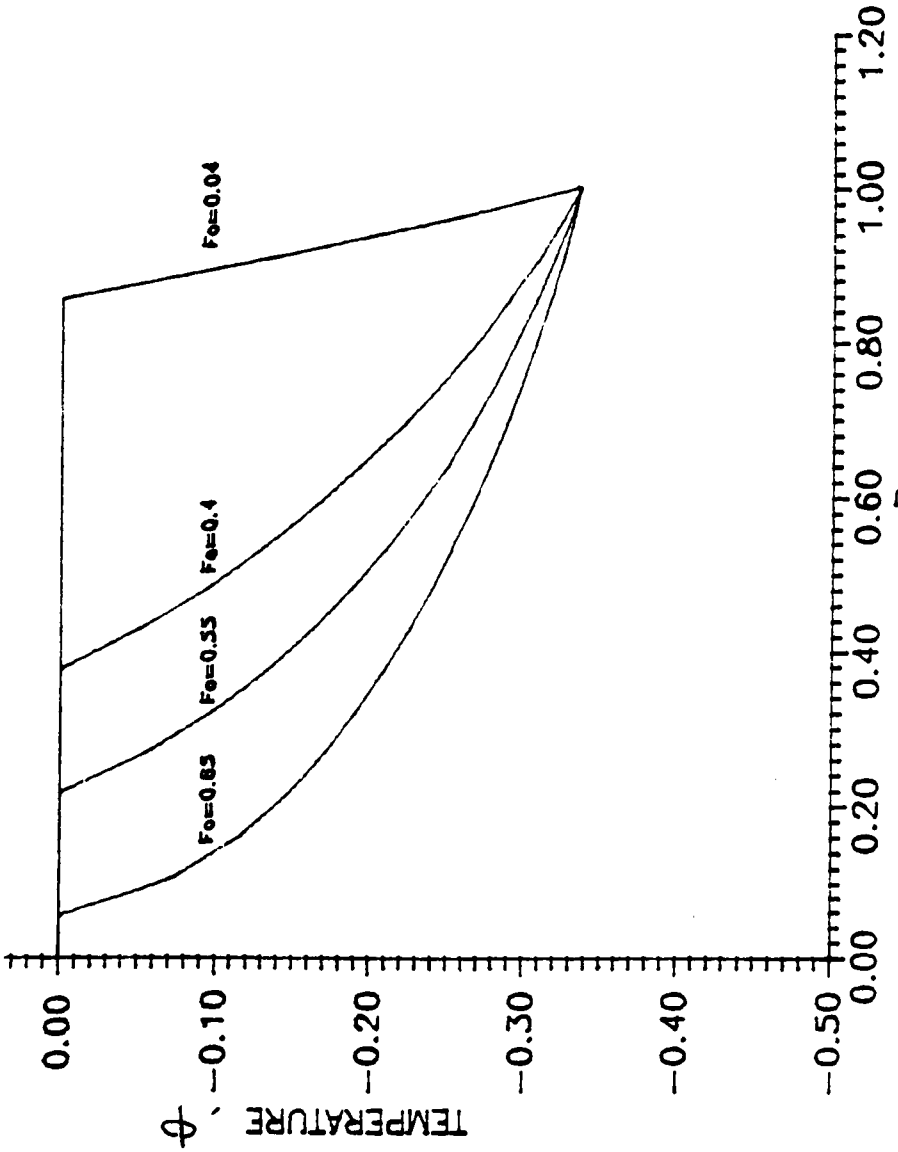


Figure 5.5 Temperature Distribution for Approximate Tear-Drop Geometry on Line of Maximum Radius

CHAPTER 6

EXPERIMENTAL STUDY

This chapter reports the results of some experimental studies of the phase-change material subjected to a medium with a convective heat transfer coefficient, h , and at temperature, T_∞. The experiments were conducted in two phases. In the first part, the geometry of the problem introduced in Chapter 5 was investigated (i.e. a cylinder protruding from a sphere).

In the second part, the problem of freezing of a cylinder with height, h, and radius, R, was considered. Note that the numerical solution for the second problem is shown in Chapter 4 for both the specified boundary temperature and the convective boundary condition cases.

In both experiments the primary interest was in

obtaining the location of the solid-liquid interface as a function of time. The results of these experiments were, then compared with those obtained by the enthalpy method.

EXPERIMENTAL SET-UP:

In the first experiment, the approximate tear-drop shaped container was made by attaching a copper tube (with 1.625 inches diameter and 8.1 inches height) to a three inch radius stainless steel sphere shell with thickness of 0.033 inches. The phase-change material used was Chevron 140 wax, which is the wax commonly used in candle production. The available thermal properties of Chevron 140 wax are given in Appendix C.

It should be pointed out that the thermal resistance of the container is assumed negligible. This is a reasonable assumption (even though stainless steel is not a good thermal conductor) because of the very small thickness of the sphere shell and, also, the fact that thermal conductivity of the wax is much smaller than that of the steel.

Six thirty six gage copper-constantan thermocouples

were glued on a very thin carrying wire and placed on the line corresponding to the maximum radius of the sphere. The carrying wire was pulled tightly from both sides, and the holes were sealed completely. The distances from the thermocouple to a point (point A in Figure 6.1 is known; therefore, the exact position of the thermocouples inside the sphere can be determined and the variation of temperature at these points can be noted as the solidification process continues.

All thermocouple data were acquired using a Hewlett-Packard 9826 microcomputer in conjunction with a 3497-A data acquisition/control unit. The computer program that was used for taking thermocouple readings is presented in Appendix D.

In the second experiment, an aluminum soft drink can (Coca Cola can)was used as the container. the rest of the experimental apparatus was identical to the set-up described above. Water at 80 F was used as the convective environment.

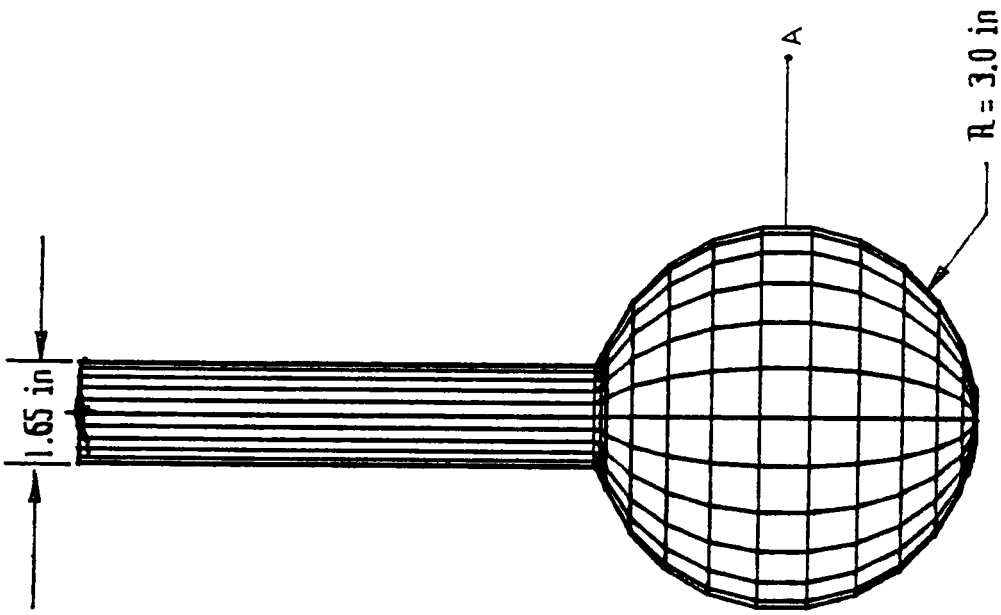


Figure 6.1 Approximate Tear-Drop Container

TEST PROCEDURE AND RESULTS:

A sufficient amount of wax was melted and poured into the mold. Then, the mold was placed in an oven at a temperature slightly higher than the melt temperature of the wax for a period of time so that the wax would be at a uniform temperature. Next, the mold was submerged in water at 80 F and the thermocouple readings were recorded at various time intervals. At any instant of time, the location of the interface can be obtained by knowing the position of the thermocouple whose temperature reading has just dropped below the melt temperature of the wax.

In Figure 6.2 the radial position of the solid-liquid interface is shown against the non- dimensional time for experiment number one (tear-drop geometry). Note that this curve exhibits the same behavior as the one obtained using the numerical method in Chapter 5. The freezing time is longer, as expected, because of the convective boundary condition (rather than specified boundary temperature).

From this graph it can be seen that the freezing rate

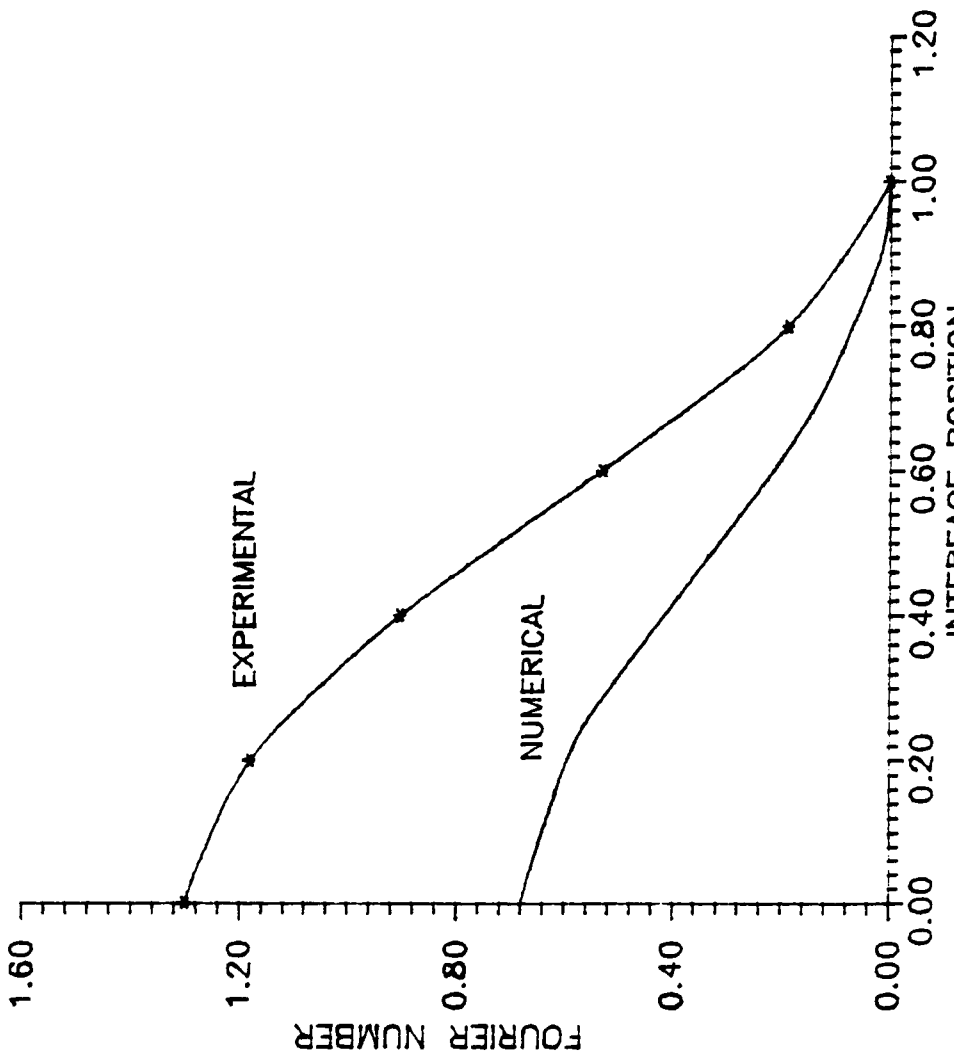


Figure 6.2 Interface Position for Approximate Tear-Drop Geometry

is higher at the initial and the final period of the solidification process, and it stays constant for the majority of the freezing.

Figure 6.3 contains the interface location results of the second experiment (circular cylinder). This graph also exhibits the general trend. By comparing this graph with those obtained using the enthalpy method with convective boundary conditions, one can estimate the Biot modulus of the experiment. Such a comparison leads to a Biot number of approximately 2., which corresponds to an effective heat transfer coefficient of about 13 Btu/ft2-hr-F.

UNCERTAINTY ANALYSIS:

Since the location of the interface is obtained indirectly, (by knowing the position of the thermocouple with temperature just below the melting point of the PCM, rather than using an equation relating the interface location to other variables) a precise method of estimating the uncertainty (such as the method of Kline and McClintok) can not be employed. However, a commonsense analysis of data is provided to determine experimental uncertainties.

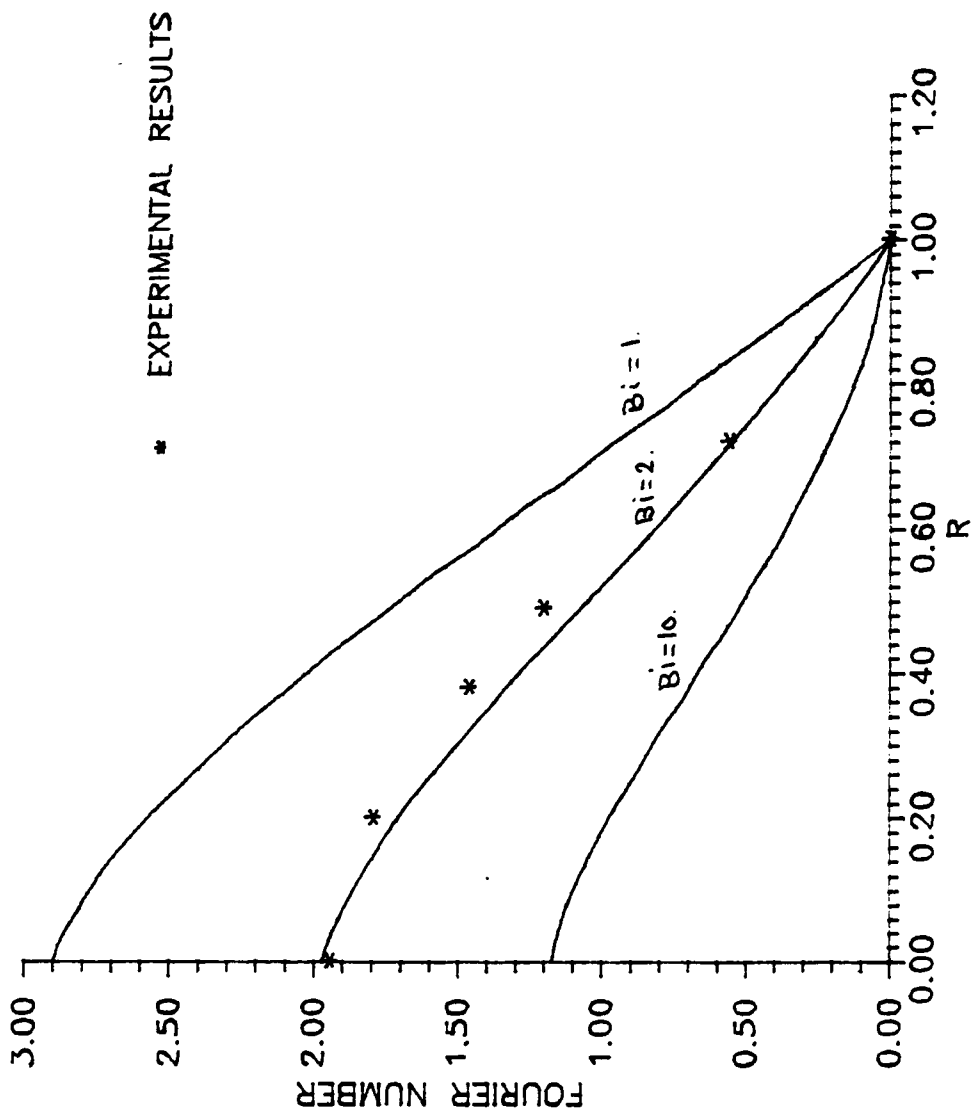


Figure 6.3 Interface Position for Circular Cylinder

The uncertainty in experimental data can be obtained by realizing that the time between successive temperature readings is the main contributing factor to the uncertainty. Thus, by estimating the speed of the solid-liquid interface and knowing the time interval between two readings one can estimate the uncertainty in the location of the interface.

This procedure is applied to the two experiments in Appendix A2 . For the first experiment the resulting uncertainty in nondimensional interface position is 0.124 at the final stages of the freezing process and it is .042 during most of the solidification period where the speed of the interface is almost constant. The same procedure is applied to the second experiment and it yields to uncertainties of 0.088 and 0.0159 for the final solidification period and the middle part of solidification respectively.

CHAPTER 7

CONCLUSION

The main objective of this study was to investigate the thermal aspects of a typical solidification process using a numerical technique. After reviewing several numerical schemes for solving phase-change problems, the so-called enthalpy method was found to be most attractive for phase-change problems because of its simplicity and versatility.

The enthalpy method, in conjunction with a fully implicit finite difference scheme, was then applied to two model axi-symmetric problems. In Chapter 4, the enthalpy method was applied to a sample solidification problem in a circular cylinder geometry and the results of temperature distribution and interface location were obtained for the cases of convective boundary condition

and specified temperature boundary condition. In both cases, the result of interface location shows that the speed of the front decreases after the initial portion of solidification and stays nearly constant for a major part of the solidification process. Also the effect of increasing the Biot modulus on the results obtained were studied. It was shown that the freezing time obtained using a large value for the Biot number is identical to that of the same problem when subjected to a fixed boundary temperature.

To check the validity of the numerical method, the results of the interface position for the case of fixed boundary temperature were checked against the known solution obtained by a different numerical approach. The two solutions agreed very closely with a maximum deviation of about six percent.

In Chapter 5, the numerical technique, developed in the earlier chapters, was employed to obtain solutions for an axi-symmetric problem with a more complex geometry. The geometry of an approximate tear-drop candle was considered and a complete set of results for the interface location and temperature distribution was obtained. The results of Chapter 4 and Chapter 5 were next compared with the

experimental result in Chapter 6.

LIMITATIONS AND RECOMMENDATIONS:

Some of the limitations of the numerical method developed in this study, along with some recommendations are presented below:

1) Uniform density throughout the entire region comprising the solid and liquid phases:
One of the major restrictions in developing the enthalpy method (or any otherr numerical numerical method for phase-change problems) is that the density of the solid phase be equal to the density of the liquid phase. This assumption is necessary because without it one has to take the void formation into account; this means keeping track of yet another moving boundary (the boundary between the void and the PCM). Furthermore, in some phase-change problems, the existence of more than a single cavity is possible, making the analysis of the phase-change process even more complicated. Shamsundar and Sparrow [34] considered the effect of density change on the phase-change process for solidification of a liquid in a long, horizontal rectangular container. However, their approach involves a number of assumptions and can not be extended

to other geometries. To this date, there have been no other successful attempts in obtaining numerical solutions for phase-change problems with density change for multi-dimensional problems. The use of experimental techniques is recommended for cases in which there is a particular interest in the formation of cavities as a result of the phase-change process.

