A numerical method to compute solidification and melting processes

S. Jana a, S. Ray b, F. Durst a,*

a Lehrstuhl für Strömungsmechanik, Universität Erlangen-Nürnberg, Cauerstr. 4, D-91058 Erlangen, Germany
b Department of Mechanical Engineering, Jadavpur University, Kolkata 700 032, India

Received 1 July 2004; received in revised form 1 June 2005; accepted 12 August 2005
Available online 11 November 2005

Abstract

This paper deals with thermodynamically consistent numerical predictions of solidification and melting processes of pure materials using moving grids. Till date, enthalpy-porosity-based formulations of numerical codes have been generally the popular choice, although because of an artificial numerical smearing of the interface, it is virtually impossible to reproduce a sharp melting/solidification front that is supposed to exist for phase changes of pure substances. Numerical techniques based on moving grid methods have been relatively less used as they rely on complex and time-consuming adaptive grid generations. Using the moving grid approach, the authors present a method to solve solidification and melting problems. A simple linear interpolation is used to slide grid nodes along the interface to handle the otherwise obtained grid skewness near the interface. The numerical approach employed is validated with standard test cases available in the literature. The capability of capturing very complex flow field structures and the superiority of the present approach over enthalpy-porosity-based formulations is discussed. The authors also demonstrate the ability of the set-up computer code to solve complex thermo-fluid processes such as occur during crystal growth in Czochralski reactors.

© 2005 Elsevier Inc. All rights reserved.

1. Introduction, state of knowledge and aim of work

Solidification and melting are important parts of manufacturing processes such as crystal growth, casting, welding, surface alloying, dip forming, spray coating and production of printed circuit electronics. In all these processes, phase changes of material are caused by the heat transfer to and from both of the phases on either side of the interface. This yields melting if the net heat is added to the solid part of the interface and solidification when the net heat is subtracted. The observed additional heat, which is involved in the conversion of one phase to another, is the latent heat and the entire heat transport problem is usually referred to as the Stefan problem [1]. Its solution states that the movement of the interface requires the addition of the latent heat in the case of melting and subtraction in the case of solidification. Furthermore, since a “sharp” interface is obtained during the phase change of a pure substance, it is also important that when these problems are...
Nomenclature

\( A \) coefficients
\( e, w, n, s \) east, west, north, south cell faces
\( E, W, N, S \) east, west, north and south nodes
\( Gr \) Grashof number
\( g \) acceleration due to gravity
\( k \) thermal conductivity
\( K \) kelvin
\( P \) pressure
\( Pr \) Prandtl number
\( b_{rh} \) right-hand side of the solution matrix
\( Ra \) Rayleigh number
\( S \) surface area
\( c \) specific heat
\( Ste \) Stefan number
\( t \) time
\( T \) temperature
\( U \) velocity
\( V \) volume

Greek symbols

\( \beta \) coefficient of thermal expansion
\( \rho \) density
\( \epsilon \) emissivity
\( v \) kinematic viscosity
\( \phi \) transport variable
\( \mu \) dynamic viscosity
\( \zeta, \eta \) coordinate lines

Subscripts

buo buoyancy
e in energy equations
f fluid
g grid
I interface
\( i, j \) direction index
l liquid
nb neighbouring nodes
\( P \) central node
p constant pressure
ref reference
s solid
\( u \) in momentum equations

Superscripts

\( n \) new
\( o \) old
solved numerically, the phase change processes are treated in a thermodynamically consistent manner in order to satisfy this criterion. Any vagueness in the implementation of the appropriate conditions at the interface, i.e., the occurrence of any kind of numerical smoothing, will yield errors in the prediction of the interface location. This in turn will not permit the correct capture of dopant and impurity distribution, alloy composition and other thermophysical properties in the melt and the solid part considered in the predictions.

In the literature, basically two different approaches are used for numerical simulation of the phase change processes (i) fixed-domain formulation and (ii) variable-domain formulation. A special class of the fixed-domain approach is based on “enthalpy-porosity formulation”. This type of formulation was first proposed by Voller et al. [2], which has subsequently been modified by several groups (see, for example, [3–7]). In this approach, the total enthalpy, rather than the temperature, is considered as the primary dependent variable in the energy equation and, hence, a set of auxiliary relations is required between them. When the substance is impure, the computational modelling is fairly simple and straightforward to implement because, in this case, the latent heat transfer takes place over a range of temperatures that demarcate the “mushy” region. The physical and transport properties of this zone must be either known, or modelled. This allows one to define a representative value for the specific heat during phase change process.

For a pure substance the early versions of enthalpy formulations [2] assumed that the phase change occurs over a narrow but predefined range of temperature (which can never be zero during computation) and thereby a pseudo-specific heat is defined for the phase change phenomenon. As a result, a sharp interface could not be numerically retrieved. However, subsequent improvements in the enthalpy-porosity formulation resulted in an iterative updating of the liquid fraction, in consistency with the thermodynamics of the associated phase change process [8,9]. Irrespective of improvements, it has always been a difficult proposition to numerically simulate a sharp interface using enthalpy-porosity formulation primarily because of the limitations in using limitingly thin grid spacings near interfacial region.

In some other attempts, moving mesh finite element methods have been developed to solve phase change problems with enthalpy-porosity formulation [10–12]. The essential feature of these methods lies in the fact that here the elements were allowed to move such that they may be clustered in the region, where the flow (solution) variables are expected to vary rapidly. In this manner, these methods eliminate the inherent problem of fixed grid formulation, where one requires to specify appropriate grid resolution throughout the entire solution domain. However, since these formulations were based on enthalpy-porosity technique, they still possess the inherent drawback of such techniques in a sense that the temperature range over which the phase change is assumed to take place should be set to zero (or, negligibly small) such that the formulation boils down to the classical Stefan problem.