2) Phase-change occurs at a discrete temperature (Tsat):
The enthalpy method is capable of finding solutions for cases involving materials that change phase diffusively over a range of temperatures, as well as those for which phase-change occurs at a discrete temperature (in fact, this is one of the advantages of the enthalpy method). However, in this study the algorithm was developed for cases involving phase-change at discrete temperature; therefore, some modifications in the computer programs must be made to examine cases involving phase-change over a range of temperatures.

3) Conduction is the only mode of heat transfer:
This may not be a good assumption for some cases (as will be seen in the next paragraph). One alternative is to solve the induced natural convection problem in the liquid phase at each time level. For this purpose, a heat

transfer and fluid flow algorithm, such as SIMPLER (Semi-Implicit Method for Pressure Linked Equations, Revised [35]) must be used in conjunction with the enthalpy algorithm. Note that as the phase-change process proceeds, the shape of the convection region varies.

4) P.C.M. is initially in the super-heated liquid state: Relaxation of this assumption would not necessitate any major changes to the computational procedure. The method shown in this study can easily be applied to cases in which the material is initially in a super-heated liquid state (i.e. the initial temperature of the material higher than the saturation temperature) as well as cases in which the P.C.M is initially at its saturation temperature (in program Super Heata the material is initially super-heated). However, in a real situation, there may be considerable natural convection before the solidification starts, and therefore, the assumption of negligible fluid motion would not be valid. Due to this fact, the numerical scheme may not be a good model for the case of super heated liquids.

There is, however, an alternative approach suggested by Nesselman [36] for cases in which initial liquid super-heat is present. Using this method, bounds on the solution can be obtained by performing two sets of calculations

each with the liquid initially at its saturation temperature. In the first set of calculations, the material's true latent heat, λ, is used . In the second set, the latent heat, $\lambda 1$,is taken to be:

$$\lambda_1 = \lambda + (e_{sup} - e)$$

Where e_{sup} is the enthalpy of the superheated liquid. In order to obtain more accurate solutions, the natural convection problem inside the liquid phase must be solved.

BIBLIOGRAPHY

BIBLIOGRAPHY:

1. Lame, G., Clapeyron, B., P., Ann. Chem. Phys., vol. 47, pp. 250-256, 1831.

2. Stefan, J., "Uber die Theorie der Eisbildung, Insbesondere Uber die Eisbildung in Polarmare", Annalen der Physik und Chemie, vol. 42, pp. 269-286, 1891.

3. Wilson, D., G., Soloman, A., A.,Buggs, P.,T.,Moving Boundary Problems, Academic Press, pp. 165-185, 1978.

4. Solomon, Alexiades, Wilson, "The Quasi-Stationary Approximation for the Stefan Problem with a Convective Bounary Condition", Int. Journal Math and Math. Sci., vol. 7, No. 3, pp. 549-563, 1984.

5. Jiji, L., M, Weinbaum, S., "A Non-Linear Perturbation Theory for Non-Similar Melting or Freezing Problems", Conduction Cu-3.4, 5th International Heat Transfer Conference, Tokyo, 1974.

6. Spaid, F. ,W., et al, "Shape Evolution of a Subliming

Surface Subjected to Unsteady Spatially Nonuniform Heat Flux", Int. Journal of Heat And Mass Transfer, Vol. 14, pp. 673-687, 1971.

7. Alexiades, V., "Rapid Freezing of Dilute Alloys", IMA Journal of Applied Math., vol. 38, pp. 67-69, 1983.

8. Alexiades, V., Class Notes for Math 5995, Seminar in Applied Mathematics, University of Tennessee, Knoxville, 1986-1987.

9. Goodman, T., R., "The Heat Balance Integral, Further Consideration and Refinements", Journal of Heat Transfer, 83c, pp. 83-86, 1961.

10. Poots, G. , "An Approximate Treatment of a Heat Conduction Problem Involving a Two-Dimensional Solidification Front", International Journal of Heat and Mass Transfer, Vol. 5, pp. 339-348, 1962.

11. Ozisik, M., N., Heat Conduction ,Wiley & Sons Inc. , 1980.

12. Biot, M. ,A. ,Journal Of Aeronaut. Sci. ,Vol. 24, pp. 857-873, 1957.

13. Lightfoot, N., M., H., Proc. London Math. Soc., Vol.31, pp. 97-116, 1929.

14. Jiji, L., M., Rathjen, K. A., "Heat Conduction With Melting Or Freezing In A Corner", Journal Of Heat Transfer, Vol. 93, pp. 101-109, 1971.

15. Budhia, H., Kreith, F., "Heat Transfer With Melting Or Freezing In A Corner", Int. Journal Of Heat And Mass Transfer, VOl. 16, pp. 195-211, 1973.

16. Boley, B., A., Journal of Math. Physics, Vol. 40, pp. 300-313, 1961.

17. Sikarskie, D.,L., Boley, B., A., Int. Journal of Solids and Structures, Vol. 1, pp. 207-234, 1965.

18. Boley, B., A., Yogoda, H., P., Quarterly Applied Mathematics, Vol. 27, pp. 223-246, 1969.

19. Siegel, R., Goldstein, M., E., Sovino, J., M., "Conformal MApping Procedure for Transient and Steady-State Two-Dimensional Solidification", Proceeding of The

Fourth International Heat Transfer Conference, Vol. 1, Paper No. Cu. 2.11, 1970.

20. Tao, L., C., "Generalized Numerical Solutions of Freezing a Saturated Liquids in Cylinders and Spheres", A.I.CH.E. Journal, Vol. 13, No. 1, pp. 165-169, 1967.

21. Springer, G., S., Olson, D., R., "Method of Solution of Axisymmetric Solidification and Melting Problems", ASME Paper No. 62-WA-246.

22. Lazaridis, A., "A Numerical Solution of The Multi-Dimensional Solidification (or Melting) Problem", Int Journal Of Heat And Mass Transfer, Vol. 13, pp. 1459-1477, 1970.

23. Bilenas, J., A., Jiji, L., M., "Numerical Solution of a Nonlinear Free Bounadry Problem of Axi-symmetric Flow in Tubes with Surface Solidification", Proceedings of the Fourth International Heat Transfer Conference, Vol. 1, Paper No. Cu. 2.1, 1970.

24. Shamsundar, N., PhD Thesis, Department of Mechanical Engineering, University of Minnesota, 1975.

25. Duda, J., L., et al ,"Analysis of Two-Dimensional Diffusion-Controlled Moving Boundary Problems", Int. Journal of Heat and Mass Transfer, Vol. 18, pp. 901-910, 1975.

26. Crank, J., Gupta, R., S., "Isotherm Migration Methods in Two Dimensions", Int. Journal of Heat And Mass Transfer, Vol. 18, pp. 1101-1107, 1975.

27. Hashemi, H., T., Sliepcenich, C., M., "A Numerical Method for Solving Two-Dimensional Problems o Heat Conduction with Change of Phase", Chemical Engineering Progress Symposium Series No. 79, Vol. 63, pp. 34-41, 1967.

28. Meyer, G., H.,"Multi_dimensional Stefan Problem", SIAM Journal on Numerical Analysis, Vol. 10, pp. 522-538, 1973.

29. Shamsundar, N.,Sparrow, E., M., "Analysis of Multidimensional Conduction Phase Change Via the Enthalpy Model",Journal of Heat Transfer, pp. 333-340, 1975.

30. Comini, G., et al "Finite Element Solution of Non-Linear Heat Conduction Problems With Special Reference to Phase Change", Int. Journal for Numerical Methods in

Engineering, Vol. 8, pp. 613-624, 1974.

31. Kamenomostskaja, S., L., "On the Stefan Problem", Quarterly Applied Mathematics, vol. 53, pp.489-514, 1961.

32. Athey, D., R., "A Finite Dufference Scheme for Moving Boundary Problems", J. Inst. Math. Applic., vol. 13, pp. 352-366, 1974.

33. Jerome, J., W., "Nonlinear Equations of Evolution for Generalized Stefan Problem", J Diff. Eqn., vol. 26, 1977.

34. Shamsundar, M., Sparrow, E., M., "Effect of Density Change on Multidimensional Conduction Phase Change", Journal of Heat Transfer, pp. 550-557, 1976.

35. Patankar, S. V., Numerical Heeat Transfer and Fluid FLlow, Hemisphere Publishing, Washington, 1980.

36.Nesselmann, K., "Systematik der Gleichungen fur Gefrieren und Schmelzen von Eisschichten nebst Anwendung auf Trommelgefrierapparate und Susswasserkuhler", vol. 1, pp. 169-172, 1949.

APPENDIXES

APPENDIX A

DERIVATION OF FINITE DIFFERENCE EQUATIONS

WHEN CURVED BOUNDARY INTERSECTS

THE STRINGS

Recall that for the interior nodes the finite difference form of the enthalpy equation is given by:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^{m}_{i,j}+\left[2.+2.\left(\frac{RN}{ZN}\right)^2\right].\ \phi^{m}_{i,j}=\left(\frac{RN^2}{\Delta Fo}\right)\theta^{m-1}_{i,j}+$$
$$\left(1-\frac{1}{2i}\right)\phi^{m}_{i,j-1}+\left(1+\frac{1}{2i}\right)\phi^{m}_{i+1,j}+\left(\frac{RN}{ZN}\right)^2\left(\phi^{m}_{i,j-1}+\phi^{m}_{0ij+1}\right) \qquad (5.6)$$

However, when a region has curved boundary, it is possible that the boundary intersects the string at some points other than the regular grid-points.

Therefore, it is necessary to derive a finite difference equation to represent the first and second derivatives for nodes near the curved boundary. In the following derivation it is assumed that the boundary is subjected to a specified temperature, T_∞.

Figure (A1-1) shows a case in which the right R string is crossed by the curved boundary. As mentioned in chapter 5 $\zeta\Delta r$ represents the distance between the grid point (i,g) and the boundary. Using Taylor series

expansion we get the following equations:

$$T_B = T_{i,j} + (\xi\Delta r)\cdot \left.\frac{\partial T}{\partial r}\right|_{i,j} + \left[\frac{(\xi\Delta r)^2}{2}\right]\cdot \left.\frac{\partial^2 T}{\partial r^2}\right|_{i,j} \qquad (A1.1)$$

$$T_{i-1,j} = T_{i,j} - (\xi\Delta r)\cdot \left.\frac{\partial T}{\partial r}\right|_{i,j} + \left[\frac{(\xi\Delta r)^2}{2}\right]\cdot \left.\frac{\partial^2 T}{\partial r^2}\right|_{i,j} \qquad (A1.2)$$

Now, if the second equation is multiplied by ξ and the resulting equation added to the first equation it yields:

$$T_B + \xi T_{i-1,j} = (\xi+1)\cdot T_{i,j} + \left[\frac{(\xi\Delta r)^2}{2}\right](\xi+1)\cdot \left.\frac{\partial^2 T}{\partial r^2}\right|_{i,j}$$

$$\left.\frac{\partial^2 T}{\partial r^2}\right|_{i,j} = \frac{2}{(\Delta r)^2}\left[\frac{T_B}{\xi(\xi+1)} + \frac{T_{i-1,j}}{(\xi+1)} - \frac{T_{i,j}}{\xi}\right] \qquad (A1.3)$$

Similarly, if $\left.\frac{\partial^2 T}{\partial r^2}\right|_{i,j}$ is eliminated between equations (A1-1)and (A1-2), an expression for $\frac{\partial T}{\partial r}$ can be obtained.

$$\left.\frac{\partial T}{\partial r}\right|_{i,j} = \frac{1}{(\Delta r)^2}\left[\frac{1}{\xi(\xi+1)}T_B + \frac{\xi}{(\xi+1)}T_{i-1,j} - \frac{(1-\xi)}{\xi}T_{i,j}\right] \qquad (A1.4)$$

The derivation of the finite difference representation of $\frac{\partial^2 T}{\partial z^2}\Big|_{i,j}$ and $\frac{\partial T}{\partial z}\Big|_{i,j}$ when one Δz string is crossed by the curved boundary is similar to the above derivation.

Now, suppose two Δz (or Δr) strings are intersected by the curved boundary.

In this case the following equations can be written using Taylor series expansions:

$$T_B = T_{i,j} + (\eta \Delta z) \cdot \frac{\partial T}{\partial z}\Big|_{i,j} + \left[\frac{(\eta \Delta z)^2}{2}\right] \cdot \frac{\partial^2 T}{\partial z^2}\Big|_{i,j} \qquad (A\,1.5)$$

$$T_B = T_{i,j} - (\eta' \Delta z) \cdot \frac{\partial T}{\partial z}\Big|_{i,j} + \left[\frac{(\eta' \Delta z)^2}{2}\right] \cdot \frac{\partial^2 T}{\partial z^2}\Big|_{i,j} \qquad (A\,1.6)$$

Multiply equation (A1-5) by η' and equation (A1-6) by η and add the two resulting equations

$$(\eta + \eta') \cdot T_B = (\eta + \eta') \cdot T_{i,j} + \left[\eta\eta' \frac{(\Delta z)^2}{2}(\eta + \eta')\right] \cdot \frac{\partial^2 T}{\partial z^2}\Big|_{i,j} \qquad (A\,1.7)$$

Hence, $\frac{\partial^2 T}{\partial z^2}\Big|_{i,j}$ is:

$$\left.\frac{\partial^2 T}{\partial z^2}\right|_{i,j} = \left[\frac{2}{\eta\eta'(\Delta z)}\right](T_B - T_{i,j}) \qquad (A1.8)$$

At this point the finite difference form of the enthalpy equation for different kinds of nodes the curved boundary can be written.

CASE ONE:

Consider the situation in which one Δr string is crossed by the curved boundary. For this case, using equations (A1-3) and (A1-4) the finite difference form of the right side of the enthalpy equation can be evaluated as:

$$\nabla^2\phi = \left[\frac{2}{(1+\xi)\Delta R^2}\right]\left(1 - \frac{\xi}{2i}\right)\phi_{i-1,j} - \left[\frac{2}{(\xi\Delta R)^2}\right]\left(1 + \frac{1-\xi}{2i}\right)\phi_{i,j}$$
$$+ \left[\frac{2}{\xi(\xi+1)(\Delta R)^2}\right]\left(1 + \frac{1}{2i}\right)\phi_B + \frac{1}{\Delta Z^2}(\phi_{i,j-1} - 2\phi_{i,j} + \phi_{i,j+1})$$

Therefore, the finite difference form of the enthalpy equation for this node is:

$$\frac{(\theta^{m}_{i,j} - \theta^{m-1}_{i,j})}{\Delta Fo} = \left[\frac{2}{(1+\xi)\Delta R^2}\right]\left(1 - \frac{\xi}{2i}\right)\phi^{m}_{i-1,j} - \left(\frac{2}{\xi\Delta R^2}\right)\left(1 + \frac{1-\xi}{2i}\right)\phi^{m}_{i,j}$$
$$+\left[\frac{2}{\xi(1+\xi)\Delta R^2}\right]\left(1 + \frac{1}{2i}\right)\phi^{m}_{B} + \frac{1}{\Delta Z^2}\left(\phi^{m}_{i,j-1} - 2\phi^{m}_{i,j} + \phi^{m}_{i,j+1}\right)$$

Multiply through by ΔR^2 and rearrange:

$$\left(\frac{RN^2}{\Delta Fo}\right)\theta^{m}_{i,j} + \left[\frac{2}{\xi}\left(1 + \frac{1-\xi}{2i}\right) + 2\left(\frac{RN}{ZN}^2\right)\right]\cdot\phi^{m}_{i,j} = \left(\frac{RN^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} +$$
$$\left[\frac{2}{\xi(1+\xi)}\right]\left(1 + \frac{1}{2i}\right)\cdot\phi^{m}_{B} + \left(\frac{2}{1+\xi}\right)\left(1 - \frac{\xi}{2i}\right)\phi^{m}_{i-1,j} +$$
$$\left(\frac{RN}{ZN}\right)^2\left(\phi^{m}_{i,j-1} + \phi^{m}_{i,j+1}\right) \qquad (A1.9)$$

CASE TWO:

Consider the case in which one Δz string (the upper) is crossed by the curved boundary. Use the appropriate expression for $\frac{\partial^2 \phi}{\partial z^2}$ in equation (A1-8) to get:

$$\nabla^2\phi\Big|_{i,j} = \frac{1}{\Delta R^2}\left[\left(1-\frac{1}{2i}\right)\phi_{i-1,j} - 2\phi i,\ j + \left(1+\frac{1}{2i}\right)\phi_{i+1,j}\right] + \frac{2}{\Delta Z^2}\left(\frac{\phi_B}{\eta(\eta+1)} + \frac{\phi_{i,j+1}}{(\eta+1)}\right)$$

Now, the finite difference form of the enthalpy equation for this node as obtained as:

$$\frac{\left(\theta^{m}_{i,j} - \theta^{m-1}_{i,j}\right)}{\Delta Fo} = \frac{1}{\Delta R^2}\left[\left(1-\frac{1}{2i}\right)\phi^{m}_{i-1,j} - 2\phi^{m}_{i,j} + \left(1+\frac{1}{2i}\right)\phi^{m}_{i+1,j}\right] + \frac{2}{\Delta Z^2}\left[\frac{\phi_B}{\eta(\eta+1)} + \frac{\phi_{i,j+1}}{(\eta+1)}\right]$$

This equation can be rearranged to get the enthalpy equation in its familiar form:

$$\left(\frac{\Delta R^2}{\Delta Fo}\right)\theta^{m}_{i,j} + 2.\left[1+\frac{1}{\eta_1}\left(\frac{\Delta R}{\Delta Z}\right)^2\right]\cdot\phi^{m}_{i,j} = \left(\frac{\Delta R^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} +$$

$$\left(1-\frac{1}{2i}\right)\phi^{m}_{i-1,j}+\left(1+\frac{1}{2i}\right)\phi^{m}_{i+1,j}+2.\left(\frac{\Delta R}{\Delta Z}\right)^{2}\left[\frac{\phi^{m}_{B}}{\eta(\eta+1)}+\frac{\phi^{m}_{i,j+1}}{(\eta+1)}\right] \qquad (A1.10)$$

Note that for the case of lower Δz intersected by the curved boundary the derivation of the finite difference equation is similar to the above derivation and the resulting equation will be identical to equation (A1-10) with the exception of $\frac{\phi^{m}_{i,j-1}}{\eta}$ changed to $\frac{\phi^{m}_{i,j+1}}{\eta}$.