Other popular methods for tracking interface and moving boundaries in an Eulerian frame are the “Volume of Fluid” (VOF) method and its refinements [13–19], the “Level Set” method [20–25] and the “Phase Field” method [26–29]. All of these methods perform very well with problems involving free surfaces. The main difference with these methods with the enthalpy formulation is that they all consider the energy equation with only temperature as the dependent variable, hence no direct relationship between enthalpy and temperature is required. At the interface, the mass and energy jump conditions are implemented in a careful manner in “VOF” and “Level Set” methods and the entire field, involving both the phases, is solved simultaneously. In the “Phase Field” method, however, the interface is not explicitly tracked and a phase field equation is obtained from the thermodynamic considerations. It is also worth mentioning that the latter method is more general and various effects, for example, Gibbs–Thomson effect, could be included in the formulation with minor modifications.

In the VOF [13–19] or in the Level Set formulation [20–25], either a colour function or a Level Set function is defined to identify the interface. This function is advected by solving an appropriate transport equation from the knowledge of the velocity field at a new time step. These methods, however, do not automatically yield the exact location of the interface. Youngs [14] was first to propose a method for reconstructing the continuous shape of the interface. This method has subsequently been modified and enhanced by various researchers [16–18]. These methods now obtain the shape of the interface as part of the solution procedure and the “jump” conditions at the interfaces are implemented in these methods for phase change problems. However, one of the drawbacks of these methods is that they assume averaged properties at the “mixed” control volumes containing both the phases, which may be considered to be a numerical trick rather than a physically
consistent model. Also, relatively simple and easily implementable algorithms to obtain the exact shape of the interface are available only when the calculations are performed on orthogonal coordinates and, hence, when the problem geometry is simple. So far, to the best of the authors' knowledge, very little effort has been put into developing methods that obtain the exact shape of the interface on generalised, three-dimensional, structured, curvilinear grids (see [30,31]). The knowledge of the exact location of the interface is, however, essential in order to implement the crystal pulling model for the simulation of the actual crystal growth process.

In some other classes of fixed-grid methods for melting and solidification, generally termed Eulerian–Lagrangian methods, the interface is tracked explicitly as a set of curves, while the actual computations for the momentum, heat and mass transfer processes are performed on fixed grids. The boundary condition at the interface is taken care of during the discretization procedure. Although these methods, in principle, are extensible to three-dimensional flows, they are complicated even in two-dimensional orthogonal coordinates (see, for example, [32]) and hence, the use of generalised curvilinear coordinates would further complicate their implementation.

Compared with the fixed-domain methods, less effort has been put into computing the phase change problems with variable domains. In the variable-domain formulation, two different sets of governing equations are solved in each of the phases. The Stefan boundary condition is used at the solid–liquid interface to calculate the interface velocity and the interface is then moved depending on this velocity after each time step. Once the interface has been moved, the grids are recreated in both the solid and the liquid phases. Hence these methods involve computations on “moving grids”. Numerical methods based on this approach usually result in sharp interfaces, and permit, in principle, all the properties around the interface to be computed with high accuracy. These methods are, therefore, often recommended in the literature for pure substance computations.

A general formulation for solving the conservation equations governing the fluid flow on moving grids has been presented by Demirdzic and Perić [33,34]. They have shown that in addition to the conservation equations for physical quantities, such as mass, momentum and energy, an additional space conservation law has to be satisfied in order to avoid the inclusion of an artificial mass source (or sink) in the continuity equation. They also pointed out that when the numerical grid is moved in only one direction and the grid velocities are calculated based on the movement of the centre of the cell faces during each time step, no apparent problem is observed with the associated space conservation law.

The earliest computations known to the authors of melting and solidification using moving coordinates employed finite differences to discretize the governing partial differential equations [35–37]. Hsu et al. [37], for example, derived an energy equation for a moving control volume and then applied an appropriate coordinate transformation to the various terms that take into account the movement of the interface. Viswanath and Jaluria [38] presented a boundary immobilisation technique of the phase front, based on one-dimensional Landau transformation. Since these methods considered the grid movement in only one direction, the requirement of the space conservation law did not arise.

So far as the melting and solidification problems with multi-dimensional grid movements are concerned, comprehensive reviews are available [39,40] in the literature. Kopetsch [41] solved the two-dimensional Czochralski crystal growth problem on deforming grids. In his formulation, he analytically treated the source terms generated from the grid movement. In another attempt, Zhang and coworkers [42,43] used adaptive grids and allowed grid movement during the crystal growth process. They converted the entire set of governing equations for two-dimensional axisymmetric problems in curvilinear coordinates and added the source terms due to grid velocities in the discretized equations. However, they did not show how the grid velocities are to be computed in their formulation, nor is it clearly mentioned whether their grid velocities would satisfy the space conservation law as pointed out by Demirdzic and Perić [33,34]. More recently, Shyy and coworkers [40,44] presented an algorithm to solve the phase change problems using moving grids. However, in their method, after the grid movement, they first calculated the grid velocities and applied the space conservation law to compute the Jacobian of the coordinate transformation (essentially the cell volume). It may be recognised that when the computational cell is moved, it automatically generates a new cell volume, which is uniquely defined and can be computed. Hence the space conservation law should only be applied to compute the correct grid (cell surfaces) velocities. To that extent, the implementation of Shyy and coworkers [40,44] may be recognised as an introduction of numerical mass source (or sink) in the algorithm, which satisfies the “pseudo” space conservation law.
The resulting algorithm, although it does not cause any numerical problem, also does not satisfy the correct (the physical) space conservation law.

From the brief literature review presented here, it is obvious that a thermodynamically consistent phase change model can always be implemented if computations of melting and solidification for pure substances are carried out on moving grids. However, a thermodynamically consistent formulation also requires the correct implementation of a space conservation law in order to avoid the inclusion of artificial mass sources (or sinks) due to grid velocities in the numerical model, which is otherwise still missing in the literature. Understanding of the physical processes involved in melting or solidification of pure metals is of immense importance in many engineering applications, such as, Czochralski crystal growth process, growth of liquid gallium under micro-gravity condition using floating zone technique, etc. Therefore, in this paper, a numerical methodology is developed based on the curvilinear, boundary-fitted, finite-volume method for solidification and melting problems for pure substances based on moving grids. The grid movements are taken care of in the discretization scheme, by satisfying the space conservation law. The grid points, lying on the interface, are also allowed to slide in order to control the grid skewness and to handle sufficiently large deformations arising out of non-uniform phase change in presence of convection.

The paper is organised as follows. In Section 2, the authors briefly summarise the governing equations that need to be solved for numerical computation of melting and solidification processes. Consideration is given to the discretization of the resultant partial differential equations of the integral form of the governing equations. The implementation of the boundary conditions is discussed and the solution algorithm employed is summarised. It is stressed that artificial mass flow terms result if the equations are not treated properly. In Section 3, computations are summarised, comprising a solidification problem, a melting problem and a problem that relates to the interface tracking of the crystal growth process in a Czochralski reactor. Conclusions, final remarks and an outlook are presented in Section 4.

2. Mathematical formulations

2.1. Governing equations

The fluid flow and related transport phenomena dealing with solidification and melting processes in moving coordinates are represented by a coupled set of non-linear partial differential equations. Since the present formulation deals with solutions on moving grids, where the size and shape of the control volumes (CV) are allowed to change with time, it is essential to represent the conservation laws for space, mass, momentum and scalar (energy) transport in their respective integral forms. The integral space conservation law for an arbitrary spatial region of volume \( V \), bounded by a closed surface \( S \), may be written as

\[
\frac{\partial}{\partial t} \int_V \mathbf{d}V - \int_S \mathbf{U}_g \cdot \mathbf{d}S_i = 0, \tag{1}
\]

where \( \mathbf{U}_g \) is the velocity of control volume surface (CS) or the grid velocity and the subscript \( i \) stands for coordinate direction. The corresponding mass conservation equation reads

\[
\frac{\partial}{\partial t} \int_V \rho_i \mathbf{d}V + \int_S \rho_i (\mathbf{U}_i - \mathbf{U}_g) \cdot \mathbf{d}S_i = 0, \tag{2}
\]

where \( \rho_i \) is the density of the fluid under consideration. For the solid, since the velocities are zero, the continuity equation reduces to the space conservation law as given in Eq. (1). The \( j \)th momentum conservation equation, on the other hand, is given as

\[
\frac{\partial}{\partial t} \int_V \rho_i U_j \mathbf{d}V + \int_S \rho_i (\mathbf{U}_i - \mathbf{U}_g) U_j \mathbf{d}S_i = \int_S \mu_i \left( \frac{\partial U_j}{\partial x_i} + \frac{\partial U_i}{\partial x_j} \right) \mathbf{d}S_i - \int_V \frac{\partial P}{\partial x_j} \mathbf{d}V + \int S_P \mathbf{d}V, \tag{3}
\]

where \( \mu_i \) is the dynamic viscosity of the fluid and \( S_P \) includes the source terms resulting in the conservation equations due to buoyancy and other effects. The momentum equation for the solid also reduces to the space conservation law as the velocities are set to zero. In addition to these, for the control volumes adjacent to the solid–liquid interface, surface tension effects may also be included in principle (see, for example, [46]) which
would result in additional source terms in the momentum equations. However, in the present study the effect of surface tension was not considered since the major objective of this paper is to demonstrate the correct modelling of phase change phenomena by using moving meshes and by satisfying the required space conservation law. The energy conservation equation is given as

\[
\frac{d}{dt} \int_V \rho c_p T dV + \int_S \rho c_p (U - U_g) T dS_i = \int_S k \frac{\partial T}{\partial x_i} dS_i + \int_V S_e dV,
\]

(4)

where \( S_e \) is the source term, which may arise due to volumetric energy generation, viscous dissipation, pressure work, etc. For the solid, a second heat conduction equation is solved

\[
\frac{d}{dt} \int_V \rho c_p s T dV - \int_S \rho s U_{g,s,c} c_p T dS_i = \int_S k_s \frac{\partial T}{\partial x_i} dS_i + \int_V S_s dV.
\]

(5)

In the above equations, \( T \) is the temperature, \( k \) is the thermal conductivity and \( c_p \) is the specific heat. The boundary conditions are applied in a problem-specific manner. In general, no-slip conditions are applied on the solid walls and the Stefan condition is applied at the interface in order to obtain the velocity of the interface, as will be explained below.

2.2. Finite-volume formulation and solution procedure

The governing equations were discretized based on a fully conservative finite-volume method using collocated, non-orthogonal, boundary-fitted grids. The domain under consideration was divided into a number of contiguous and non-overlapping control volumes of volume \( V \) bounded by cell faces \( S_i \) as shown in Fig. 1. The cell-centred approach (Fig. 1) was used, where the computational nodes were placed at the centre of each CV. The boundary nodes and the nodes on the solid–liquid interface, needed for implementation of boundary conditions and the Stefan condition, respectively, were placed at the centre of boundary CV faces (see Fig. 2). The details of discretization procedure are available in the book by Ferziger and Perić [47]; however, the overall formulation is briefly presented here for the sake of completeness.