CASE THREE:

Here, we consider the case in which one Δr (the right) and one Δz (the upper) is crossed by the curved boundary. For this situation using the appropriate expressions for $\frac{\partial^2 \phi}{\partial R^2}$ and $\frac{\partial^2 \phi}{\partial Z^2}$, $\nabla^2 \phi|_{i,j}$ can be obtained as:

$$\nabla^2\phi\Big|_{i,j} = \left[\frac{2}{(1+\xi)\Delta R^2}\right]\cdot\left(1-\frac{\xi}{2i}\right)\phi_{i-1,j} - \left(\frac{2}{\xi\Delta R^2}\right)\left(1+\frac{1-\xi}{2i}\right)\phi_{i,j} + \left[\frac{2}{\xi(1+\xi)\Delta R^2}\right]\left(1+\frac{1}{2i}\right)\phi_{i+1,j} + \left(\frac{2}{\Delta Z^2}\right)\left[\frac{\phi_B}{\eta(\eta+1)} + \frac{\phi_{i,j+1}}{(\eta+1)}\right]$$

The finite difference equation for this case can be written as:

$$\left(\frac{\Delta R^2}{\Delta Fo}\right)\theta^m_{i,j} + 2.\left[\frac{1}{\xi}\left(1+\frac{1-\xi}{2i}\right) + \frac{1}{\eta}\left(\frac{\Delta R}{\Delta Z}\right)^2\right]\cdot\phi^m_{i,j} = \left(\frac{\Delta R^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} + \left[\frac{2}{\xi(1+\xi)}\right]\left(1+\frac{1}{2i}\right)\phi^m_B + \left(\frac{2}{1+\xi}\right)\left(1-\frac{\xi}{2i}\right)\phi^m_{i-1,j} + 2.\left(\frac{\Delta R}{\Delta Z}\right)^2\cdot\left[\frac{\phi^m_B}{\eta(\eta+1)} + \frac{\phi^m_{i,j-1}}{(\eta+1)}\right] \qquad (A\,1.11)$$

CASE FOUR:

Finally, the case in which three string (one r and two Δz) are intersected by the curved boundary is considered. For this case, the Laplacian $\nabla^2\phi|_{i,j}$ is given by.

$$\nabla^2\phi\Big|_{i,j} = \left[\frac{2}{(1+\xi)\Delta R^2}\right]\cdot\left(1-\frac{\xi}{2i}\right)\phi_{i-1,j} - \left(\frac{2}{\xi\Delta R^2}\right)\left(1+\frac{1-\xi}{2i}\right)\phi_{i,j} + \left[\frac{2}{\xi(1+\xi)\Delta R^2}\right]\left(1+\frac{1}{2i}\right)\phi_B + \left(\frac{2}{\eta\eta'\Delta Z^2}\right)[\phi_B + \phi_{i,j}]$$

And the finite difference form of the enthalpy equation for this case is:

$$\left(\frac{\Delta R^2}{\Delta Fo}\right)\theta^m_{i,j} + 2.\left[\frac{1}{\xi}\left(1+\frac{1-\xi}{2i}\right)+\frac{1}{\eta\eta'}\left(\frac{\Delta R}{\Delta Z}\right)^2\right]\cdot\phi^m_{i,j} = \left(\frac{\Delta R^2}{\Delta Fo}\right)\theta^{m-1}_{i,j} + \left[\frac{2}{\xi(1+\xi)}\right]\left(1+\frac{1}{2i}\right)\phi^m_B + \left(\frac{2}{1+\xi}\right)\left(1-\frac{\xi}{2i}\right)\phi^m_{i-1,j} + 2.\left(\frac{\Delta R}{\Delta Z}\right)^2\left(\frac{\phi^m_B}{\eta\eta'}\right) \qquad (A\,1.12)$$

APPENDIX A2

UNCERTAINTY ANALYSIS FOR EXPERIMENTAL RESULTS

As mentioned in Chapter 6, the main contributing factor to the experimental uncertainty was the time interval between successive temperature reading. For both experiments the time interval was kept constant at one minute. This translates to Fourier number interval of 0.0375. For the first experiment the speed of the front for the final period and the middle period is calculated using data from Figure 6.2 :

$$\frac{\partial R}{\partial F_o} \approx \frac{\Delta R}{\Delta F_o}\Big|_{\text{Final Period}} = \frac{0.2}{0.6} = 3.33$$

$$\frac{\partial R}{\partial F_o} \approx \frac{\Delta R}{\Delta F_o}\Big|_{\text{Middle Period}} = \frac{0.2}{0.18} = 1.11$$

Therefore, the experimental uncertainty can be estimated as:

$$\omega_R = \frac{\partial R}{\partial F_o}\omega_{F_o} = (3.3)(.0375) = 0.125 \quad \text{Final Period}$$

$$\omega_R = \frac{\partial R}{\partial F_o}\omega_{F_o} = (1.11)(.0375) = 0.0416 \quad \text{Middle Period}$$

Similarly, for the second experiment, using data from Figure 6.3 we have:

$$\frac{\partial R}{\partial F_o} \approx \frac{\Delta R}{\Delta F_o}\Big|_{\text{Final Period}} = \frac{0.2}{0.17} = 1.176$$

$$\frac{\partial R}{\partial F_o} \approx \frac{\Delta R}{\Delta F_o}\Big|_{\text{Middle Period}} = \frac{0.115}{0.27} = 0.426$$

Which results in an experimental uncertainty of:

$$\omega_R = (1.176)(.0375) = 0.044 \quad \text{Final Period}$$

$$\omega_R = (0.426)(.0375) = 0.016 \quad \text{Middle Period}$$

APPENDIX B

LISTING OF COMPUTER PROGRAMS

```
C=====================================================================
C
C       TITLE:  CANDLV3.FOR (CANDLE VERSION 3)
C
C        TASK:  THIS PROGRAM EMPLOYS THE ENTHALPY METHOD FOR
C               SOLVING HEAT TRANSFER PROBLEMS INVOLVING PHASE
C               CHANGE. THE GAUSS-SEIDEL ITERATIVE SCHEME IS USED
C               TO SOLVE THE SYSTEM OF SIMOULTANEOUS EQUATIONS.
C               VARIOUS RESULTS (SUCH AS ENTHALPY DISTRIBUTION,
C               TEMPERATURE DISTRIBUTION, INTERFACE LOCATION...)
C               CAN BE OBTAINED (AS OUTPUTS OF THE PROGRAM) BY
C               SETTING THE APPROPRIATE SWITCHES EQUAL TO 1.
C
C    GEOMETRY:  APPROXIMATE "TEAR-DROP" SHAPE.
C
C       INPUT:  INPUT DATA FILE IS GENERATED USING SITCAN3
C               SUBROUTINE IN "WRITE" MODE. THE INPUT CAN ALSO
C               BE MODIFIED DIRECTLY IN THE INPUT DATA FILE.
C
C
C---------------------------------------------------------------------
C      UPDATED: FEB. 19, 1988
C=====================================================================
C
C                                NOMENCLATURE
C
C    I/O...INPUT/OUTPUT
C      I: INPUT
C      O: OUTPUT
C      I/O: INPUT/OUTPUT
C    VAR...TYPE OF VARIABLE: INTEGER/REAL/CHARACTER
C      I: INTEGER
C      R: REAL
C      C: CHARACTER
C    DIM...DIMENSION:0,1,2,3,...
C
C---------------------------------------------------------------------
C   VARNAME I/O VAR DIM  DESCRIPTION
C   ------- --- --- ---  ------------------------------------------
C   CS      I    R   0    SPECIFIC HEAT FOR SOLID PHASE OF PCM
C   CF1     -    R   0    COEFFICIENT 1 IN GENERAL ENTHALPY EQUATION
C   CF2     -    R   0    COEFFICIENT 2 IN GENERAL ENTHALPY EQUATION
C   CF3     -    R   0    COEFFICIENT 3 IN GENERAL ENTHALPY EQUATION
C   DR      I    R   1    DIMENSIONAL DELTA R
C   DR1     O    R   2    RADIAL POSITION USED IN INTERFACE
C                         LOCATION SECTION
C   DZ      I    R   1    DIMENTIONAL DELTA Z
C   DZ1     O    R   2    VERTICAL POSITION USED IN INTERFACE
C                         LOCATION SECTION
C   DFO     -    R   0    NON-DIMENSIONAL TIME-STEP
```

```
C  EPS      I     R   0    MAXIMUM DEVIATION OF ENTHALPY AT ANY NODE
C                           FROM PREVIOUS ITERATION
C  ETA1     -     R   0    FRACTION OF DZ DISTANCE FROM FROM
C                           LOWER CURVED BOUNDARY
C  ETA2     -     R   0    FRACTION OF DZ DISTANCE FROM FROM
C                           UPPER CURVED BOUNDARY
C  FI       I/O   R   3    NON-DIMENSIONAL TEMPERATURE
C  FIB      I     R   0    NON-DIMENSIONAL BOUNDARY TEMPERATURE
C  FO       -     R   0    NON-DIMENSIONAL TIME (FOURIER NO.)
C  FLAG     -     I   0    INDICATOR FOR CONVERGENCE
C  I        -     I   0    INDEX FOR R DIRECTION
C  IOF      P     I   0    INPUT/OUTPUT FILE NUMBER
C  IMAX     -     I   0    MAXIMUM VALUE OF INDEX I
C  J        -     I   0    INDEX FOR Z DIRECTION
C  JMAX     I     I   0    MAXIMUM VALUE OF INDEX J
C  JMAX1    -     I   0    NUMBER OF POINTS IN Z DIRECTION IN
C                           SPHERICAL SECTION
C  JRMAX    -     I   0    J CORRESPONDING TO MAX. RADIUS OF SPHERE
C  K        I     I   0    NUMBER OF TIME-STEPS
C  LAND     I     R   0    LATENT HEAT OF FUSION  OF PCM
C  LEN      -     R   0    LENGTH OF THE SPHERICAL PART
C  M        -     I   0    INDEX FOR TIME
C  MAXNUM   I     I   0    MAXIMUM NUMBER OF ITERATION
C  MF       -     I   0    VALUE OF TIME INDEX AT INSTANT
C                           CENTER FREEZES
C  NUMB     -     I   0    ITERATION COUNTER
C  NVI      -     I   1    NUMBER OF NODES IN R DIRECTION
C                           FOR A GIVEN Z
C  R        I     R   0    RADIUS OF SPHERICAL   SECTION
C  RAT      -     R   0    RATIO OF DR/DZ
C  RC       O     R   1    RADIAL POSITION USED IN INTERFACE
C                           LOCATION SECTION ON RMAX SECTION
C  RCY      I     R   0    RADIUS OF CYLINDRICAL SECTION
C  RN       -     R   0    NON-DIMENSIONAL DELTA R
C  RMAX     -     R   1    MAXIMUM RADIUS FOR GIVEN Z
C  RP       -   R→  1→     R COORDINATE OF GRID-POINT
C  RNP      -     R   1    R COORDINATE OF GRID-POINT (NON-DIMENSIONAL)
C  SWO11    I     I   0    SWITCH FOR OUTPUT TO TAPE11 (ON:1,OFF:0)
C  SWO12    I     I   0    SWITCH FOR OUTPUT TO TAPE12 (ON:1,OFF:0)
C  SWO13    I     I   0    SWITCH FOR OUTPUT TO TAPE13 (ON:1,OFF:0)
C  SWO14    I     I   0    SWITCH FOR OUTPUT TO TAPE14 (ON:1,OFF:0)
C  SWRW     P     I   0    SWITCH FOR READ OR WRITE MODE OF
C                           SUBROUTINE SITCAN3 (READ:1, WRITE:0)
C  TIMEF    -     R   0    VALUE OF FOURIER NUMBER AT THE INSTANT
C                           CENTER FREEZES
C  TEMP     -     R   2    TEMPERATURE ARRAY USED IN TEMPERATURE
C                           DISTRIBUTION SCTION
C  TH       I/O   R   3    NON-DIMENSIONAL ENTHALPY
C  THMOJ    -     R   1    DUMMY VARIABLE FOR ENTHALPY FOR I .EQ. 0
C  THMIJ    -     R   1    DUMMY VARIABLE FOR ENTHALPY FOR I .NE. 0
```

```
C  TSI      -    R   0    FRACTION OF DR DISTANCE FROM FROM
C                         RIGHT CURVED BOUNDARY
C  TSAT     I    R   0    SATURATION TEMPERATURE OF PCM
C  ZN       -    R   1    NON-DIMENSIONAL DELTA Z
C  ZP       -    R   1    Z COORDINATE OF GRID-POINT
C  ZNP      -    R   1    Z COORDINATE OF GRID-POINT (NON-DIMENSIONAL)
C
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
C
C============================
C:: DECLARATION OF VARIABLES
C----------------------------
C:: COMMON VARIABLES:
C--------------------
      COMMON /PORT01/ IMAX,JMAX,K,MAXNUM
      INTEGER         IMAX,JMAX,K,MAXNUM
C----------------------------
      COMMON /PORT03/ SWO11,SWO12,SWO13,SWO14,SWO15,SWO16
      INTEGER         SWO11,SWO12,SWO13,SWO14,SWO15,SWO16
C----------------------------
      COMMON /PORT04/ R,RCY
      REAL            R,RCY
C----------------------------
      COMMON /PORT05/ DR,      DZ
      REAL            DR(0:60),DZ(0:60)
C----------------------------
      COMMON /PORT06/ TSAT,LAND,CS
      REAL            TSAT,LAND,CS
C----------------------------
      COMMON /PORT07/ FIB,EPS
      REAL            FIB,EPS
C----------------------------
      COMMON /PORT08/ DFO
      REAL            DFO
C----------------------------
C:: INTERNAL VARIABLES:
C----------------------------
        INTEGER I,J,JMAX1,JRMAX,M,NUMB,FLAG,SWRW,IOF
        INTEGER NVI(0:60),KKM(90)
C----------------------------
        REAL    FO,LEN
        REAL    RP(0:60),ZP(0:60),RN(0:60),ZN(0:60)
        REAL    RC(0:90),DR2(0:30),RMAX(0:60),RNP(0:60),ZNP(0:60)
        REAL    DR1(0:90,100),DZ1(0:90,100),TEMP(0:90,0:60)
        REAL    TH(0:90,0:60,0:60),FI(0:90,0:60,0:60)
C----------------------------
C:: START ::
C----------
C
C.........ENTERING THE PARAMETERS OF THE PROBLEM
```

```
C
      SWRW=1
      IOF =19
      CALL SITCAN3(IOF,SWRW)
C
C.....COMPUTING VALUES FOR ZP, RP, ZNP, AND RNP
C
      ZP(0) =0.
      ZN(0) =DZ(0)/R
      ZNP(0)=0.
      DO 10001 J=1,JMAX
        ZP(J) =ZP(J-1)+DZ(J)
        ZN(J) =DZ(J)/R
        ZNP(J)=ZP(J)/R
10001    CONTINUE
      RP(0) =0.
      RN(0) =DR(0)/R
      RNP(0)=0.
      DO 10002 I=1,IMAX
        RP(I) =RP(I-1)+DR(I)
        RN(I) =DR(I)/R
        RNP(I)=RP(I)/R
10002    CONTINUE
         LEN=R+SQRT(R**2.-RCY**2.)
      DO 10003 J=1,JMAX
        IF (ZP(J) .GT. LEN) THEN
          JMAX1=J-1
          GOTO 10004
        END IF
10003    CONTINUE
10004    CONTINUE
C
C.....EVALUATE VALUES FOR RMAX(J) AND NVI(J) AS Z INCREASES
C
         DO 4 JJ=0,JMAX
           IF(JJ .LE. JMAX1) THEN
             RMAX(JJ)=SQRT(R**2.-(R-ZP(JJ))**2.)
         DO 10005 I=0,IMAX
           IF(RP(I) .GE. RMAX(JJ)) THEN
             NVI(JJ)=I-1
             GOTO 10015
           END IF
10005        CONTINUE
         WRITE(6,*) 'ERROR : RMAX(J) IS GREATER THAN MAX. RP(J)'
10015        CONTINUE
           ELSE
             RMAX(JJ)=RCY
         DO 10006 I=0,IMAX
           IF(RP(I) .GE. RMAX(JJ)) THEN
             NVI(JJ)=I-1
```

```
            GOTO 10016
          END IF
10006       CONTINUE
        WRITE(6,*) 'ERROR : RMAX(J) IS GREATER THAN MAX. RP(J)'
10016       CONTINUE
          END IF
4       CONTINUE
C
C.........BOUNDARY CONDITION ON TOP
C
        DO 5 M=1,K
          DO 6 I=0,NVI(JMAX)+1
            TH(M,I,JMAX)=FIB
            FI(M,I,JMAX)=FIB
6         CONTINUE
5       CONTINUE
C
C.........BOUNDARY CONDITION ON SIDE
C
        DO 15 M=1,K
          TH(M,0,0)=FIB
          DO 16 J=1,JMAX-1
            II1=NVI(J)+1
           DO 17 I=II1,IMAX
              TH(M,I,J)=FIB
17          CONTINUE
16        CONTINUE
15      CONTINUE
C
C.........SET THE INITIAL CONDITION
C
        DO 7 J=1,JMAX-1
          DO 8 I=0,NVI(J)
            FI(0,I,J)=0.
            TH(0,I,J)=1.
8         CONTINUE
7       CONTINUE
C
C.......<<START OF COMPUTATION AND ITERATION PROCESS>>
C=================================================================
C
        DO 10 M=1,K
C
C.........INITIALIZING FOR EACH TIME-STEP
C
          DO 12 J2=1,JMAX-1
            DO 14 I2=0,NVI(J2)
              TH(M,I2,J2)=TH(M-1,I2,J2)
14          CONTINUE
12        CONTINUE
```

```
C
C.........KEEPING TRACK OF NUMBER OF ITERATIONS USING NUMB
C
          NUMB=1
C
C.........USE FLAG TO CHECK THE CONVERGANCE
C
100       FLAG=0
          DO 20 J=1,JMAX-1
C
C.........FOR I=0 USE THE FOLLOWING SET OF EQUATIONS
C
            IF(TH(M,1,J) .LT. 0.) THEN
                FI(M,1,J)=TH(M,1,J)
            ELSE
                FI(M,1,J)=0.
            END IF
            IF(TH(M,0,J-1) .LT. 0.) THEN
                FI(M,0,J-1)=TH(M,0,J-1)
            ELSE
                FI(M,0,J-1)=0.
            END IF
            IF(TH(M,0,J+1) .LT. 0.) THEN
                FI(M,0,J+1)=TH(M,0,J+1)
            ELSE
                FI(M,0,J+1)=0.
            END IF
C
C.........RS  REPRESENTS THE RIGHT HAND SIDE OF THE GOVERNING EQUATION
C
            RS=(RN(I)**2./DFO)*TH(M-1,0,J)+4.*FI(M,1,J)
     *         +(RN(I)/ZN(J))**2.*(FI(M,0,J-1)+FI(M,0,J+1))
            IF(RS .LT. 0.) THEN
              THMOJ=RS/(RN(I)**2./DFO-4.+(RN(I)/ZN(J))**2.+4.)
            ELSE
            THMOJ=RS/(RN(I)**2./DFO)
            END IF
C
C.........CHECK THE CONVERGANCE
C
            DIFF=ABS(TH(M,0,J)-THMOJ)
            IF(DIFF .GT. EPS) FLAG=1
            TH(M,0,J)=THMOJ
            IF(J .LE. JMAX1) THEN
          TSI=(RMAX(J)-RP(NVI(J)))/DR(NVI(J))
            ELSE
              TSI=1.
            END IF
            DO 30 I=1,NVI(J)
C
```

```
C.........CF1,CF2 AND CF3 ARE COEFFICIENTS OF THE GENERAL EQUATION
C           INTRODUCED IN CHAPTER FIVE.
C
                CF1=1.
                CF2=1.
                CF3=1.
                IF(RP(I) .GT. RMAX(J-1)) THEN
                  ETA1=(J*DZ(J)-R+SQRT(R**2.-RP(I)**2.))/DZ(J)
                  IF(ETA1 .LT. .0005) GO TO 115
                  IF(RP(I) .GT. RMAX(J+1)) THEN
C
C.........COMPUTING THE COEFFICIENTS WHEN BOTH LOWER AND UPPER J
C                 STRINGS ARE INTERSECTED BY THE CURVED BOUNDARY
C
                    ETA2=(R+SQRT(R**2.-RP(I)**2.)-J*DZ(J))/DZ(J)
                    CF1=1./(ETA2*ETA1)
                    CF2=2.*CF1
                    CF3=CF2
                  ELSE
C
C.........COMPUTING THE COEFFICIENTS WHEN THE LOWER J STRING IS
C                 INTERSECTED BY THE CURVED BOUNDARY
C
                    CF1=1./ETA1
                    CF3=2./(1.+ETA1)
                    CF2=CF3/ETA1
                  END IF
                ELSE
                  IF(RP(I) .GT. RMAX(J+1)) THEN
C
C.........COMPUTING THE COEFFICIENTS WHEN THE UPPER J STRING IS
C                 INTERSECTED BY THE CURVED BOUNDARY
C
                    ETA2=(R+SQRT(R**2.-RP(I)**2.)-J*DZ(J))/DZ(J)
                  IF(ETA2 .LT. EPS) GO TO 115
                    CF1=1./ETA2
                    CF2=2./(1.+ETA2)
                    CF3=CF2/ETA2
                  END IF
              END IF
                IF(TH(M,I+1,J) .LT. 0.) THEN
                  FI(M,I+1,J)=TH(M,I+1,J)
                ELSE
                  FI(M,I+1,J)=0.
                END IF
                IF(TH(M,I-1,J) .LT. 0.) THEN
                  FI(M,I-1,J)=TH(M,I-1,J)
                ELSE
                  FI(M,I-1,J)=0.
                END IF
```

```
                IF(TH(M,I,J+1) .LT. 0.) THEN
                  FI(M,I,J+1)=TH(M,I,J+1)
                ELSE
                  FI(M,I,J+1)=0.
                END IF
                IF(TH(M,I,J-1) .LT. 0.) THEN
                  FI(M,I,J-1)=TH(M,I,J-1)
                ELSE
                  FI(M,I,J-1)=0.
                END IF
                IF(I .LT. NVI(J)) THEN
C
C.........EUALUATING THE RIGHT HAND SIDE OF THE EQUATION
C
              RS=(RN(I)**2./DFO)*TH(M-1,I,J)+(1.-1./(2.*I))*FI(M,I-1,J)
     $               +(1.+1./(2.*I))*FI(M,I+1,J)+(RN(I)/ZN(J))**2.
     $               *(CF2*FI(M,I,J-1)+CF3*FI(M,I,J+1))
                ELSE
C
C.........THE APPROPRIATE FORM OF THE RIGHT HAND SIDE IF
C                 THE I STRING IS ITERSECTED BY THE CURVED BOUNDARY
C
                  IF(TSI .LT. .0005) GO TO 115
                  RS=(RN(I)**2./DFO)*TH(M-1,I,J)+
     *               (2./(TSI*(1.+TSI)))*(1.+1./
     *               (2*I))*FI(M,I+1,J)+(2./(1.+TSI))*(1.-TSI/(2.*I))*
     *               FI(M,I-1,J)+(RN(I)/ZN(J))**2.*(CF2*FI(M,I,J-1)
     *               +CF3*FI(M,I,J+1))
                END IF
                IF(RS .LT. 0.) THEN
                 IF(I .LT. NVI(J)) THEN
                  THMIJ=RS/(RN(I)**2./DFO+2.*(1.+(RN(I)/ZN(J))**2.*CF1))
                 ELSE
                  THMIJ=RS/(RN(I)**2./DFO+2.*((1.+(1.-TSI)/(2.*I))/TSI+
     *            (RN(I)/ZN(J))**2.*CF1))
                 END IF
                ELSE
                  THMIJ=RS/(RN(I)**2./DFO)
                END IF
                DIFF=ABS(TH(M,I,J)-THMIJ)
C
C.........CHECK THE CONVERGECE
C
                IF(DIFF .GT. 0.0005) FLAG=1
                TH(M,I,J)=THMIJ
                GO TO 30
115             TH(M,I,J)=FIB
30             CONTINUE
20          CONTINUE
            NUMB=NUMB+1
```

```
C
C.........CONTINUE THE ITERATION AND CHECK TO SEE IF MAXIMUM NUMBER
C               OF ITERATION IS EXEEDED
C
      IF(NUMB .GT. MAXNUM) THEN
            WRITE(6,*)'XXXXXX ERROR XXXXX'
            WRITE(6,*)'** MAX. NUMBER OF ITERATIONS EXCEEDED **'
            STOP
          END IF
C
          IF(FLAG .EQ. 1) GO TO 100
      DO 101 J=0,JMAX
      DO 101 I=0,NVI(J)
        IF(TH(M,I,J) .GT. 0.)GOTO 10
101   CONTINUE
    MF=M
    TIMEF=DFO*MF
    GOTO 1011
10      CONTINUE
C
1011    CONTINUE
      WRITE(6,*) '========================================'
      WRITE(6,*) ' **** CONVERGANCE  ACHIEVED ****'
      WRITE(6,*) '----------------------------------------'
      WRITE(6,*) '  FINAL TIME-INDEX:',MF
      WRITE(6,*) '  TOTAL SULIDIFICATION TIME:',TIMEF
      WRITE(6,*) '----------------------------------------'
CXXXXXXXXXXXXXXXXXXXXXXXXXXX
      IF (SWO11 .EQ. 1) THEN
CXXXXXXXXXXXXXXXXXXXXXXXXXXX
C--------------------------------------------
C:: THIS SECTION SENDS THE OUTPUT TO TAPE11 !
C--------------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
      WRITE(6,*)'  WRITING OUTPUT TO TAPE11'
      WRITE(11,*)' <<<< ENTHALPY DISTRIBUTION FOR >>>>'
      WRITE(11,*)' <<<<   TEAR-DROP GEOMETRY      >>>>'
      WRITE(11,*)' ----------------------------------'
C
        DO 40 M3=1,MF,3
         TFO=M3*DFO
         WRITE(11,23) TFO,M3
         DO 50 J3=JMAX,0,-1
           IND=NVI(J3)+1
         WRITE(11,25) (TH(M3,I3,J3),I3=0,IND)
50       CONTINUE
40      CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
```

```
C
Cxxxxxxxxxxxxxxxxxxxxxxxxxxx
      IF(SW012 .EQ. 1) THEN
Cxxxxxxxxxxxxxxxxxxxxxxxxxxx
C--------------------------------------
C:: THIS SECTION COMPUTES COORDINATES !
C:: OF THE INTERFACE                  !
C--------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
        DO 141 M=1,MF
          KK=0
          DO 142 J=1,JMAX-1
            DO 143 I=0,NVI(J)
             IF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *          .AND. TH(M,I,J+1) .EQ. 1.) THEN
              KK=KK+1
              DZ1(M,KK)=ZNP(J)+ZN(J)/2.-TH(M,I,J)*ZN(J)
              DR1(M,KK)=RNP(I)
             ELSEIF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *          .AND. TH(M,I-1,J) .EQ. 1.) THEN
              KK=KK+1
              DZ1(M,KK)=ZNP(J)
              DR1(M,KK)=RNP(I-1)+RN(I-1)/2.+TH(M,I,J)*RN(I)
             ELSEIF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *               .AND. TH(M,I,J-1) .EQ. 1.) THEN
              KK=KK+1
              DZ1(M,KK)=ZNP(J-1)+ZN(J-1)/2.+TH(M,I,J)*ZN(J)
              DR1(M,KK)=RNP(I)
             END IF
143         CONTINUE
142        CONTINUE
           KKM(M)=KK
141      CONTINUE
      WRITE(6,*) 'ENTER TIME INDEX TO BE USED FOR INT. LOCATION'
      WRITE(6,*) 'M = ? '
      READ(*,*) ML
      WRITE(6,*)'  WRITING OUTPUT TO TAPE12'
      WRITE(12,*) '<<<< RESULTS OF INTERFACE LOCATION >>>>'
      WRITE(12,*) '<<<< ---------------------------- >>>>'
      TFOIN=ML*DFO
      WRITE(12,*) 'FOURIER NO. = ',TFOIN
        DO 151 L=1,KKM(ML)
          WRITE(12,801) DR1(ML,L),DZ1(ML,L)
151     CONTINUE
C
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
Cxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

```
C-----------------------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
        DO 831 M=0,MF
          DO 831 I=0,NVI(JRMAX)
            IF(TH(M,I,JRMAX).GT. 0. .AND. TH(M,I,JRMAX) .LT. 1.) THEN
              RC(M)=RNP(I-1)+RN(I-1)/2.+TH(M,I,J)*RN(I)
C         RC(M)=(RP(I)-DR(I)/2.)/R+TH(M,I,J)
              IF(I .EQ. 0) RC(M)=TH(M,I,JRMAX)*RN(I)/2.
            END IF
831     CONTINUE
      WRITE(6,*)'  WRITING OUTPUT TO TAPE14'
      WRITE(14,*) '<<<< INTERFACE LOCATION ON RMAX >>>>'
      WRITE(14,*) '------------------------------------'
        DO 832 MM=0,MF,2
           FOUR=MM*DFO
           WRITE(14,301) RC(MM),FOUR
832     CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
Cxxxxxxxxxxxxxxxxxxxxxxxxxxx
      IF (SWO15 .EQ. 1) THEN
Cxxxxxxxxxxxxxxxxxxxxxxxxxxx
C-------------------------------------------
C:: THIS SECTION SENDS THE OUTPUT TO TAPE15 !
C-------------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
      WRITE(6,*)'  WRITING OUTPUT TO TAPE15'
      WRITE(15,*)' <<<< 2-D TEMPERATURE DISTRIBUTION FOR >>>>'
      WRITE(15,*)'   <<<<      TEAR-DROP GEOMETRY      >>>>'
      WRITE(15,*)'     ----------------------------------'
C
        DO 4015 M3=1,MF,3
         TFO=M3*DFO
         WRITE(15,2315) TFO,M3
         DO 5015 J3=N,0,-1
           IND=NVI(J3)+1
         WRITE(15,2515) (FI(M3,I3,J3),I3=0,IND)
5015     CONTINUE
4015    CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
Cxxxxxxxxxxxxxxxxxxxxxxxxxxx
      IF (SWO16 .EQ. 1) THEN
Cxxxxxxxxxxxxxxxxxxxxxxxxxxx
C-------------------------------------------
C:: THIS SECTION SENDS THE OUTPUT TO TAPE16 !
C-------------------------------------------
```

```
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
      WRITE(6,*) 'ENTER TIME INDEX TO BE USED FOR SURFER PROG.'
      WRITE(6,*) 'ML= ? '
      READ(*,*) ML
      WRITE(6,*)'WRITING OUTPUT TO TAPE16'
      WRITE(16,*)' <<<< 2-D TEMPERATURE DISTRIBUTION FOR >>>>'
      WRITE(16,*)' <<<<     USE IN SURFER PROGRAM    >>>>'
      WRITE(16,*)' ------------------------------------------'
C
      DO 1601 J=0,JMAX
        IND=NVI(J)+1
        DO 1602 I=0,IND
          WRITE(16,1611) RNP(I),ZNP(J),FI(ML,I,J)
1602    CONTINUE
1601  CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C::.......................................................................
      WRITE(6,*)                         ' ---------------------------------'
      WRITE(6,*)                         ' ***** LIST OF OUTPUT FILES ***** '
      WRITE(6,*)                         ' ---------------------------------'
      IF(SWO11 .EQ. 1 ) WRITE(6,*) '  ENTHALPY DISTRIBUTION   :: TAPE11'
      IF(SWO12 .EQ. 1 ) WRITE(6,*) '  INTERFACE LOCATION      :: TAPE12'
      IF(SWO13 .EQ. 1 ) WRITE(6,*) '  TEMP. DISTRIBUTION      :: TAPE13'
      IF(SWO14 .EQ. 1 ) WRITE(6,*) '  INTERFACE ON RMAX       :: TAPE14'
      IF(SWO15 .EQ. 1 ) WRITE(6,*) '  2-D TEMP. DISTRIBUTION :: TAPE15'
      IF(SWO16 .EQ. 1 ) WRITE(6,*) '  INPUT TO SURFER PROG.   :: TAPE16'
C........................................................................
C::   FORMAT STATEMENTS:
C---------------------
23       FORMAT(///2X,'FO=',F7.5,2X,'M=',I4)
25       FORMAT(1X,20(F6.3))
801      FORMAT(1X,F8.5,1X,F8.5)
802      FORMAT(1X,F8.5,1X,20F8.3)
301      FORMAT(2X,F8.5,2X,F8.5)
2315     FORMAT(///2X,'FO=',F7.5,2X,'M=',I4)
2515     FORMAT(1X,20(F6.3))
1611     FORMAT(1X,3(F8.4))
         STOP
         END
C
C=======================================================================
C
C   INPUT/OUPUT SUBROUTINE
    SUBROUTINE SITCAN3(IOF,SWRW)
C=======================================================================
C
C      TITLE:  SITCAN3
C
C       TASK:  THIS SUBROUTINE CAN BE USED IN TWO MODES (READ
```

```
C               MODE WHEN SWRW IS 1 AND WRITE MODE WHEN SWRW
C               IS 0) TO READ FROM (OR WRITE TO) TAPE IOF THE
C               DATA NECESSARY FOR PROGRAM CANDLV3.FOR.
C
C
C-----------------------------------------------------------------------
C     UPDATED: FEB. 20, 1987
C=======================================================================
C
C                                NOMENCLATURE
C
C   I/O...INPUT/OUTPUT
C     I: INPUT
C     O: OUTPUT
C     I/O: INPUT/OUTPUT
C   VAR...TYPE OF VARIABLE: INTEGER/REAL/CHARACTER
C     I: INTEGER
C     R: REAL
C     C: CHARACTER
C   DIM...DIMENSION:0,1,2,3,...
C
C-----------------------------------------------------------------------
C  VARNAME I/O VAR DIM  DESCRIPTION
C  ------- --- --- ---  ------------------------------------------------
C  CS       I    R   0    SOLID SPECIFIC HEAT OF PCM
C  DR       I    R   1    SPATIAL STEP-SIZE IN R DIRECTION
C  DZ       I    R   1    SPATIAL STEP-SIZA IN Z DIRECTION
C  DFO      -    R   0    NON-DIMENSIONAL TIME-STEP
C  EPS      I    R   0    MAXIMUM DEVIATION OF ENTHALPY AT ANY NODE
C                         FROM PREVIOUS ITERATION
C  FIB      I    R   0    SPECIFIED BOUNDARY TEMPERATURE
C  I        -    I   0    INDEX FOR R DIRECTION
C  IOF      P    I   0    INPUT/OUTPUT FILE NUMBER
C  IMAX     -    I   0    MAXIMUM VALUE OF INDEX I
C  J        -    I   0    INDEX IN Z DIRECTION
C  JMAX     I    I   0    MAXIMUM VALUE OF INDEX J
C  LAND     I    R   0    LATENT HEAT OF FUSION  OF PCM
C  MAXNUM   I    I   0    MAXIMUM NUMBER OF ITERATION
C  R        I    R   0    RADIUS OF SPHERICAL   SECTION
C  RCY      I    R   0    RADIUS OF CYLINDRICAL SECTION
C  SWO11    I    I   0    SWITCH FOR OUTPUT TO TAPE11 (ON:1,OFF:0)
C  SWO12    I    I   0    SWITCH FOR OUTPUT TO TAPE12 (ON:1,OFF:0)
C  SWO13    I    I   0    SWITCH FOR OUTPUT TO TAPE13 (ON:1,OFF:0)
C  SWO14    I    I   0    SWITCH FOR OUTPUT TO TAPE14 (ON:1,OFF:0)
C  TSAT     I    R   0    SATURATION TEMPERATURE OF PCM
C
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
C
C============================
C:: DECLARATION OF VARIABLES
```

```
C---------------------------
C:: COMMON VARIABLES:
C--------------------
      COMMON /PORT01/ IMAX,JMAX,K,MAXNUM
      INTEGER         IMAX,JMAX,K,MAXNUM
C---------------------------
      COMMON /PORT03/ SWO11,SWO12,SWO13,SWO14,SWO15,SWO16
      INTEGER         SWO11,SWO12,SWO13,SWO14,SWO15,SWO16
C---------------------------
      COMMON /PORT04/ R,RCY
      REAL            R,RCY
C---------------------------
      COMMON /PORT05/ DR,      DZ
      REAL            DR(0:60),DZ(0:60)
C---------------------------
      COMMON /PORT06/ TSAT,LAND,CS
      REAL            TSAT,LAND,CS
C---------------------------
      COMMON /PORT07/ FIB,EPS
      REAL            FIB,EPS
C---------------------------
      COMMON /PORT08/ DFO
      REAL            DFO
C---------------------------
C:: PASS VARIABLES:
C------------------
      INTEGER         SWRW
      INTEGER         IOF
C---------------------------
      IF(SWRW .EQ. 0) THEN
        WRITE(IOF,3000)
        WRITE(IOF,3001)
        WRITE(IOF,*)
        WRITE(IOF,*)
      ELSE IF(SWRW .EQ. 1) THEN
        READ (IOF,*)
        READ (IOF,*)
        READ (IOF,*)
        READ (IOF,*)
      END IF
      IF(SWRW .EQ. 0) THEN
        WRITE(IOF,2000)
        WRITE(IOF,*) '<< LOOP CONTROLLERS >>'
        WRITE(IOF,2000)
        WRITE(IOF,1000)  IMAX
        WRITE(IOF,1001)  JMAX
        WRITE(IOF,10016) K
        WRITE(IOF,10017) MAXNUM
        WRITE(IOF,10018) DFO
        WRITE(IOF,10019) FIB
```

```
      WRITE(IOF,10020) EPS
      WRITE(IOF,*)
    ELSE IF(SWRW .EQ. 1) THEN
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,1010) IMAX
      READ(IOF,1011) JMAX
      READ(IOF,1011) K
      READ(IOF,1011) MAXNUM
      READ(IOF,1012) DFO
      READ(IOF,1012) FIB
      READ(IOF,1012) EPS
      READ(IOF,*)
    END IF
    IF(SWRW .EQ. 0) THEN
      WRITE(IOF,2000)
      WRITE(IOF,*) '<< SWITCHES FOR OUTPUT FILES >>'
      WRITE(IOF,2000)
      WRITE(IOF,1007)  SWO11
      WRITE(IOF,1008)  SWO12
      WRITE(IOF,1009)  SWO13
      WRITE(IOF,10010) SWO14
      WRITE(IOF,10011) SWO15
      WRITE(IOF,10012) SWO16
      WRITE(IOF,*)
    ELSE IF(SWRW .EQ. 1) THEN
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,1017)  SWO11
      READ(IOF,1017)  SWO12
      READ(IOF,1017)  SWO13
      READ(IOF,1017)  SWO14
      READ(IOF,1017)  SWO15
      READ(IOF,1017)  SWO16
      READ(IOF,*)
    END IF
    IF(SWRW .EQ. 0) THEN
      WRITE(IOF,2000)
      WRITE(IOF,*) '<< PHYSICAL PROPERTIES >>'
      WRITE(IOF,2000)
      WRITE(IOF,10013) TSAT
      WRITE(IOF,10014) LAND
      WRITE(IOF,10015) CS
      WRITE(IOF,*)
    ELSE IF(SWRW .EQ. 1)THEN
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,*)
```