A fully implicit temporal differentiation was used and the time step \( \delta t \) was first chosen. As has been shown [33,34], the space conservation equation, given in Eq. (1), may be solved explicitly and the grid velocities, \( U_{g,i} \), at each of the control volume faces may be obtained. In other words, one may use this equation to calculate volume fluxes through the CV faces which result from the motion of the respective boundaries. Once the new positions of control volumes (see Fig. 3) have been obtained, these volume fluxes and hence the boundary

![Fig. 1. A typical 2D generalised control volume.](image-url)
velocities that satisfy the space conservation law, can be easily computed. Substituting these boundary velocities, one may rewrite the space conservation equation as

$$\frac{d}{dt} \int_{V} dV - \sum_{i} \int_{S_{i}} U_{g,i} dS_{i} = 0, \quad j = e, w, n, s. \quad (6)$$

In the discretized form, the space conservation law may also be written as

$$\frac{V_{n}^{o} - V_{e}^{o}}{\delta t} = \sum_{j} \frac{\delta V_{j}}{\delta t} = \sum_{j} (U_{g,j} S_{i})_{j}, \quad (7)$$

Fig. 2. Numerical treatment of control volumes adjacent to interface and its movement.

Fig. 3. Movement of typical 2D generalised control volume using cell centred approach (the shaded regions shows swept volumes due to grid movement).
where $\delta V_j$ represents the volume swept by the $j$th face of the CV during time $\delta t$, as indicated by the shaded area in Fig. 3. From the Stefan condition for energy balance at the solid–liquid interface, the coordinates of the CV vertices, lying on the interface, were obtained at a new time level. Numerical grids in both the solid and liquid domains were then recreated. In this manner, the new and the old locations for all the CV vertices were obtained. From this knowledge, the volume swept by the ‘e’ cell face, for example, during time $\Delta t$ was obtained as

$$\frac{(\delta V_e)_{\text{G}}}{\Delta t} = \frac{1}{\Delta t} (\delta r_e \times \delta d_e),$$

(8)

where $\delta r_e$ and $\delta d_e$ are vectors, as shown in Fig. 3.

In a sequential solution method, the net mass fluxes were considered to be known for all the conservation equations other than the continuity equation and, hence, the mass fluxes in these equations were treated as if they were on a stationary grid. Therefore, the net mass flux, $\dot{m}_e$, through a cell face ‘e’, for example, was calculated as

$$\dot{m}_e = \int_{S_e} (U_i - U_{g,i}) dS_i.$$  

(9)

However, in this approach, the mass conservation equation requires special attention and the unsteady term has to be treated in a way which is consistent with the space conservation law. For a fluid of constant density, the mass conservation equation, Eq. (2), may be written as

$$\frac{d}{dt} \int_V \rho \, dV - \int_S U_g \, dS_i + \int_S U_i \, dS_i = 0.$$  

(10)

The first two terms in the above equation represent the space conservation law, hence they may be dropped by virtue of Eq. (1). This leads to the following continuity constraint:

$$\int_S U_i \, dS_i = 0.$$  

(11)

The above discretization method ensured that the unsteady term and the mass fluxes due to grid velocities satisfied the space conservation equation and a strict mass conservation law was obtained.

### 2.2.1. Modifications for axisymmetric problems

In the case of axisymmetric geometries, which are essentially two-dimensional in a cylindrical coordinate system, all derivatives with respect to the circumferential direction are zero and all the variables are functions of only axial and radial coordinates. Details of the mass, momentum and energy conservation equations in cylindrical coordinates may be found in Bird et al. [48] and in their discretized forms are given in detail in Ferziger and Perić [47], so they are not repeated here. However, the swept volumes for axisymmetric cases with any arbitrary number of faces were calculated based on the formula for calculating the volumes of each cells. If $N_v$ is the number of nodes defining the vertices in which the swept volume lies, then the swept volume is given by

$$\delta V = \frac{1}{6} \sum_{i=1}^{N_v} (z_{i-1} - z_i)(r_{i-1}^2 + r_i^2 + r_{i-1} r_i),$$

(12)

where $z$ and $r$ are the axial and radial distances.

### 2.2.2. Discretization

All the conservation equations have the same general form, represented by the energy equation for fluid as shown in Eq. (4). By taking into account the shape of control volumes, the representative conservation equation to be discretized may be written as

$$\frac{d}{dt} \int_V \rho \phi \, dV + \sum_i \int_{S_i} \left[ \rho (U_i - U_{g,i}) \phi - \Gamma \frac{\partial \phi}{\partial x_i} \right] dS_i = \int_{S} s_{\phi} dV,$$  

(13)
which has four distinctive parts: rate of change, convection, diffusion and source (the mass conservation equation, however, does not have a diffusion term). The rate-of-change and source terms were integrated over the cell volume, whereas the convection and diffusion terms formed the sum of fluxes through the control volume faces. The evaluation of each of the terms has been described in detail [34,47]. However, they are presented here for the sake of completeness.

2.2.3. Approximation of time integral

Solidification and melting are generally transient phenomena, where the explicit schemes are too restrictive owing to stability limitations. Hence implicit schemes are often preferred and the simplest choice is the first-order Euler scheme. In this method, the surface and volume integrals are calculated implicitly at the new time level and, hence, the solution from the previous time step appears only in the unsteady term. The value of the dependent variable, \( \phi \), was defined at the centre of the CV (node P in Fig. 1). Therefore, averaging over the entire CV, the time integral may be expressed as

\[
\frac{d}{dt} \int_V \rho \phi dV \approx \left( \frac{\rho \phi V}{\rho} \right)_p^{n+1} - \left( \frac{\rho \phi V}{\rho} \right)_p^n,
\]

where \( n \) and \( n + 1 \) represent two subsequent (old and the new, respectively) time levels.