```
      READ(IOF,1012) TSAT
      READ(IOF,1012) LAND
      READ(IOF,1012) CS
      READ(IOF,*)
    END IF
    IF(SWRW .EQ. 0) THEN
      WRITE(IOF,2000)
      WRITE(IOF,*) '<< GEOMETRICAL PARAMETERS >>'
      WRITE(IOF,2000)
      WRITE(IOF,1002) RCY
      WRITE(IOF,1003) R
    ELSE IF(SWRW .EQ. 1)THEN
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,1012) RCY
      READ(IOF,1013) R
    END IF
C..... READ/WRITE VALUES FOR DR & DZ
    IF(SWRW .EQ. 0) THEN
      WRITE(IOF,2001)
      WRITE(IOF,*) 'DR    ::'
      WRITE(IOF,2001)
      WRITE(IOF,1006) (DR(I1),I1=0,9)
      IF(IMAX .GT. 9) THEN
      WRITE(IOF,1006) (DR(I2),I2=10,19)
      END IF
      IF(IMAX .GT.19) THEN
      WRITE(IOF,1006) (DR(I3),I3=20,29)
      END IF
      IF(IMAX .GT.29) THEN
      WRITE(IOF,1006) (DR(I4),I4=30,39)
      END IF
      IF(IMAX .GT. 39) WRITE(6,*) '<< ERROR >> : IMAX GREATER THAN 39'
    ELSE IF(SWRW .EQ. 1) THEN
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,*)
      READ(IOF,1006) (DR(I1),I1=0,9)
      IF(IMAX .GT. 9) THEN
      READ(IOF,1006) (DR(I2),I2=10,19)
      END IF
      IF(IMAX .GT.19) THEN
      READ(IOF,1006) (DR(I3),I3=20,29)
      END IF
      IF(IMAX .GT.29) THEN
      READ(IOF,1006) (DR(I4),I4=30,39)
      END IF
      IF(IMAX .GT. 39) WRITE(6,*) '<< ERROR >> : IMAX GREATER THAN 39'
    END IF
```

```
      IF(SWRW .EQ. 0) THEN
        WRITE(IOF,2001)
        WRITE(IOF,*) 'DZ    ::'
        WRITE(IOF,2001)
        WRITE(IOF,1006) (DZ(J1),J1=0,9)
        IF(JMAX .GT. 9) THEN
        WRITE(IOF,1006) (DZ(J2),J2=10,19)
        END IF
        IF(JMAX .GT.19) THEN
        WRITE(IOF,1006) (DZ(J3),J3=20,29)
        END IF
        IF(JMAX .GT.29) THEN
        WRITE(IOF,1006) (DZ(J4),J4=30,39)
        END IF
        IF(JMAX .GT. 39) WRITE(6,*) '<< ERROR >> : JMAX GREATER THAN 39'
      ELSE IF(SWRW .EQ. 1) THEN
        READ(IOF,*)
        READ(IOF,*)
        READ(IOF,*)
        READ(IOF,1006) (DZ(J1),J1=0,9)
        IF(JMAX .GT. 9) THEN
        READ(IOF,1006) (DZ(J2),J2=10,19)
        END IF
        IF(JMAX .GT.19) THEN
        READ(IOF,1006) (DZ(J3),J3=20,29)
        END IF
        IF(JMAX .GT.29) THEN
        READ(IOF,1006) (DZ(J4),J4=30,39)
        END IF
        IF(JMAX .GT. 39) WRITE(6,*) '<< ERROR >> : JMAX GREATER THAN 39'
      END IF
 1000     FORMAT(1X,'IMAX  ','=',1X,'[',I2,']')
 1010     FORMAT(1X,6X      ,1X ,1X,1X ,I2    )
 1001     FORMAT(1X,'JMAX  ','=',1X,'[',I2,']')
 1011     FORMAT(1X, 6X     ,1X ,1X,1X ,I2    )
 1012     FORMAT(1X, 6X     ,1X ,1X,1X ,F10.6    )
 3000     FORMAT(20X,'<<< INPUT FILE FOR PROGRAM CANDLV3.FOR >>>')
 3001     FORMAT(20X,'*****************************************')
 2000     FORMAT(16('=='))
 2001     FORMAT( 5('--'))
 1002     FORMAT(1X,'RCY   ','=',1X,'[',F10.6,']')
 1003     FORMAT(1X,'R     ','=',1X,'[',F10.6,']')
 1013     FORMAT(1X, 6X     ,1X ,1X,1X ,F10.6    )
 1006     FORMAT(15(F7.4))
 1007     FORMAT(1X,'SWO11 ','=',1X,'[',I1,']')
 1008     FORMAT(1X,'SWO12 ','=',1X,'[',I1,']')
 1009     FORMAT(1X,'SWO13 ','=',1X,'[',I1,']')
10010     FORMAT(1X,'SWO14 ','=',1X,'[',I1,']')
10011     FORMAT(1X,'SWO15 ','=',1X,'[',I1,']')
10012     FORMAT(1X,'SWO16 ','=',1X,'[',I1,']')
```

```
10013     FORMAT(1X,'TSAT   ','=',1X,'[',F10.6,']')
10014     FORMAT(1X,'LAND   ','=',1X,'[',F10.6,']')
10015     FORMAT(1X,'CS     ','=',1X,'[',F10.6,']')
10016     FORMAT(1X,'K      ','=',1X,'[',I2,']')
10017     FORMAT(1X,'MAXNUM','=',1X,'[',I2,']')
10018     FORMAT(1X,'DFO    ','=',1X,'[',F10.6,']')
10019     FORMAT(1X,'FIB    ','=',1X,'[',F10.6,']')
10020     FORMAT(1X,'EPS    ','=',1X,'[',F10.6,']')
1017      FORMAT(1X,6X       ,1X ,1X,1X ,I1     )
      RETURN
      END
```

```
C=======================================================================
C
C      TITLE:  HEATA2.FOR(HEATA VERSION 2)
C
C       TASK:  THIS PROGRAM EMPLOYS THE ENTHALPY METHOD FOR
C              SOLVING HEAT TRANSFER PROBLEMS INVOLVING PHASE
C              CHANGE. THE GAUSS-SEIDEL ITERATIVE SCHEME IS USED
C              TO SOLVE THE SYSTEM OF SIMOULTANEOUS EQUATIONS.
C              VARIOUS RESULTS (SUCH AS ENTHALPY DISTRIBUTION,
C              TEMPERATURE DISTRIBUTION, INTERFACE LOCATION...)
C              CAN BE OBTAINED (AS OUTPUTS OF THE PROGRAM) BY
C              SETTING THE APPROPRIATE SWITCHES EQUAL TO 1.
C
C   GEOMETRY:  CIRCULAR CYLINDER WITH RADIUS R AND HEIGHT H
C
C
C-----------------------------------------------------------------------
C     UPDATED: DEC  9, 1987
C=======================================================================
C
C                              NOMENCLATURE
C
C   I/O...INPUT/OUTPUT
C     I: INPUT
C     O: OUTPUT
C     I/O: INPUT/OUTPUT
C   VAR...TYPE OF VARIABLE: INTEGER/REAL/CHARACTER
C     I: INTEGER
C     R: REAL
C     C: CHARACTER
C   DIM...DIMENSION  1,2,3,...
C
C-----------------------------------------------------------------------
C  VARNAME I/O VAR DIM  DESCRIPTION
C  ------- --- --- ---  -----------------------------------------
C  DR      O    R   2    RADIAL POSITION USED IN COORD. OF
C                        INTERFACE COMPUTATION SECTION
C  DZ      O    R   2    VERTICAL POSITION USED IN COORD. OF
C                        INTERFACE COMPUTATION SECTION
C  DFO     -    R   0    NON-DIMENSIONAL TIME-STEP
C  FI      I/O  R   3    NON-DIMENSIONAL TEMPERATURE
C  FIB     I    R   0    SPECIFIED BOUNDARY TEMPERATURE
C  FO      -    R   0    NON-DIMENSIONAL TIME (FOURIER NO.)
C  FOA     O    R   1    TIME VARIABLE USED IN FROZ. FRACTION SECTION
C  FLAG    -    I   0    INDICATOR FOR CONVERGENCE
C  I       -    I   0    INDEX IN R DIRECTION
C  J       -    I   0    INDEX IN Z DIRECTION
C  K       I    I   0    NUMBER OF TIME-STEPS
C  KKM     O    I   1    NUMBER OF POINTS LOCATED ON THE INTERFACE
C                        FOR A GIVEN TIME-LEVEL
```

```
C  M        -     I   0     TIME INDEX
C  MAXNUM   I     I   0     MAXIMUM NUMBER OF ITERATION
C  N        I     I   0     TOTAL NUMBER OF POINTS IN R DIR.
C  NN       I     I   0     TOTAL NUMBER OF POINTS IN Z DIR.
C  NTL      -     I   0     TOTAL NUMBER OF CONTROL VOLUMES
C  NUMB     -     I   0     ITERATION COUNTER
C  RN       -     R   0     NON-DIMENSIONAL DELTA R
C  SWO51    I     I   0     SWITCH FOR OUTPUT TO TAPE51 (ON:1,OFF:0)
C  SWO52    I     I   0     SWITCH FOR OUTPUT TO TAPE52 (ON:1,OFF:0)
C  SWO53    I     I   0     SWITCH FOR OUTPUT TO TAPE53 (ON:1,OFF:0)
C  TEMP     -     R   2     TEMPERATURE ARRAY USED IN TEMPERATURE
C                           DISTRIBUTION SCTION
C  TCSCV    -     R   0     TOTAL NUMBER OF COMPLETELY SOLID C.V.'S
C  TPSCV    -     R   0     TOTAL NUMBER OF PARTIALLY  SOLID C.V.'S
C  TH       I/O   R   3     NON-DIMENSIONAL ENTHALPY
C
C23456789012345678901234567890123456789012345678901234567890123456789012
C
C============================
      INTEGER I,J,NN,N,M,K,NUMB,FLAG,MAXNUM
      INTEGER SWO51,SWO52,SWO53
      INTEGER KKM(18)
C--------------------------
      REAL    FO,DFO,FIB,RN,PSCV,TCSCV,TPSCV
      REAL    FFCT(0:18),FOA(0:18)
      REAL    DR(0:18,0:500),DZ(0:18,0:500)
      REAL    TH(0:18,0:10,0:50),FI(0:18,0:10,0:50)
C
C.........ENTERING THE PARAMETERS OF THE PROBLEM
C
      DATA SWO51/1/
      DATA SWO52/1/
      DATA SWO53/1/
      MAXNUM=10
      K=18
      N=5
      NN=40
      DFO=0.025
      RN=0.2
      FIB=-1.0
C
C.....  BOUNDARY CONDITION ON TOP AND BOTTOM
C
      DO 1 M1=1,K
         DO 2 II=0,N
           FI(M1,II,NN)=FIB
           TH(M1,II,NN)=FIB
           FI(M1,II,0)=FIB
           TH(M1,II,0)=FIB
2        CONTINUE
```

```
1       CONTINUE
C
C.....  BOUNDARY CONDITION ON SIDE
C
        DO 3 M2=1,K
          DO 4 JJ=1,NN-1
            FI(M2,N,JJ)=FIB
            TH(M2,N,JJ)=FIB
4         CONTINUE
3       CONTINUE
C
C.....  INITIAL CONDITION
C
C
C.......<<START OF COMPUTATION AND ITERATION PROCESS>>
C=================================================================
C
        DO 5 I1=0,N
          DO 6 J1=0,NN
            FI(0,I1,J1)=0.
            TH(0,I1,J1)=1.
6         CONTINUE
5       CONTINUE
        DO 10 M=1,K
C
C.....  INITIALIZING FOR EACH TIME STEP
C
          DO 12 I2=0,N-1
            DO 14 J2=1,NN-1
              TH(M,I2,J2)=TH(M-1,I2,J2)
14          CONTINUE
12        CONTINUE
C
C.........KEEPING TRACK OF NUMBER OF ITERATIONS USING NUMB
C
          NUMB=1
C
C.........USE FLAG TO CHECK THE CONVERGANCE
C
100       FLAG=0
          DO 20 I=0,N-1
            DO 30 J=1,NN-1
C
C.........FOR I=0 USE THE FOLLOWING SET OF EQUATIONS
C
              IF(I .EQ. 0) THEN
                IF(TH(M,I+1,J) .LT. 0.) THEN
                  FI(M,I+1,J)=TH(M,I+1,J)
                ELSE
                  FI(M,I+1,J)=0.
```

```
                    END IF
                    IF(TH(M,I,J-1) .LT. 0.) THEN
                      FI(M,I,J-1)=TH(M,I,J-1)
                    ELSE
                      FI(M,I,J-1)=0.
                    END IF
                    IF(TH(M,I,J+1) .LT. 0.) THEN
                      FI(M,I,J+1)=TH(M,I,J+1)
                    ELSE
                      FI(M,I,J+1)=0.
                    END IF
C
C.........RS  REPRESENTS THE RIGHT HAND SIDE OF THE GOVERNING EQUATION
C
                    RS=(RN**2./DFO)*TH(M-1,I,J)+4.*FI(M,I+1,J)+FI(M,I,J-1)
     *              +FI(M,I,J+1)
                    IF(RS .LT. 0) THEN
                      THMOJ=RS/(RN**2./DFO+6.)
                    ELSE
                      THMOJ=RS/(RN**2./DFO)
                    END IF
C
C.........CHECK THE CONVERGANCE
C
                    DIFF=ABS(TH(M,0,J)-THMOJ)
                    IF(DIFF .GT. 0.005) FLAG=1
                    TH(M,0,J)=THMOJ
C
C.........USE THE FOLLOWING BRANCH WHEN I IS NOT ZERO
C
                  ELSE
                    IF(TH(M,I+1,J) .LT. 0.) THEN
                      FI(M,I+1,J)=TH(M,I+1,J)
                    ELSE
                      FI(M,I+1,J)=0.
                    END IF
                    IF(TH(M,I-1,J) .LT. 0.) THEN
                      FI(M,I-1,J)=TH(M,I-1,J)
                    ELSE
                      FI(M,I-1,J)=0.
                    END IF
                    IF(TH(M,I,J+1) .LT. 0.) THEN
                      FI(M,I,J+1)=TH(M,I,J+1)
                    ELSE
                      FI(M,I,J+1)=0.
                    END IF
                    IF(TH(M,I,J-1) .LT. 0.) THEN
                      FI(M,I,J-1)=TH(M,I,J-1)
                    ELSE
                      FI(M,I,J-1)=0.
```

```
                  END IF
                  RS=(RN**2./DFO)*TH(M-1,I,J)+(1.-1./(2.*I))*FI(M,I-1,J)
     *               +(1.+1./(2.*I))*FI(M,I+1,J)+FI(M,I,J-1)+FI(M,I,J+1)
                  IF(RS .LT. 0.) THEN
                    THMIJ=RS/(RN**2./DFO+4.)
                  ELSE
                    THMIJ=RS/(RN**2./DFO)
                  END IF
                  DIFF=ABS(TH(M,I,J)-THMIJ)
C
C.........CHECK THE CONVERGECE
C
                  IF(DIFF .GT. 0.005) FLAG=1
                  TH(M,I,J)=THMIJ
                END IF
30            CONTINUE
20          CONTINUE
            NUMB=NUMB+1
C
C.........CONTINUE THE ITERATION AND CHECK TO SEE IF MAXIMUM NUMBER
C               OF ITERATION IS EXEEDED
C
            IF(NUMB .GT. MAXNUM) THEN
              WRITE(6,*)'XXXXXX ERROR XXXXX'
              WRITE(6,*)'** MAX. NUMBER OF ITERATIONS EXCEEDED **'
              STOP
            END IF
            IF(FLAG .EQ. 1) GO TO 100
      DO 101 I=0,N
      DO 101 J=0,NN
        IF(TH(M,I,J) .GE. 0.)GOTO 10
101   CONTINUE
      MF=M
      GOTO 1011
10      CONTINUE
C
1011   CONTINUE
       WRITE(6,*)'PROGRAM HEATA CONVERGED'
C
       IF(SW051 .EQ. 1) THEN
C--------------------------------------------
C:: THIS SECTION SENDS THE OUTPUT TO TAPE51 !
C--------------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
       WRITE(6,*)'WRITING OUTPUT TO TAPE51'
       WRITE(51,*)' ::   TAPE51:: ----------------------------------'
       WRITE(51,*)' <<<< ENTHALPY DISTRIBUTION IN CYLINDER >>>>'
       WRITE(51,*)' ---------------------------------------------'
         DO 40 M3=1,MF
           TFO=M3*DFO
```

```
          WRITE(51,23) TFO
23        FORMAT(///2X,'FO=',F7.5)
          DO 50 J3=0,NN
            WRITE(51,25) (TH(M3,I3,J3),I3=0,N)
25          FORMAT(6(1X,F6.3))
50        CONTINUE
40      CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
C
      IF(SWO52 .EQ. 1) THEN
C-------------------------------------
C:: THIS SECTION COMPUTES COORDINATES !
C;; OF THE INTERFACE                 !
C-------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
        DO 141 M=1,MF
          KK=0
          DO 142 J=1,NN-1
            DO 143 I=0,N-1
              IF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *           .AND. TH(M,I,J+1) .EQ. 1.) THEN
                KK=KK+1
                DZ(M,KK)=RN/2.+J*RN-TH(M,I,J)*RN
                DR(M,KK)=I*RN
              ELSEIF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *                .AND. TH(M,I-1,J) .EQ. 1.) THEN
                KK=KK+1
                DZ(M,KK)=J*RN
                DR(M,KK)=RN/2.+(I-1)*RN+TH(M,I,J)*RN
              ELSEIF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *                .AND. TH(M,I,J-1) .EQ. 1.) THEN
                KK=KK+1
                DZ(M,KK)=RN/2.+(J-1)*RN+TH(M,I,J)*RN
                DR(M,KK)=I*RN
              END IF
143         CONTINUE
142       CONTINUE
          KKM(M)=KK
141     CONTINUE
      WRITE(6,*)'WRITING OUTPUT TO TAPE52'
      WRITE(52,*)' ::   TAPE52:: ----------------------------------'
      WRITE(52,*)'<<<< RESULTS OF INTERFACE LOCATION >>>>'
      WRITE(52,*)' ----------------------------------------------'
        WRITE(6,*) 'ENTER TIME INDEX FOR INTERFACE LOCATION CURVE'
        WRITE(6,*) 'M = ?'
        READ(*,*) MINT
        DO 151 L=1,KKM(MINT)
        WRITE(52,801) DR(MINT,L),DZ(MINT,L)
```