2.2.4. Approximation of surface and volume integrals

The surface and volume integrals were obtained using the midpoint approximation rule, which can be shown to be second-order accurate. However, in the collocated variable arrangement, the value of the integrands in Eq. (13), are not known at the cell-face centre. Hence the interpolation and numerical differentiation are required to express the cell-face values of the variables and their derivatives in terms of their nodal values. Cell-face values of the variables were generally approximated using linear interpolation, which is a second-order interpolation. However, if the centre of the cell face does not lie on the line connecting two adjacent cell centres, the integral approximation will not be second-order accurate. The second-order accuracy was restored by adding a deferred correction term, obtained by interpolating the cell centre gradients (see [47] for details).

The convection flux of any transported quantity, \( \phi \), may be approximated over the CV surface as

\[
\int_{S_x} \rho \phi (U_i - U_{e,i}) dS_i \approx \sum_i \left( \rho U_{e,i} n_{e,i} s_{e} \right)^{n+1} - \rho \left( \frac{\partial V}{\partial t} \right)_{e}^{n+1} \phi_{e}^{n+1},
\]

where \( \phi_{e} \) represents the mean value of the variable \( \phi \) at the cell face ‘e’. The diffusive flux of \( \phi \), on the other hand, may be approximated over the CV as

\[
- \int_{S_x} \Gamma_{\phi} \frac{\partial \phi}{\partial x_i} dS_i \approx - \sum_i \left( \Gamma_{\phi} \frac{\partial \phi}{\partial x_i} n_{i} \right) s_{e}.
\]

The cell face mean values, appearing in the convective fluxes, were obtained by blending the upwind differencing scheme (UDS) and the central difference scheme (CDS) using the differed correction approach suggested by Khosla and Rubin [49]. It is known that the first-order UDS is a bounded discretization scheme that yields a diagonal-dominant, positive-definite coefficient matrix. However, the truncation error terms for such schemes contain even-order derivatives, which tend to diffuse sharp gradients in the solution. Therefore, in many cases, solutions are required on a finer grid to obtain sufficiently small truncation and diffusion errors. CDS is, however, formally more accurate than UDS, but is prone to generate unphysical oscillatory solutions (or even divergence of the iterative solution procedure) if employed on coarse grids and at relatively higher velocities. Employing a linear combination of UDS and CDS with the deferred correction approach, the coefficient matrix resulting from the discretization procedure still remains strictly positive definite and diagonally dominant. The deferred correction was implemented in the following manner:

\[
\phi_{e} = \phi_{e}^{UDS} + \gamma(\phi_{e}^{CDS} - \phi_{e}^{UDS}),
\]

where \( \gamma \) is a blending parameter.
where $\gamma$ is the blending factor, which is an input parameter for the computations and varies between zero (pure UDS) and unity (pure CDS). In the present study, $\gamma$ was chosen as unity.

The source term, $s_{phi}$, was integrated over the CV by indirectly assuming the mean value theorem as follows:

$$\int_V s_{phi} dV \approx s_{phi,p} V.$$  \hfill (18)

The sources and boundary conditions used in the present study are discussed in the following section.

2.2.5. Natural convection

The density changes with temperature, combined with gravity, produce buoyancy-driven or natural convection flow in the liquid. In order to treat the body forces, the Boussinesq approximation was employed, where the density was considered to be constant in the unsteady and convective terms and was allowed to vary only in the body-force term. Therefore, an additional term appeared as a source in the momentum equations as

$$s_{u(\text{bou,j})} = -\rho_{\text{ref}} g \beta (T - T_{\text{ref}}),$$ \hfill (19)

where $\rho_{\text{ref}}$ and $T_{\text{ref}}$ represents the reference density and temperature, respectively and $\beta$ is the isobaric expansion coefficient.

2.2.6. Boundary conditions

The governing equations were subjected to the following boundary conditions at the phase-change interface. Since the phase change of pure substances occur at a single temperature, the temperatures of liquid and solid at the interface has to be equal. Therefore, one may write

$$T_l = T_s = T_I,$$ \hfill (20)

where $T_I$ is the temperature at the interface, i.e., the melting (or solidification) temperature. The energy balance was applied at the interface (Stefan condition, see Fig. 2 for clarification) as follows:

$$\rho_L V_I \cdot n = \lambda_s \frac{\partial T}{\partial n} \bigg|_n - \lambda_l \frac{\partial T}{\partial n} \bigg|_l,$$ \hfill (21)

where $n$ is the unit normal to the interface, $V_I$ is the interface velocity vector and $L$ is the latent heat. It may be noted here that Eq. (21) relates the normal heat fluxes from both the phases to the normal velocity of the interface, which is required to obtain the new location of the interface as well as for implementation of the mass conservation equation. As a result, the tangential velocity component does not explicitly appear in the formulation. In addition to this, the mass fluxes were also balanced at the interface as follows:

$$\rho_l n \cdot (U_l - V_I) = \rho_s n \cdot (U_s - V_I).$$ \hfill (22)

The no-slip condition was also implemented while applying the boundary conditions for the momentum equations at the solid walls and the interfaces.

For wall boundaries, the mass fluxes in the pressure-correction equation were enforced to zero; however, at the solid–liquid interface, since mass is assumed to pass through the interface, the mass fluxes were equated to mass fluxes due to grid velocities for consistency.

2.2.7. Final form of discretized equations

Summation of the fluxes through all the faces of a given CV results in an algebraic equation which links the value of the dependent variable at the CV centre with the neighbouring values. The equation may also be written in a conventional manner as

$$A_P \phi_P = \sum_{nb} A_{nb} \phi_{nb} + b_{\phi}, \quad nb = E, W, N, S.$$ \hfill (23)

The coefficients $A_{nb}$ contain contributions of the neighbouring nodes, nb, arising out of convection and diffusion fluxes as defined by Eqs. (3)–(5). The central coefficient $A_P$, on the other hand, includes the contributions from all the neighbours and the transient term. In some of the cases, where source term linearization was applied, it also contained part of the source terms. $b_{\phi}$ contains all the terms those are treated as known (source terms, deferred corrections and part of the unsteady term).
2.2.8. **Pressure velocity coupling**

The coupling of the dependent variables was obtained on the basis of the iterative SIMPLE algorithm developed by Patankar and Spalding [50,51]. The solution started with a guessed pressure field. The linearized momentum equations were then solved in a sequential manner. The set of linear equations was relaxed and a prescribed number of “inner” iterations (sweeps) were performed with the solution algorithm, described later. This yielded the velocity fields. Since, in all probability, the initially guessed pressure field may be considered to be incorrect, the velocities may not satisfy the continuity equation (2). The pressure, the associated velocities and the mass fluxes were then required to be corrected from the discretized continuity equation, such that the corrected values of these variables satisfy the continuity constraint. The pressure-correction equation was derived from the continuity equation, the discrete form of which has the same structure as the general discretization equation (23). These equations were solved with the same solution algorithm as used for the momentum equations. Since in melting and solidification phenomena the flow is generally non-isothermal, the energy equation was treated next to account correctly for the buoyancy forces. Residual norms were computed for all the conservation equations and were normalized by appropriate reference quantities. If any of the normalized residual norms was above the convergence criterion, the algorithm returned to the momentum equations using the stored values of the dependent variables, mass fluxes and diffusivities in order to assemble new finite-volume coefficients. Otherwise, the iteration process was terminated.

In order to check the mass conservation, the values of the velocity components at the cell face centres were required. With a collocated variable arrangement straightforward linear interpolation would lead to decoupling of velocities and pressure and eventually, to the production of oscillatory (checkerboard) solutions (see [50] for details). Special interpolations, as suggested by Rhie [52], were therefore introduced to overcome the problem.

2.2.9. **Solution of algebraic equations**

For the solution domain as a whole, a system of \( N \) simultaneous equations were obtained with \( N \) unknowns, where \( N \) is the number of control volumes. The coefficient matrix of such a system contains non-zero elements only on the five diagonals. The strongly implicit procedure (SIP) of Stone [53], based on an incomplete LU factorization of the coefficient matrix, was used for the solution. If the source term is also a function of \( \phi \), then a suitable linearization may sometimes provide another positive contribution to the central coefficient of the matrix, which makes the iterative solution procedure more stable. Another way of enhancing the stability of the solution method, especially when the equations are non-linear and coupled, is under-relaxation.

In the present study, both the techniques were simultaneously used to accelerate convergence.

2.2.10. **Grid generation**

In the present moving grid formulation, two separate domains were considered for the melt and the solid. Once the converged solution for a particular time step had been obtained, from the Stefan condition, as mentioned earlier, the velocities of the interface nodes, lying at the centre of boundary surfaces of interface control volumes, were calculated. These nodes were then moved in the desired direction from the knowledge of the interface velocities and the chosen time step. The cell vertices of interface control volumes were subsequently placed by linear interpolation. It may be noted that the interface nodes are shared by both the liquid and the solid domains (phases) in the physical space. This procedure essentially completes the interface reconstruction.

For most engineering problems of practical interest, owing to the presence buoyancy (and also surface tension in some cases) driven convection, generally a non-uniform phase change occurs. As a result, if no special care is taken, the numerical grid at the interface becomes clustered near the region where the phase change takes place at a comparatively lower rate. It has been the experience of the present authors that non-uniform grid clustering often leads to numerical instabilities and the solutions subsequently diverge after a few time steps. In order to overcome this difficulty with the moving grid formulation and in order to achieve better control on the grid skewness at the interface, the nodes were allowed to slide on the interface. The new locations of the interface nodes were obtained by simple linear (or spline) interpolation using the normalized arc lengths as a parametric variable with an initial proportionate distance between the nodes as a basis. While obtaining the new positions, the use of parametric variable ensured that the geometry of the interface in the physical space remained unaltered. This procedure is briefly provided in Appendix A, with a few examples.
Once the new locations of the cell vertices at the interface has been obtained, the numerical grid in both the solid and the liquid domains were then generated by algebraic mapping [54]. In principle, for more complicated domains, other grid generation techniques, such as the elliptic grid generation method, may be adopted to achieve better control over the numerical grid. However this would also lead to more computational overhead.