```
801     FORMAT(1X,F8.5,1X,F8.5)
151      CONTINUE
C
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
C
      IF(SWO53 .EQ. 1) THEN
C-----------------------------
C:: THIS SECTION CALCULATES  !
C;; THE FROZEN FRACTION      !
C-----------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
        NTL=N*NN
        DO 201 M=1,MF
C
C.......INITIALIZING NUMBER OF COMPLETELY SOLID AND
C               PARTIALLY SOLID CONTROL VOLUMES
C
        TCSCV=N
        TPSCV=0.
          DO 202 I=0,N
            DO 203 J=1,NN-1
C
C.........CHECK TO SEE IF THE C.V. IS COMPLETELY SOLID
C
             IF(TH(M,I,J) .LE. 0.) THEN
C
C.........CHECK TO SEE IF THE C.V. IS A HALF CONTROL VOLUME
C
              IF(I .EQ. 0 .OR. I .EQ. N)THEN
                TCSCV=TCSCV+0.5
              ELSE
                TCSCV=TCSCV+1.
              END IF
C
C.........CHECK TO SEE IF THE C.V. IS PARTIALLY SOLID
C
              ELSEIF(TH(M,I,J) .GT. 0. .AND.
     *               TH(M,I,J) .LT. 1.)THEN
C
C.........CHECK TO SEE IF THE C.V. IS A HALF CNTROL VOLUME
C
                IF(I .EQ. 0 .OR. I .EQ. N)THEN
                  PSCV=(1.-TH(M,I,J))/2.
                ELSE
                  PSCV=1.-TH(M,I,J)
                END IF
                  TPSCV=TPSCV+PSCV
```

```
              END IF
203           CONTINUE
202         CONTINUE
C
C.........COMPUTE THE FROZEN FRACTION FOR A GIVEN TIME
C
              FFCT(M)=(TCSCV+TPSCV)/NTL
              FOA(M)=M*DFO
201       CONTINUE
        WRITE(6,*)'WRITING OUTPUT TO TAPE53'
        WRITE(53,*)' ::   TAPE53:: ----------------------------------'
        WRITE(53,*)'<<<< RESULTS OF FROZEN FRACTION >>>>'
        WRITE(53,*)' ------------------------------------------------'
          DO 211 M=1,MF
            WRITE(53,802) FOA(M),FFCT(M)
802       FORMAT(1X,F9.6,2X,F9.6)
211       CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
        END IF
C
        WRITE(6,*)                    ' ---------------------------------'
        WRITE(6,*)                    ' ***** LIST OF OUTPUT FILES *****'
        WRITE(6,*)                    ' ---------------------------------'
        IF(SWO51 .EQ. 1 ) WRITE(6,*) '  ENTHALPY DISTRIBUTION :: TAPE51'
        IF(SWO52 .EQ. 1 ) WRITE(6,*) '  INTERFACE LOCATION    :: TAPE52'
        IF(SWO53 .EQ. 1 ) WRITE(6,*) '  FROZEN FRACTION       :: TAPE53'
          STOP
          END
```

```
C=====================================================================
C
C       TITLE:  HEAT1D.FOR
C
C        TASK:  THIS PROGRAM EMPLOYS THE ENTHALPY METHOD FOR
C               SOLVING HEAT TRANSFER PROBLEMS INVOLVING PHASE
C               CHANGE. THE GAUSS-SEIDEL ITERATIVE SCHEME IS USED
C               TO SOLVE THE SYSTEM OF SIMOULTANEOUS EQUATIONS.
C               VARIOUS RESULTS (SUCH AS ENTHALPY DISTRIBUTION,
C               TEMPERATURE DISTRIBUTION, INTERFACE LOCATION...)
C               CAN BE OBTAINED (AS OUTPUTS OF THE PROGRAM) BY
C               SETTING THE APPROPRIATE SWITCHES EQUAL TO 1.
C
C    GEOMETRY:  LONG CIRCULAR CYLINDER (1-D APPROXIMATION)
C
C       INPUT:  FOR THIS PROGRAM ALL INPUT PROVIDED THROUGH
C               DATA STATEMENTS
C
C
C---------------------------------------------------------------------
C     UPDATED: DEC.  9, 1987
C=====================================================================
C
C                              NOMENCLATURE
C
C    I/O...INPUT/OUTPUT
C      I: INPUT
C      O: OUTPUT
C      I/O: INPUT/OUTPUT
C    VAR...TYPE OF VARIABLE: INTEGER/REAL/CHARACTER
C      I: INTEGER
C      R: REAL
C      C: CHARACTER
C    DIM...DIMENSION:0,1,2,3,...
C
C---------------------------------------------------------------------
C  VARNAME I/O VAR DIM  DESCRIPTION
C  ------- --- --- ---  ----------------------------------------
C  DR      -    R   1    SPATIAL STEP-SIZE IN R DIRECTION
C                        USED IN TEMP. DISTRIBUTION SECTION
C  DFO     I    R   0    NON-DIMENSIONAL TIME-STEP
C  EPS     I    R   0    MAXIMUM DEVIATION OF ENTHALPY AT ANY NODE
C                        FROM PREVIOUS ITERATION
C  FI      I/O  R   3    NON-DIMENSIONAL TEMPERATURE
C  FIB     I    R   0    SPECIFIED BOUNDARY TEMPERATURE
C  FO      -    R   0    NON-DIMENSIONAL TIME (FOURIER NO.)
C  FLAG    -    I   0    INDICATOR FOR CONVERGENCE
C  I       -    I   0    INDEX IN R DIRECTION
C  N       I    I   0    MAXIMUM VALUE OF INDEX I
C  K       I    I   0    NUMBER OF TIME-STEPS
```

```
C  M        -     I    0     TIME INDEX
C  MAXNUM   I     I    0     MAXIMUM NUMBER OF ITERATION
C  MF       -     I    0     VALUE OF TIME INDEX AT INSTANT
C                             CENTER FREEZES
C  NUMB     -     I    0     ITERATION COUNTER
C  RN       I     R    0     NON-DIMENSIONAL DELTA R
C  SWO41    I     I    0     SWITCH FOR OUTPUT TO TAPE41 (ON:1,OFF:0)
C  SWO42    I     I    0     SWITCH FOR OUTPUT TO TAPE42 (ON:1,OFF:0)
C  SWO33    I     I    0     SWITCH FOR OUTPUT TO TAPE33 (ON:1,OFF:0)
C  TEMP1..  O     R    2     TEMPERATURE VARIABLE USED IN TEMPERATURE
C                             DISTRIBUTION SCTION
C  TH       I/O   R    3     NON-DIMENSIONAL ENTHALPY
C  TIMEF    -     R    0     VALUE OF FOURIER NUMBER AT THE INSTANT
C                             CENTER FREEZES
C
C23456789012345678901234567890123456789012345678901234567890123456789012
C
C=========================
      INTEGER I,M,N,K,NUMB,FLAG,MAXNUM
      INTEGER SWO41,SWO42
C-------------------------
      REAL    FO,DFO,FIB,RN
      REAL    TEMP,TEMP1,TEMP2,TEMP3,TEMP4
      REAL    DR(0:41)
      REAL    TH(0:500,0:41),FI(0:500,0:41)
C
C.........ENTERING THE PARAMETERS OF THE PROBLEM
C
      DATA SWO41/1/
      DATA SWO42/1/
      DATA EPS  /.0005/
      DATA K,N,FIB/499,10,-0.3/
      DATA MAXNUM/25/
      DATA RN,DFO/.1,.01/
C
C.......BOUNDARY CONDITION ON SIDE
C
     DO 3 M2=1,K
        FI(M2,N)=FIB
        TH(M2,N)=FIB
3     CONTINUE
C
C.......INITIAL CONDITION
C
      DO 5 I1=0,N
        FI(0,I1)=0.
        TH(0,I1)=1.
5     CONTINUE
      DO 10 M=1,K
C
```

```
C.......INITIALIZING FOR EACH TIME STEP
C
          DO 12 I2=0,N-1
              TH(M,I2)=TH(M-1,I2)
12        CONTINUE
          NUMB=1
100       FLAG=0
          DO 20 I=0,N-1
              IF(I .EQ. 0) THEN
                IF(TH(M,I+1) .LT. 0.) THEN
                  FI(M,I+1)=TH(M,I+1)
                ELSE
                  FI(M,I+1)=0.
                END IF
                RS=(RN**2./DFO)*TH(M-1,I)+4.*FI(M,I+1)
                IF(RS .LT. 0) THEN
                  THMO=RS/(RN**2./DFO+4.)
                ELSE
                  THMO=RS/(RN**2./DFO)
                END IF
                DIFF=ABS(TH(M,0)-THMO)
                IF(DIFF .GT. EPS) FLAG=1
                TH(M,0)=THMO
              ELSE
                IF(TH(M,I+1) .LT. 0.) THEN
                  FI(M,I+1)=TH(M,I+1)
                ELSE
                  FI(M,I+1)=0.
                END IF
                IF(TH(M,I-1) .LT. 0.) THEN
                  FI(M,I-1)=TH(M,I-1)
                ELSE
                  FI(M,I-1)=0.
                END IF
                RS=(RN**2./DFO)*TH(M-1,I)+(1.-1./(2.*I))*FI(M,I-1)
     *             +(1.+1./(2.*I))*FI(M,I+1)
                IF(RS .LT. 0.) THEN
                  THMI=RS/(RN**2./DFO+2.)
                ELSE
                  THMI=RS/(RN**2./DFO)
                END IF
                DIFF=ABS(TH(M,I)-THMI)
                IF(DIFF .GT. EPS) FLAG=1
                TH(M,I)=THMI
              END IF
20        CONTINUE
          NUMB=NUMB+1
          IF(NUMB .GT. MAXNUM) THEN
            WRITE(6,*)'XXXXXX ERROR XXXXX'
            WRITE(6,*)'** MAX. NUMBER OF ITERATIONS EXCEEDED **'
```

```
          STOP
        END IF
C
        IF(FLAG .EQ. 1) GO TO 100
      DO 101 I=0,N-1
        IF(TH(M,I) .GT. 0.)GOTO 10
101   CONTINUE
    MF=M
    TIMEF=DFO*MF
    GOTO 1011
10      CONTINUE
C
1011    CONTINUE
      WRITE(6,*) '========================================'
      WRITE(6,*) ' **** CONVERGANCE  ACHIEVED ****'
      WRITE(6,*) '----------------------------------------'
      WRITE(6,*) '  FINAL TIME-INDEX:',MF
      WRITE(6,*) '  TOTAL SULIDIFICATION TIME:',TIMEF
      WRITE(6,*) '----------------------------------------'
Cxxxxxxxxxxxxxxxxxxxxxxxxxxxx
C
      IF (SWO41 .EQ. 1) THEN
C-------------------------------------------
C:: THIS SECTION SENDS THE OUTPUT TO TAPE11 !
C-------------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
      WRITE(6,*)'WRITING OUTPUT TO TAPE11'
      WRITE(41,*)' ::   TAPE41:: ---------------------'
      WRITE(41,*)' <<<< ENTHALPY DISTRIBUTION FOR >>>>'
      WRITE(41,*)' <<<<   THE CYLINDER            >>>>'
      WRITE(41,*)' -----------------------------------'
C
        DO 40 M3=1,MF
          TFO=M3*DFO
          WRITE(41,23) TFO      ,M3
23        FORMAT(2X,'FO=',F7.5,2X,I5)
            WRITE(41,25) (TH(M3,I3),I3=0,N)
25         FORMAT(15(1X,F6.3))
40      CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
C
      IF(SWO42 .EQ. 1) THEN
C-------------------------------------
C:: THIS SECTION PROVIDES DATA FOR    !
C;; TEMPERATURE DISTRIBUTION          !
C-------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
```

```
C
      WRITE(6,*)'WRITING OUTPUT TO TAPE42'
      WRITE(42,*)' ::   TAPE42:: ----------------------------------'
      WRITE(42,*)' <<<< TEMPERATURE DISTRIBUTION FOR  CYLINDER >>>>'
      WRITE(42,*)' <<<<   WITH FIXED TEMP. BOUNDARY CONDITION   >>>>'
      WRITE(42,*)' ----------------------------------------------'
C.....CALCULATIONS FOR TEMP. DISTRIBUTION DATA
        DO 201 I=0,N
          DR(I)=(0.5+I)*RN
          DR(0)=0.
          DR(N)=1.
        TEMP=FI(MF,I)
        WRITE(42,*) DR(I),TEMP
C801    FORMAT(1X,F8.6,2X,F8.6)
201     CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
      WRITE(6,*)                    ' ---------------------------------'
      WRITE(6,*)                    ' ***** LIST OF OUTPUT FILES *****'
      WRITE(6,*)                    ' ---------------------------------'
      IF(SWO41 .EQ. 1 ) WRITE(6,*) '  ENTHALPY DISTRIBUTION :: TAPE41'
      IF(SWO42 .EQ. 1 ) WRITE(6,*) '  TEMP. DISTRIBUTION    :: TAPE42'
        STOP
        END
```

```
C=====================================================================
C
C      TITLE:  CONV1D.FOR
C
C       TASK:  THIS PROGRAM EMPLOYS THE ENTHALPY METHOD FOR
C              SOLVING HEAT TRANSFER PROBLEMS INVOLVING PHASE
C              CHANGE. THE GAUSS-SEIDEL ITERATIVE SCHEME IS USED
C              TO SOLVE THE SYSTEM OF SIMOULTANEOUS EQUATIONS.
C              VARIOUS RESULTS (SUCH AS ENTHALPY DISTRIBUTION,
C              TEMPERATURE DISTRIBUTION, INTERFACE LOCATION...)
C              CAN BE OBTAINED (AS OUTPUTS OF THE PROGRAM) BY
C              SETTING THE APPROPRIATE SWITCHES EQUAL TO 1.
C
C   GEOMETRY:  LONG CIRCULAR CYLINDER (1-D APPROXIMATION)
C
C   BOUNDARY:  CONVECTION
C
C      INPUT:  INPUT DATA IN FOR035.DAT
C
C
C---------------------------------------------------------------------
C     UPDATED: DEC.  9, 1987
C=====================================================================
C
C                              NOMENCLATURE
C
C   I/O...INPUT/OUTPUT
C     I: INPUT
C     O: OUTPUT
C     I/O: INPUT/OUTPUT
C  VAR...TYPE OF VARIABLE: INTEGER/REAL/CHARACTER
C     I: INTEGER
C     R: REAL
C     C: CHARACTER
C  DIM...DIMENSION:0,1,2,3,...
C
C---------------------------------------------------------------------
C  VARNAME I/O VAR DIM  DESCRIPTION
C  ------- --- --- ---  -----------------------------------------
C  BI      I    R   0    BIOT MODULUS
C  DR      -    R   1    SPATIAL STEP-SIZE IN R DIRECTION
C                        USED IN TEMP. DISTRIBUTION SECTION
C  DFO     I    R   0    NON-DIMENSIONAL TIME-STEP
C  EPS     I    R   0    MAXIMUM DEVIATION OF ENTHALPY AT ANY NODE
C                        FROM PREVIOUS ITERATION
C  FI      I/O  R   3    NON-DIMENSIONAL TEMPERATURE
C  FIB     I    R   0    SPECIFIED BOUNDARY TEMPERATURE
C  FO      -    R   0    NON-DIMENSIONAL TIME (FOURIER NO.)
C  FLAG    -    I   0    INDICATOR FOR CONVERGENCE
C  I       -    I   0    INDEX IN R DIRECTION
```

```
C  N        I     I    0     MAXIMUM VALUE OF INDEX I
C  K        I     I    0     NUMBER OF TIME-STEPS
C  M        -     I    0     TIME INDEX
C  MF       -     I    0     VALUE OF TIME INDEX AT INSTANT
C                             CENTER FREEZES
C  MAXNUM   I     I    0     MAXIMUM NUMBER OF ITERATION
C  NUMB     -     I    0     ITERATION COUNTER
C  RC       O     R    1     RADIAL POSITION USED IN INTERFACE
C  RN       I     R    0     NON-DIMENSIONAL DELTA R
C                             POSITION ON MAX. RADIUS SECTION
C  STE      I     R    0     STEFAN NUMBER
C  SWO31    I     I    0     SWITCH FOR OUTPUT TO TAPE31 (ON:1,OFF:0)
C  SWO32    I     I    0     SWITCH FOR OUTPUT TO TAPE32 (ON:1,OFF:0)
C  SWO33    I     I    0     SWITCH FOR OUTPUT TO TAPE33 (ON:1,OFF:0)
C  TEMP1..  O     R    2     TEMPERATURE VARIABLE USED IN TEMPERATURE
C                             DISTRIBUTION SCTION
C  TH       I/O   R    3     NON-DIMENSIONAL ENTHALPY
C  TIMEF    -     R    0     VALUE OF FOURIER NUMBER AT THE INSTANT
C                             CENTER FREEZES
C
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
C
C=========================
      INTEGER I,M,N,K,NUMB,FLAG,MAXNUM,MF
      INTEGER SWO31,SWO32,SWO33
C-------------------------
      REAL    FO,DFO,FIB,STE,BI,RN,EPS,TIMEF
      REAL    TEMP1,TEMP2,TEMP3,TEMP4
      REAL    RC(0:1000),DR(0:11)
      REAL    TH(0:1000,0:11),FI(0:1000,0:11)
C
C.........ENTERING THE PARAMETERS OF THE PROBLEM
C
      DATA SWO31/1/
      DATA SWO32/1/
      DATA SWO33/1/
      MAXNUM=15
      K=1000
      N=10
      READ(35,*) STE,BI,EPS
      DFO=0.0025
      RN=0.1
C
C.........ENTER THE INITIAL CONDITION FOR THE PROBLEM
C
      DO 5 I1=0,N
          FI(0,I1)=0.
          TH(0,I1)=1.
5     CONTINUE
C
```

```
C.........THE START OF THE ITERATIVE PROCESS
C
        DO 10 M=1,K
C
C.........INITIALIZING FOR EACH TIME STEP
C
          DO 12 I2=0,N-1
              TH(M,I2)=TH(M-1,I2)
12        CONTINUE
          NUMB=1
100       FLAG=0
          DO 20 I=0,N-1
              IF(I .EQ. 0) THEN
                IF(TH(M,I+1) .LT. 0.) THEN
                  FI(M,I+1)=TH(M,I+1)
                ELSE
                  FI(M,I+1)=0.
                END IF
C
C.........EVALUATING THE RIGHT HAND SIDE THE ENTHALPY EQUATION
C
                RS=(RN**2./DFO)*TH(M-1,I)+4.*FI(M,I+1)
C
C.........CHECK TO SEE WHETHER THE ELEMENT IS ONE-PHASE OR TWO-PHASE
C
                IF(RS .LT. 0) THEN
                  THMO=RS/(RN**2./DFO+4.)
                ELSE
                  THMO=RS/(RN**2./DFO)
                END IF
                DIFF=ABS(TH(M,0)-THMO)
C
C.........CHECK THE CONVERGENCE
C
                IF(DIFF .GT. EPS) FLAG=1
                TH(M,0)=THMO
C
C.........PROCEDURE FOR NON-ZERO NALUES OF I
C
              ELSE
                IF(TH(M,I+1) .LT. 0.) THEN
                  FI(M,I+1)=TH(M,I+1)
                ELSE
                  FI(M,I+1)=0.
                END IF
                IF(TH(M,I-1) .LT. 0.) THEN
                  FI(M,I-1)=TH(M,I-1)
                ELSE
                  FI(M,I-1)=0.
                END IF
```