2.3. Solution algorithm

The overall solution algorithm for the calculation of melting and solidification phenomena in irregular domains, by explicitly moving the phase-change interface, may now be summarised as follows:

- **Step 1**: Provide the initial grid for the solid and the liquid domains and values of the dependent variables (solution for time $t_0$, i.e., the initial condition).
- **Step 2**: Advance the time by $\delta t$ and solve the solid and the liquid domains independently.
- **Step 3**: Assemble and solve Eq. (13) for velocity components, employing available dependent variables.
- **Step 4**: Assemble and solve Eq. (13) for pressure correction and use calculated values to correct velocity components, pressure and mass fluxes.
- **Step 5**: Assemble and solve Eq. (13) for temperature, which is coupled with the momentum equations owing to buoyancy, and update the fluid properties (if the fluid properties are temperature dependent).
- **Step 6**: Return to step 3 and repeat until the sum of the absolute residuals in the momentum and continuity equations has fallen by the desired orders of magnitude. In the present computations, when the absolute residuals reached $1 \times 10^{-6}$ the solution was considered to be converged.
- **Step 7**: Using the newly solved temperature field, apply the Stefan condition for phase change at each nodal location at the interface to determine the new position of the interface. Then slide the boundary nodes along the interface to avoid clustering of the grid nodes near the region of rapid phase change.
- **Step 8**: Generate the numerical grid in both the liquid and the solid domains.
- **Step 9**: Calculate mass fluxes through cell faces and change of volume of cells due to the moving grid using Eq. (8) for non-axisymmetric geometry and Eq. (12) for axisymmetric geometry.
- **Step 10**: Advance the time by another increment $\delta t$ and return to step 2; repeat until the prescribed number of steps is completed.

3. Numerical results and comparisons

3.1. Solidification computations

In order to validate the mathematical formulation for the solidification processes, presented in Section 2, and the corresponding solution algorithm, computations were performed on tin solidification inside a rectangular cavity. The results were compared with the experimental data and the corresponding numerical predictions of Wolff and Viskanta [55]. Hence the comparison was performed with a set of data to simulate the experiments of Wolff and Viskanta [55], which is also commonly referred to in the literature. In the author’s computations, 99.99% tin was considered as a pure metal with constant properties for each phase. All the relevant thermophysical properties of tin were obtained from Hedges [56] and from the Metals Handbook [57], yielding

- densities: $\rho_l = \rho_s = 6970 \text{ kg/m}^3$,
- viscosities: $\mu_l = \mu_s = 2.3140 \times 10^{-3} \text{ kg/m s}$,
- thermal conductivities: $k_l = k_s = 56.5 \text{ W/m K}$.

Hence the material properties were known at all the points in the computational domain and, as mentioned in Section 2, the surface tension forces at the interface were neglected.
The actual computational domain was that of Wolff and Viskanta [55] and hence consisted of a region 8.89 cm long and 6.67 cm high. The entire cavity was initially filled with liquid tin at 233 °C. At time $t = 0$ the vertical solid wall at the right-hand side of the computational domain was suddenly subjected to a constant temperature of $T_c = 229 ^\circ$C, which is below the solidification temperature of tin. The left-hand side wall was maintained at $T_h = 233 ^\circ$C, i.e., at a temperature above the solidification temperature of tin. The top and bottom walls were considered to be adiabatic. The interface was kept at the solidification temperature of $T_I = 231.9 ^\circ$C. All these boundary conditions and the resultant dimensionless parameters are given in Table 1 and the schematics of the computational domain along with the coordinate system used in the present study are shown in Fig. 4. The liquid was assumed to be an incompressible Newtonian fluid and the flow was considered to be laminar and two-dimensional. The Boussinesq approximation was employed in order to account for the natural convection in the melt.

It is obvious from Section 2 that the present moving grid method requires two distinct domains for both phases. In the experiment of Wolff and Viskanta [55], however, the entire domain was initially filled with molten tin. This difficulty of non-existent solid phase at the initial state was overcome by creating a small (about 0.1% of the total length of the cavity) solid layer of constant thickness at the right-hand side of the computational domain. The conservation equations for mass, momentum and energy in the liquid and the energy conservation equation in the solid were then solved in time with the above initial and enforced boundary conditions given in Table 1. Computations for the movement of the interface resulted from Stefan’s balance of heat. All these are described in Section 2.

In order to carry out the numerical computations, the entire computational domain was first divided into $80 \times 60$ control volumes, with the liquid and the solid regions each containing $40 \times 60$ control volumes. The

| Boundary conditions | $T_{\text{cold}} = 229.0 ^\circ$C  
|                     | $T_{\text{hot}} = 233.0 ^\circ$C |  
|                     | No-slip condition at walls and solid–liquid interface |  
|                     | Stefan condition at the interface |  
| Initial conditions  | $T(\text{liq}) = 233.0 ^\circ$C |  
|                     | $u = v = 0$ |  
| Non-dimensional parameters | $Ra = 1.59 \times 10^5$ |  
|                     | $Ste = 1.32 \times 10^{-2}$ |  

Fig. 4. Schematic of the physical model used in tin and gallium simulations.
A numerical grid was also clustered near the solid–liquid interface and near the two isothermal walls for better resolution of the heat fluxes. In order to check the grid independency of the solutions, the grid resolution was further increased to $160 \times 120$, with equal number of control volumes in both the phases and as far as the interface locations are concerned, no visible variation was observed till $t = 1.462$ h, which is the maximum time chosen for the computational study. For the convection terms, a blending of central and upwind differencing schemes with a blending factor $\gamma = 0.8$ was chosen. The time discretization was carried out by Euler implicit method and a time step of 1.0 s was employed for the entire simulation. The solution was considered to be converged when the absolute residual of all the conservation equations became less than $1 \times 10^{-6}$, a value that is usually adopted for these kinds of computations.