```
C
C.........EVALUATING THE RIGHT HAMD SIDE OF THE EQUATION AND DIVIDING
C                 BY THE APPROPRIATE COEFFICIENT DEPENDING ON WHETHER
C                 THE ELEMENT IS A ONE-PHASE OR A TWO-PHASE ELEMENT
C
                  RS=(RN**2./DFO)*TH(M-1,I)+(1.-1./(2.*I))*FI(M,I-1)
     *               +(1.+1./(2.*I))*FI(M,I+1)
                  IF(RS .LT. 0.) THEN
                    THMI=RS/(RN**2./DFO+2.)
                  ELSE
                    THMI=RS/(RN**2./DFO)
                  END IF
                  DIFF=ABS(TH(M,I)-THMI)
C
C.........CHECK FOR CONVERGENCE
C
                  IF(DIFF .GT. EPS) FLAG=1
                  TH(M,I)=THMI
                END IF
20          CONTINUE
C
C.........PROCEDURE FOR ELEMENTS LOCATED ON THE CONVECTIVE BOUNDARY
C
            IF(TH(M,N-1) .LT. 0.) THEN
             FI(M,N-1)=TH(M,N-1)
            ELSE
              FI(M,N-1)=0.
            END IF
            RS=(RN**2./DFO)*TH(M-1,N)+2.*FI(M,N-1)
     *         -2.*BI*STE*RN*(1.+1./(2.*N))
            IF (RS .LT. 0.) THEN
              THMN=RS/(RN**2./DFO+2.*BI*(1.+1./(2.*N))*RN+2.)
            ELSE
              THMN=RS/(RN**2./DFO)
            END IF
            DIFF=ABS(TH(M,N)-THMN)
C
C.........CHECK THE CONVERGENCE
C
            IF(DIFF .GT. 0.001) FLAG=1
            TH(M,N)=THMN
C
C.........CONTINUE THE ITERATION
C
             NUMB=NUMB+1
             IF(NUMB .GT. MAXNUM ) THEN
               WRITE(6,*)'XXXXXX ERROR XXXXX'
               WRITE(6,*)'** MAX. NUMBER OF ITERATIONS EXCEEDED **'
               STOP
             END IF
```

```
C
C
            IF(FLAG .EQ. 1) GO TO 100
        DO 101 I=0,N-1
          IF(TH(M,I) .GT. 0.)GOTO 10
101     CONTINUE
      MF=M
      TIMEF=DFO*MF
      GOTO 1011
10        CONTINUE
C
1011      CONTINUE
        WRITE(6,*) '======================================'
        WRITE(6,*) ' **** CONVERGANCE  ACHIEVED ****'
        WRITE(6,*) '--------------------------------------'
        WRITE(6,*) '  FINAL TIME-INDEX:',MF
        WRITE(6,*) '  TOTAL SULIDIFICATION TIME:',TIMEF
        WRITE(6,*) '--------------------------------------'
Cxxxxxxxxxxxxxxxxxxxxxxxxxxxx
C
        IF (SWO31 .EQ. 1) THEN
C-------------------------------------------
C:: THIS SECTION SENDS THE OUTPUT TO TAPE31 !
C-------------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
C.........PRINT THE VALUES OF THE NON-DIMENSIONAL ENTHALPY
C                   FOR VARIOUS TIMES
C
        WRITE(6,*)'WRITING OUTPUT TO TAPE31'
        WRITE(31,*)' <<<< ENTHALPY DISTRIBUTION FOR 1-D CYLINDER >>>>'
        WRITE(31,*)' <<<<   WITH CONVECTIVE BOUNDARY CONDITION   >>>>'
        WRITE(31,*)' ------------------------------------------------'
C
          DO 40 M3=1,MF
            TFO=M3*DFO
            WRITE(31,23) TFO,M3
23          FORMAT(2X,'FO=',F7.5,2X,I5)
              WRITE(31,25) (TH(M3,I3),I3=0,N)
25           FORMAT(15(1X,F6.3))
40        CONTINUE
C
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
        END IF
C
C
        IF(SWO32 .EQ. 1) THEN
C-----------------------------------
C:: THIS SECTION PROVIDES DATA FOR    !
C;; TEMP. VS. R PLOTS                 !
```

```
C
C
           IF(FLAG .EQ. 1) GO TO 100
       DO 101 I=0,N-1
         IF(TH(M,I) .GT. 0.)GOTO 10
101    CONTINUE
     MF=M
     TIMEF=DFO*MF
     GOTO 1011
10       CONTINUE
C
1011     CONTINUE
       WRITE(6,*) '========================================'
       WRITE(6,*) ' **** CONVERGANCE  ACHIEVED ****'
       WRITE(6,*) '----------------------------------------'
       WRITE(6,*) '  FINAL TIME-INDEX:',MF
       WRITE(6,*) '  TOTAL SULIDIFICATION TIME:',TIMEF
       WRITE(6,*) '----------------------------------------'
Cxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
C
       IF (SWO31 .EQ. 1) THEN
C---------------------------------------------
C:: THIS SECTION SENDS THE OUTPUT TO TAPE31 !
C---------------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
C.........PRINT THE VALUES OF THE NON-DIMENSIONAL ENTHALPY
C                   FOR VARIOUS TIMES
C
       WRITE(6,*)'WRITING OUTPUT TO TAPE31'
       WRITE(31,*)' <<<< ENTHALPY DISTRIBUTION FOR 1-D CYLINDER >>>>'
       WRITE(31,*)' <<<<   WITH CONVECTIVE BOUNDARY CONDITION   >>>>'
       WRITE(31,*)' ----------------------------------------------'
C
         DO 40 M3=1,MF
           TFO=M3*DFO
           WRITE(31,23) TFO,M3
23         FORMAT(2X,'FO=',F7.5,2X,I5)
             WRITE(31,25) (TH(M3,I3),I3=0,N)
25          FORMAT(15(1X,F6.3))
40       CONTINUE
C
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
       END IF
C
C
       IF(SWO32 .EQ. 1) THEN
C-------------------------------------
C:: THIS SECTION PROVIDES DATA FOR    !
C;; TEMP. VS. R PLOTS                 !
```

```
C-------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
C
      WRITE(6,*)'WRITING OUTPUT TO TAPE32'
      WRITE(32,*)' <<<< TEMPERATURE DISTRIBUTION FOR  CYLINDER >>>>'
      WRITE(32,*)' <<<<   WITH CONVECTIVE BOUNDARY CONDITION   >>>>'
      WRITE(32,*)' ---------------------------------------------------'
        DO 201 I=0,N
          DR(I)=(0.5+I)*RN
          DR(0)=0.
          DR(N)=1.
          TEMP1=1.+FI(411,I)
          TEMP2=1.+FI(350,I)
          TEMP3=1.+FI(300,I)
          TEMP4=1.+FI(200,I)
         WRITE(32,801) DR(I),TEMP1,TEMP2,TEMP3,TEMP4
801     FORMAT(1X,F8.5,1X,F9.6,1X,F9.6,1X,F9.6,1X,F9.6)
201     CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
C
      IF(SWO33 .EQ. 1) THEN
C-------------------------------------
C:: THIS SECTION COMPUTES COORDINATES !
C;; OF THE INTERFACE                  !
C-------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
        DO 202 M=0,MF,2
            DO 202 I=0,N
          IF(TH(M,I) .GT. 0. .AND. TH(M,I) .LT. 1.) THEN
          RC(M)=(I-.5+TH(M,I))*RN
         IF(I .EQ. N) THEN
               RC(M)=(I-0.5+TH(M,I)/2.)*RN
         END IF
         IF(I .EQ. 0) RC(M)=TH(M,I)*RN/2.
         END IF
202     CONTINUE
      WRITE(6,*)'WRITING OUTPUT TO TAPE33'
      WRITE(33,*) '<<<< RESULTS OF INTERFACE LOCATION >>>>'
      WRITE(33,*) '---------------------------------- >>>>'
      RC(0)=1.
        DO 203 M=0,MF,4
         FOUR=M*DFO
         WRITE(33,309) RC(M),FOUR
309     FORMAT(2X,F8.5,2X,F8.5)
203     CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
```

```
      END IF
C
      WRITE(6,*)                    ' --------------------------------'
      WRITE(6,*)                    ' ***** LIST OF OUTPUT FILES *****'
      WRITE(6,*)                    ' --------------------------------'
      IF(SWO31 .EQ. 1 ) WRITE(6,*) '  ENTHALPY DISTRIBUTION :: TAPE31'
      IF(SWO32 .EQ. 1 ) WRITE(6,*) '  TEMP. DISTRIBUTION    :: TAPE32'
      IF(SWO33 .EQ. 1 ) WRITE(6,*) '  INTERFACE LOCATION    :: TAPE33'
        STOP
        END
```

```
C=======================================================================
C
C      TITLE:  SUPER SHEATA.FOR
C
C       TASK:  THIS PROGRAM EMPLOYS AN ITERATIVE SCHEME
C              TO CALCULATE TRANSIANT TEMPERATURE DISTRBUTION
C              IN A 2-D CIRCULAR CYLINDER CONTAINING A PHASE CHANGE
C              MATERIAL USING THE ENTHALPY METHOD WITH SPECIFIED
C              TEMPERATURE ON THE BOUNDARY.
C              IN THIS PROGRAM THE LIQUID IS INITIALLY IN THE
C              SUPER-HEATED STATE
C
C-----------------------------------------------------------------------
C     UPDATED: DEC.  9, 1987
C=======================================================================
C
C                              NOMENCLATURE
C
C   I/O...INPUT/OUTPUT
C     I: INPUT
C     O: OUTPUT
C     I/O: INPUT/OUTPUT
C   VAR...TYPE OF VARIABLE: INTEGER/REAL/CHARACTER
C     I: INTEGER
C     R: REAL
C     C: CHARACTER
C   DIM...DIMENSION:0,1,2,3,...
C
C-----------------------------------------------------------------------
C  VARNAME I/O VAR DIM  DESCRIPTION
C  ------- --- --- ---  ----------------------------------------
C  DR      O    R   2    RADIAL POSITION USED IN COORD. OF
C                        INTERFACE COMPUTATION SECTION
C  DZ      O    R   2    VERTICAL POSITION USED IN COORD. OF
C                        INTERFACE COMPUTATION SECTION
C  DFO     -    R   0    NON-DIMENSIONAL TIME-STEP
C  FI      I/O  R   3    NON-DIMENSIONAL TEMPERATURE
C  FIB     I    R   0    SPECIFIED BOUNDARY TEMPERATURE
C  FO      -    R   0    NON-DIMENSIONAL TIME (FOURIER NO.)
C  FOA     O    R   1    TIME VARIABLE USED IN FROZ. FRACTION SECTION
C  FLAG    -    I   0    INDICATOR FOR CONVERGENCE
C  I       -    I   0    INDEX IN R DIRECTION
C  J       -    I   0    INDEX IN Z DIRECTION
C  K       I    I   0    NUMBER OF TIME-STEPS
C  KKM     O    I   1    NUMBER OF POINTS LOCATED ON THE INTERFACE
C                        FOR A GIVEN TIME-LEVEL
C  M       -    I   0    TIME INDEX
C  MAXNUM  I    I   0    MAXIMUM NUMBER OF ITERATION
C  N       I    I   0    TOTAL NUMBER OF POINTS IN R DIR.
C  NN      I    I   0    TOTAL NUMBER OF POINTS IN Z DIR.
```

```
C  NTL      -     I   0     TOTAL NUMBER OF CONTROL VOLUMES
C  NUMB     -     I   0     ITERATION COUNTER
C  RN       -     R   0     NON-DIMENSIONAL DELTA R
C                            CONTROL VOLUME
C                            POSITION ON MAX. RADIUS SECTION
C  SWO51    I     I   0     SWITCH FOR OUTPUT TO TAPE51 (ON:1,OFF:0)
C  SWO52    I     I   0     SWITCH FOR OUTPUT TO TAPE52 (ON:1,OFF:0)
C  SWO53    I     I   0     SWITCH FOR OUTPUT TO TAPE53 (ON:1,OFF:0)
C  TEMP     -     R   2     TEMPERATURE ARRAY USED IN TEMPERATURE
C                            DISTRIBUTION SCTION
C  TCSCV    -     R   0     TOTAL NUMBER OF COMPLETELY SOLID C.V.'S
C  TPSCV    -     R   0     TOTAL NUMBER OF PARTIALLY  SOLID C.V.'S
C  TH       I/O   R   3     NON-DIMENSIONAL ENTHALPY
C
C23456789012345678901234567890123456789012345678901234567890123456789012
C
C=========================
      REAL              TH(0:18,0:6,0:40),FI(0:18,0:6,0:40)
C-------------------------
      INTEGER I,J,NN,N,M,K,NUMB,FLAG,MAXNUM
      INTEGER SWO51,SWO52,SWO53
      INTEGER KKM(18)
C-------------------------
      REAL    FO,DFO,FIB,RN,PSCV,TCSCV,TPSCV
      REAL    FFCT(0:18),FOA(0:18)
      REAL    DR(18,90),DZ(18,90)
C
C.........ENTERING THE PARAMETERS OF THE PROBLEM
C
      DATA SWO51/1/
      DATA SWO52/1/
      DATA SWO53/1/
      MAXNUM=10
      K=18
      N=5
      NN=40
      DFO=0.025
      RN=0.2
      FIB=-1.0
C.....BOUNDARY CONDITION ON TOP AND BOTTOM
      DO 1 M1=1,K
        DO 2 II=0,N
          FI(M1,II,NN)=FIB
          TH(M1,II,NN)=FIB
          FI(M1,II,0)=FIB
          TH(M1,II,0)=FIB
2       CONTINUE
1     CONTINUE
C     BOUNDARY CONDITION ON SIDE
      DO 3 M2=1,K
```

```
            DO 4 JJ=1,NN-1
              FI(M2,N,JJ)=FIB
              TH(M2,N,JJ)=FIB
    4         CONTINUE
    3       CONTINUE
    C.....  INITIAL CONDITION
            DO 5 I=0,N
              DO 6 J=0,NN
                TH(0,I,J)=1.2
    6         CONTINUE
    5       CONTINUE
            DO 10 M=1,K
    C.....  INITIALIZING FOR EACH TIME STEP
              DO 12 I2=0,N-1
                DO 14 J2=1,NN-1
                  TH(M,I2,J2)=TH(M-1,I2,J2)
    14          CONTINUE
    12        CONTINUE
              NUMB=1
    100       FLAG=0
              DO 20 I=0,N-1
                DO 30 J=1,NN-1
                  IF(I .EQ. 0) THEN
                    IF(TH(M,I+1,J) .LT. 0.) THEN
                      FI(M,I+1,J)=TH(M,I+1,J)
                    ELSE IF(TH(M,I+1,J) .LT. 1.) THEN
                      FI(M,I+1,J)=0.
                    ELSE
                      FI(M,I+1,J)=TH(M,I+1,J)-1.
                    END IF
                    IF(TH(M,I,J-1) .LT. 0.) THEN
                      FI(M,I,J-1)=TH(M,I,J-1)
                    ELSE IF(TH(M,I,J-1) .LT. 1.) THEN
                      FI(M,I,J-1)=0.
                    ELSE
                      FI(M,I,J-1)=TH(M,I,J-1)-1.
                    END IF
                    IF(TH(M,I,J+1) .LT. 0.) THEN
                      FI(M,I,J+1)=TH(M,I,J+1)
                    ELSE IF(TH(M,I,J+1) .LT. 1.) THEN
                      FI(M,I,J+1)=0.
                    ELSE
                      FI(M,I,J+1)=TH(M,I,J+1)-1.
                    END IF
                    RS=(RN**2./DFO)*TH(M-1,I,J)+4.*FI(M,I+1,J)+FI(M,I,J-1)
          *         +FI(M,I,J+1)
                    IF(RS .LT. 0) THEN
                      THMOJ=RS/(RN**2./DFO+6.)
                    ELSE IF(RS .LT. 1.) THEN
                      THMOJ=RS/(RN**2./DFO)
```

```
                ELSE
                  THMOJ=(RS+6.)/(RN**2./DFO+6.)
                END IF
                DIFF=ABS(TH(M,0,J)-THMOJ)
                IF(DIFF .GT. 0.005) FLAG=1
                TH(M,0,J)=THMOJ
              ELSE
                IF(TH(M,I+1,J) .LT. 0.) THEN
                  FI(M,I+1,J)=TH(M,I+1,J)
                ELSE IF(TH(M,I+1,J) .LT. 1.) THEN
                  FI(M,I+1,J)=0.
                ELSE
                  FI(M,I+1,J)=TH(M,I+1,J)-1.
                END IF
                IF(TH(M,I-1,J) .LT. 0.) THEN
                  FI(M,I-1,J)=TH(M,I-1,J)
                ELSE IF(TH(M,I-1,J) .LT. 1.) THEN
                  FI(M,I-1,J)=0.
                ELSE
                  FI(M,I-1,J)=TH(M,I-1,J)-1.
                END IF
                IF(TH(M,I,J+1) .LT. 0.) THEN
                  FI(M,I,J+1)=TH(M,I,J+1)
                ELSE IF(TH(M,I,J+1) .LT. 1.) THEN
                  FI(M,I,J+1)=0.
                ELSE
                  FI(M,I,J+1)=TH(M,I,J+1)-1.
                END IF
                IF(TH(M,I,J-1) .LT. 0.) THEN
                  FI(M,I,J-1)=TH(M,I,J-1)
                ELSE IF(TH(M,I,J-1) .LT. 1.) THEN
                  FI(M,I,J-1)=0.
                ELSE
                  FI(M,I,J-1)=TH(M,I,J-1)-1.
                END IF
                RS=(RN**2./DFO)*TH(M-1,I,J)+(1.-1./(2.*I))*FI(M,I-1,J)
     *             +(1.+1./(2.*I))*FI(M,I+1,J)+FI(M,I,J-1)+FI(M,I,J+1)
                IF(RS .LT. 0.) THEN
                  THMIJ=RS/(RN**2./DFO+4.)
                ELSE IF(RS .LT. 1.) THEN
                  THMIJ=RS/(RN**2./DFO)
                ELSE
                  THMIJ=(RS+4.)/(RN**2./DFO+4.)
                END IF
                DIFF=ABS(TH(M,I,J)-THMIJ)
                IF(DIFF .GT. 0.005) FLAG=1
                TH(M,I,J)=THMIJ
              END IF
   30       CONTINUE
   20     CONTINUE
```

```
          NUMB=NUMB+1
          IF(NUMB .GT. MAXNUM) THEN
            PRINT*,'XXXXXX ERROR XXXXX'
            PRINT*,'** MAX. NUMBER OF ITERATIONS EXCEEDED **'
            STOP
          END IF
          IF(FLAG .EQ. 1) GO TO 100
10      CONTINUE
C
      PRINT*,'PROGRAM SHEATA CONVERGED'
C
      IF(SWO51 .EQ. 1) THEN
C--------------------------------------------
C:: THIS SECTION SENDS THE OUTPUT TO TAPE51 !
C--------------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
      PRINT*,'WRITING OUTPUT TO TAPE51'
      WRITE(51,*)' ::   TAPE51:: ----------------------------------'
      WRITE(51,*)' <<<< ENTHALPY DISTRIBUTION IN CYLINDER >>>>'
      WRITE(51,*)' ----------------------------------------------------'
        DO 40 M3=1,K
          TFO=M3*DFO
          WRITE(51,23) TFO
23        FORMAT(///2X,'FO=',F7.5)
          DO 50 J3=0,NN
            WRITE(51,25) (TH(M3,I3,J3),I3=0,N)
25         FORMAT(6(1X,F6.3))
50        CONTINUE
40      CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
C
      IF(SWO52 .EQ. 1) THEN
C--------------------------------------
C:: THIS SECTION COMPUTES COORDINATES !
C;; OF THE INTERFACE                  !
C--------------------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
        DO 141 M=1,K
          KK=0
          DO 142 J=0,NN
            DO 143 I=0,N
              IF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *           .AND. TH(M,I,J+1) .EQ. 1.) THEN
                KK=KK+1
                DZ(M,KK)=RN/2.+J*RN-TH(M,I,J)*RN
                DR(M,KK)=I*RN
              ELSEIF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *              .AND. TH(M,I,J-1) .EQ. 1.) THEN
```

```
                  KK=KK+1
                  DZ(M,KK)=RN/2.+(J-1)*RN+TH(M,I,J)*RN
                  DR(M,KK)=I*RN
                ELSEIF(TH(M,I,J) .GT. 0. .AND. TH(M,I,J) .LT. 1.
     *                 .AND. TH(M,I-1,J) .EQ. 1.) THEN
                  KK=KK+1
                  DZ(M,KK)=J*RN
                  DR(M,KK)=RN/2.+(I-1)*RN+TH(M,I,J)*RN
                END IF
143          CONTINUE
142        CONTINUE
            KKM(M)=KK
141       CONTINUE
      PRINT*,'WRITING OUTPUT TO TAPE52'
      WRITE(52,*)' ::   TAPE52:: ---------------------------------'
      WRITE(52,*)'<<<< RESULTS OF INTERFACE LOCATION >>>>'
      WRITE(52,*)' --------------------------------------------------'
         DO 151 L=1,KKM(2)
        WRITE(52,801) DR(2,L),DZ(2,L),DR(6,L),DZ(6,L),DR(9,L),
     *                DZ(9,L),DR(12,L),DZ(12,L)
801     FORMAT(1X,F8.5,1X,F8.5,1X,F8.5,1X,F8.5,1X,F8.5,1X,F8.5,
     *         1X,F8.5,1X,F8.5)
151      CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
C
      IF(SW053 .EQ. 1) THEN
C-----------------------------
C:: THIS SECTION CALCULATES   !
C:; THE FROZEN FRACTION       !
C-----------------------------
C::TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
C
        NTL=N*(NN+1)
        DO 201 M=1,K
        TCSCV=0.
        TPSCV=0.
          DO 202 I=0,N
            DO 203 J=0,NN
             IF(TH(M,I,J) .LE. 0.) THEN
              IF(I .EQ. 0) THEN
                TCSCV=TCSCV+0.5
              ELSE
                TCSCV=TCSCV+1.
              END IF
              ELSEIF(TH(M,I,J) .GT. 0. .AND.
     *               TH(M,I,J) .LT. 1.)THEN
                IF(I .EQ. 0)THEN
```

```
                  PSCV=(1.-TH(M,I,J))/2.
                ELSE
                  PSCV=1.-TH(M,I,J)
                END IF
                  TPSCV=TPSCV+PSCV
              END IF
203         CONTINUE
202       CONTINUE
            FFCT(M)=(TCSCV+TPSCV)/NTL
            FOA(M)=M*DFO
201     CONTINUE
      PRINT*,'WRITING OUTPUT TO TAPE53'
      WRITE(53,*)' ::   TAPE53:: ----------------------------------'
      WRITE(53,*)'<<<< RESULTS OF FROZEN FRACTION >>>>'
      WRITE(53,*)' ----------------------------------------------'
        DO 211 M=1,K
          WRITE(53,802) FOA(M),FFCT(M)
802     FORMAT(1X,F9.6,2X,F9.6)
211     CONTINUE
C::BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
      END IF
C
      PRINT*,                          ' ----------------------------------'
      PRINT*,                          ' ***** LIST OF OUTPUT FILES *****'
      PRINT*,                          ' ----------------------------------'
      IF(SWO51 .EQ. 1 ) PRINT*,        '  ENTHALPY DISTRIBUTION :: TAPE51'
      IF(SWO52 .EQ. 1 ) PRINT*,        '  INTERFACE LOCATION    :: TAPE52'
      IF(SWO53 .EQ. 1 ) PRINT*,        '  FROZEN FRACTION       :: TAPE53'
        STOP
        END
```

APPENDIX C

THERMAL PROPERTIES OF CHEVRON 140 WAX

103.12 12/83
Paraffin Waxes

CHEVRON REFINED WAXES

TYPICAL TEST DATA*

		Chevron Refined Wax						
		128	129	133	140	143	150	155
CPS Number		260401	260402	260403	260404	260405	260406	260407
Property	ASTM Test Method							
Melting Point	D 87							
°C		53	52	55	59	60	64	66
°F		127	126	131	138	141	148	151
°F AMP		130	129	134	141	144	151	154
Penetration	D 1321							
at 25°C (77°F)		14	17	13	10	11	10	14
at 38°C (100°F)		60	105	72	19	27	18	36
FDA Compliance	21CFR: S172.886 S178.3710	Yes	Yes	Yes	Yes	Yes	Yes	Yes
USDA Approval	–	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Color, Saybolt (Melted)	D 156	+28	+28	+28	+28	+28	+28	+28
Oil Content, Mass %	D 721	0.3	0.3	0.3	0.3	0.3	0.3	0.3
Blocking Point	D 1465							
°C		38	35	39	43	43	50	43
°F		100	95	102	110	110	122	110
Viscosity, Kinematic, cSt	D 445							
at 100°C (212°F)		3.0	3.9	3.5	3.8	4.3	4.7	6.7
Density, g/ml								
at 25°C (77°F)	Chevron	0.900	0.905	0.905	0.915	0.915	0.920	0.920
at 80°C (176°F)	D 4052	0.760	0.775	0.765	0.770	0.775	0.775	0.785
Flash Point, COC	D 92							
°C		235	245	195	240	245	255	270
°F		455	470	380	465	475	490	520
Autoignition Temperature	D 2155							
°C		330	340	335	340	345	365	370
°F		630	645	635	645	655	690	700

	Solid	Transition	Liquid
Coefficient of Thermal Expansion, $(°C)^{-1}$	–		6.3×10^{-4}
Heat of Fusion			
Cal./g		55	
Btu/Lb		100	
Specific Heat			
Cal./g-°C	0.5		0.5
Btu/Lb-°F	0.5		0.5
Thermal Conductivity			
Cal./Sec.-m-°C	3×10^{-4}		6×10^{-4}
Btu/Hr-Ft-°F	0.7		0.15

*These are average values. They may be changed without notice. Minor variations which do not affect performance are to be expected.

Chevron

APPENDIX D

COMPUTER CODE FOR TAKING THERMOCOUPLE READINGS

```
!THERMOCOUPLE READING PROGRAM
!
!
OPTION BASE 1
DIM V(6,5),Tav(6,100),Tim(100)
CLEAR 709
OUTPUT 709:"AF0AL5VR1SO1"
OUTPUT 709;"AS"
Tf=TIMEDATE
  PRINT " ","TIME(S)","T1(C)","T2(C)","T3(C)","T4(C)","T5(C)"
  Aa=30.
FOR K=1 TO 100
FOR I=1 TO 5
  ENTER 709;Vint
FOR B=1 TO 6
    ENTER 709;V(B,I)
    OUTPUT 709;"AS"
    V(B,I)=V(B,I)-Vint      ! ICE BATH CHANNEL 0
NEXT B
NEXT I
Tim(K)=TIMEDATE-Tf
  !
  !CALCULATE AVG. VOLTAGE FOR EACH T.C.
  !
  !
  FOR H=1 TO 6
    Vavg(H)=0.
    FOR I=1 TO 5
      Vavg(H)=Vavg(H)+V(H,I)
    NEXT I
    Vavg(H)=Vavg(H)/5
  NEXT H
  PRINT " ","-------------------------------------------------------------
  PRINT " ",DROUND(Tim(K),4);
  !
  ! LOOP TO CONVERT THE FROM EMF TO TEMPERATURE
  FOR H=2 TO 6
    V2=Vavg(H)*1000.
  !
  T1=-.001588515*V2^7+.023396843*V2^6-.135293163*V2^5+.388477456*V2^4
  T2=-.536749212*V2^3-.36600667*V2^2+25.889347897*V2
    Tav(H,K)=T1+T2
  PRINT " ",DROUND(Tav(H,K),4);
 !PRINT "     ",DROUND(K,4),DROUND(Tim(K),4),DROUND(Tav(H,K),4);
  NEXT H
  PRINT "  "
 !PRINT DROUND(K,2),DROUND(Tim(K),2),DROUND(Tav(H,K),2)
  WAIT Aa
  NEXT K
    !
    !Store data in 'TEMP'
    !
    !
    CREATE BDAT "TEMP",420,8
    ASSIGN @Path1 TO "TEMP"
    OUTPUT @Path1;Tim(*),Tav(*)
  STOP
  END
```

APPENDIX E

INPUT DATA FILE FOR CANDL3.FOR

```
                    <<< INPUT FILE FOR PROGRAM CANDLV3.FOR >>>
                    ******************************************

 ============================ << LOOP CONTROLLERS >> ==================
IMAX   = [34]
JMAX   = [35]
K      = [89]
MAXNUM= [50]
DFO    = [  0.010000]
FIB    = [ -0.336363]
EPS    = [  0.000100]
 ========================== << SWITCHES FOR OUTPUT FILES >> =========
SWO11 = [1]
SWO12 = [1]
SWO13 = [1]
SWO14 = [1]
SWO15 = [1]
SWO16 = [1]
 ================================
  << PHYSICAL PROPERTIES >>
 ================================
TSAT   = [ 59.000000]
LAND   = [ 55.000000]
CS     = [  0.500000]
 ================================
  << GEOMETRICAL PARAMETERS >>
 ================================
RCY    = [  0.200000]
R      = [  0.750000]
-----------
DR     ::
-----------
0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400
0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400
0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400 0.0400
0.0400 0.0400 0.0400 0.0400 0.0400 0.0000 0.0000 0.0000 0.0000 0.0000
DZ     ::
-----------
0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000
0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000
0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.1000
0.1000 0.1000 0.1000 0.1000 0.1000 0.1000 0.0000 0.0000 0.0000 0.0000
```

APPENDIX F

SAMPLE OUTPUT DATA FILE :

TAPE11 :: ENTHALPY DISTRIBUTION FOR TEAR DROP GEOMETRY

<<<< ENTHALPY DISTRIBUTION FOR >>>>
<<<< TEAR-DROP GEOMETRY >>>>

FO=0.04000 M= 4

-0.336-0.336-0.336-0.336-0.336-0.336
-0.094-0.126-0.178-0.235-0.289-0.336
0.947 0.737-0.036-0.163-0.258-0.336
1.000 0.943-0.011-0.145-0.250-0.336
1.000 0.962-0.007-0.142-0.248-0.336
1.000 0.965-0.007-0.141-0.248-0.336
1.000 0.965-0.007-0.141-0.248-0.336
1.000 0.965-0.007-0.141-0.248-0.336
1.000 0.965-0.007-0.141-0.248-0.336
1.000 0.965-0.007-0.141-0.248-0.336
1.000 0.965-0.007-0.141-0.248-0.336
1.000 0.965-0.007-0.141-0.248-0.336
1.000 0.965-0.007-0.141-0.248-0.336
1.000 0.966-0.007-0.141-0.248-0.336
1.000 0.966-0.006-0.141-0.248-0.336
1.000 0.970-0.006-0.140-0.247-0.336
1.000 0.987-0.002-0.136-0.244-0.336
1.000 1.000 0.156-0.116-0.221-0.336
1.000 1.000 1.000 0.892 0.213-0.093-0.168-0.221-0.268-0.318-0.336
1.000 1.000 1.000 1.000 1.000 0.939 0.320 0.590 0.211-0.059-0.140-0.212-0.284-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.967 0.868 0.420-0.079-0.183-0.281-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.956 0.456-0.089-0.183-0.281-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.786-0.043-0.157-0.266-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.685-0.080-0.186-0.298-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.903-0.130-0.258-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.962-0.130-0.258-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.985-0.080-0.186-0.298-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.786-0.043-0.157-0.266-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.956 0.456-0.089-0.183-0.281-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.992 0.879 0.425-0.078-0.163-0.251-0.336
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.892-0.014-0.122-0.205-0.292-0.336
0.243 1.000 1.000 1.000 1.000 1.000 1.000 0.617-0.053-0.158-0.336
-0.336

FO=0.14000 M= 14

-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.335-0.335-0.335-0.336-0.336-0.336
-0.335-0.335-0.335-0.335-0.336-0.336
-0.334-0.335-0.335-0.335-0.336-0.336
-0.334-0.334-0.335-0.335-0.336-0.336
-0.334-0.334-0.335-0.335-0.336-0.336
-0.3 -0.334-0.335-0.335-0.336-0.336
-0.334-0.334-0.335-0.335-0.336-0.336
-0.334-0.334-0.334-0.335-0.336-0.336

-0.334-0.334-0.334-0.335-0.336-0.336
-0.334-0.334-0.334-0.335-0.336-0.336
-0.334-0.334-0.334-0.335-0.336-0.336
-0.334-0.334-0.334-0.335-0.336-0.336
-0.334-0.334-0.334-0.335-0.336-0.336
-0.333-0.333-0.334-0.334-0.335-0.336
-0.331-0.331-0.332-0.333-0.335-0.336
-0.325-0.325-0.327-0.330-0.333-0.336
-0.307-0.309-0.313-0.320-0.328-0.336
-0.262-0.264-0.270-0.287-0.306-0.336
-0.158-0.161-0.170-0.186-0.209-0.234-0.259-0.284-0.308-0.329-0.336
 0.733 0.699 0.509-0.020-0.072-0.106-0.134-0.162-0.193-0.223-0.254-0.285-0.315-0.336
 1.000 1.000 1.000 0.988 0.916 0.799 0.607-0.011-0.066-0.108-0.148-0.188-0.227-0.265-0.302-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.994 0.916 0.617-0.043-0.090-0.138-0.186-0.231-0.274-0.314-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.976 0.458-0.060-0.115-0.168-0.218-0.265-0.309-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.545-0.060-0.119-0.174-0.227-0.276-0.321-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.995-0.004-0.079-0.145-0.204-0.257-0.305-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.986-0.004-0.079-0.145-0.204-0.257-0.305-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.547-0.059-0.118-0.174-0.227-0.276-0.321-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.983 0.473-0.059-0.114-0.167-0.217-0.265-0.309-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.984 0.744-0.030-0.082-0.134-0.183-0.229-0.273-0.314-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.824-0.028-0.077-0.125-0.173-0.218-0.261-0.300-0.336
 0.939 1.000 1.000 1.000 1.000 1.000 0.515-0.046-0.095-0.154-0.217-0.267-0.309-0.336
-0.912 0.811 1.000 1.000 1.000 1.000 0.232-0.033-0.073-0.165-0.336
-0.336

 FQ=0.24000 M= 24
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.336-0.336-0.336-0.336-0.336-0.336
-0.335-0.335-0.335-0.335-0.336-0.336
-0.331-0.332-0.332-0.334-0.335-0.336
-0.322-0.322-0.324-0.328-0.332-0.336
-0.296-0.297-0.302-0.309-0.321-0.336
-0.236-0.237-0.242-0.251-0.262-0.275-0.290-0.304-0.318-0.332-0.336
-0.139-0.141-0.148-0.161-0.176-0.192-0.210-0.228-0.247-0.266-0.285-0.304-0.322-0.336
 0.545 0.520 0.267-0.040-0.071-0.094-0.116-0.141-0.166-0.192-0.218-0.243-0.268-0.292-0.315-0.336
 1.000 1.000 1.000 0.977 0.906 0.794 0.397-0.043-0.079-0.113-0.146-0.179-0.211-0.242-0.270-0.297-0.322-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.964 0.722-0.034-0.075-0.117-0.157-0.194-0.229-0.262-0.292-0.319-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.981 0.081-0.061-0.112-0.157-0.197-0.234-0.268-0.299-0.327-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.760-0.046-0.093-0.138-0.180-0.219-0.255-0.288-0.317-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.765-0.046-0.093-0.138-0.180-0.219-0.255-0.287-0.317-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.995 0.125-0.060-0.110-0.155-0.196-0.233-0.267-0.298-0.327-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.894-0.010-0.060-0.107-0.150-0.190-0.226-0.260-0.291-0.319-0.336
 1.000 1.000 1.000 1.000 1.000 1.000 0.985 0.368-0.046-0.085-0.125-0.164-0.200-0.234-0.266-0.295-0.322-0.336
 1.000 1.000 1.000 1.000 1.000 0.892-0.027-0.062-0.101-0.142-0.182-0.220-0.254-0.284-0.312-0.336
 0.869 1.000 1.000 1.000 0.981-0.005-0.045-0.085-0.131-0.184-0.240-0.293-0.316-0.336
-0.012 0.532 1.000 1.000 1.000 0.319-0.024-0.047-0.081-0.167-0.336
-0.336

VITA:

Bahram Zandi was born in Teheran Iran, on July 17 1957. Shortly after his birth, he moved along with his family to West Germany, where his father was pursuing a Doctorate degree in Geology. Mr. Zandi attended Hadaf high school in Teheran. In September of 1977, he was accepted in The National University in Teheran with a major in Mathematics. In January of 1979, Mr. Zandi entered The University of Tennessee and received a Bachelor of science in Mechanical Engineering with high honors in summer of 1983. In September of 1983, Mr. Zandi began his graduate studies in The University of Tennessee. He is currently a PhD candidate in Mechanical Engineering and is expected to graduate in summer of 1990. Mr. Zandi's current research is in the area of natural convection cooling of electronic equipment.
Mr. Zandi is a member of Tau Beta Pi and Phi Kappa Phi national honor societies.