The progression of the solid–liquid interface with time is shown in Fig. 5 and the corresponding numerical grids in both domains, which illustrate the movement of the grid, are shown in Fig. 6. At very early times (not shown), the solid–liquid interface turned out to be almost vertical (linear, without any curvature), indicating that the heat transfer was predominantly governed by conduction and had an almost linear temperature distribution throughout the entire computational domain. As time progressed, buoyancy-driven natural convection in the liquid metal became important and started to influence the local heat transfer rate at the solid–liquid interface. At 0.077 and 0.165 h, clear development of buoyancy roll was observed, typical of natural convection flows in differentially heated closed cavities. The interface became slightly curved at the regions where the flow impinges the solid–liquid interface. Near the bottom and top of the walls, the interface was more or less straight, which may be attributed to a reduction of heat flux due to the presence of the small recirculation zones trapped between the primary buoyancy roll and the solid regions as shown in Fig. 7. At these times the present predictions are in very good agreement with the predictions of Wolff and Viskanta [55]; however, some deviations from the experimental results were observed.

At later times (0.529 and 1.462 h), due to the movement of the solidification front towards left wall, the liquid region was substantially reduced. Multiple rolls were observed at the bottom of the liquid region that block the flow of the colder fluid downwards and result in a non-uniform heat flux distribution along the interface. The curvature of the interface was increased near the region where the primary buoyancy roll impinged on the interface. At a much later time (1.462 h), the present model shows very good agreement with the experimental data, which was otherwise not obtained by Wolff and Viskanta [55] in their numerical predictions. The apparent discrepancy between the experimental and numerical results may be attributed to various

![Fig. 5. Comparisons of the interface locations in tin solidification with the experimental and numerical results of Wolff and Viskanta [55].](image)
reasons. Wolf and Viskanta [55], for example, attributed this observation to the idealisation of the mathematical models and boundary conditions, which is otherwise absent in the experiments. However, the cause of this deviation may also be identified as the pour-out technique that was adopted in the experiment to identify the location of the interface.

3.2. Melting computations

Since the present algorithm is capable of handling any type of phase change problem with sharp interfaces, the convection-dominated melting of pure gallium inside a cavity was next simulated with the proposed algorithm as part of the validation exercise. The problem is designed to simulate the experimental results of Gau and Viskanta [58] since theirs is the most widely cited literature used for the verification of numerical models. The domain of computation is defined by a rectangular cavity that is 8.89 cm long and 6.35 cm in high and has a similar configuration to that shown in Fig. 4. The thermophysical properties of pure gallium are taken from [59]:

- \( \rho_l = 6.093 \times 10^3 \text{ kg/m}^3 \),
- \( \mu_l = 1.81 \times 10^{-3} \text{ kg/m s} \),
- \( k_l = 32.0 \text{ W/m K} \).

In the computation, following the experimental conditions, initially the entire solid gallium block and the cold wall were maintained at 28.3 °C. The melting temperature of gallium was taken as 29.78 °C [59]. The top

Fig. 6. Grid distribution in tin solidification at various time steps: (a) 0.077, (b) 0.165, (c) 0.529 and (d) 1.462 h.
Fig. 7. Streamlines and isotherms at various times for tin solidification. The isotherms are in °C and in the liquid plotted with $\Delta T = 0.2$ °C and in the solid with $\Delta T = 0.4$ °C: (a) 0.077 h; (b) 0.529; (c) 1.462.
and bottom walls were assumed to be adiabatic, whereas the hot and cold side walls were considered to be isothermal. At time $t = 0$, i.e., at the onset of the computation, the temperature of the hot wall was suddenly raised to 38°C, which initiated the melting process. In the experiment, the entire cavity was initially filled with solid gallium. However, as mentioned earlier, according to the present algorithm, a distinct liquid domain was necessary in order to start the computation. This problem was solved by selecting a very small liquid domain ($\delta = 0.1\%$ of the solid domain) at $t = 0$. The initial size of the solid domain was also changed to $\delta = 0.05\%$ of the original solid domain. The final results at subsequent time steps were observed to remain unchanged with the small change in the size of the initial liquid domain.

The dimensionless parameters and the boundary conditions for the gallium melting problem are given in Table 2. For the unsteady contribution, a fully implicit Euler scheme was used with a time step of 0.01 s. The solution was considered to be converged when the absolute residual of all the conservation equations became less than $1 \times 10^{-9}$. Other control parameters, for example, blending factor, etc., were kept same as used for solidification computations. The problem was simulated until 600 s of process time. In order to check the influence of numerical grid on the solutions, computations were carried out using $60 \times 80$ and $120 \times 160$ control volumes, with equal number of grids in both phases. The results obtained from this study are presented in Fig. 8 in the form of flow structure in the liquid region. It is obvious from the figure that with

Table 2

<table>
<thead>
<tr>
<th>Boundary conditions</th>
<th>$T_{\text{cold}} = 28.3$ °C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$T_{\text{hot}} = 38.0$ °C</td>
</tr>
<tr>
<td></td>
<td>No-slip condition at walls and solid–liquid interface</td>
</tr>
<tr>
<td></td>
<td>Stefan condition at the interface</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Initial conditions</th>
<th>$T_{\text{solid}} = 28.3$ °C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$u = v = 0$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Non-dimensional parameters</th>
<th>$Ra = 7.0 \times 10^5$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$Ste = 0.046$</td>
</tr>
<tr>
<td></td>
<td>$Pr = 0.0216$</td>
</tr>
</tbody>
</table>

Fig. 8. Gallium melting: comparison of flow structure at 90 s obtained with two different grids.
the present method, the number of rolls and the overall flow structure qualitatively remain the same for two different grids, however, it is also evident that the solution was yet to reach a grid independent state. At this point, it may be mentioned here that the main objective of the present paper is not to demonstrate the features of gallium melting problem, but to present a thermodynamically consistent approach to solve phase change problem on a moving grids. Hence, no further study was carried out to obtain a grid independent solution and all the subsequent results are presented for 60 × 80 grid.

It may be mentioned that for this particular problem, the flow structure in the melt has been a point of controversy since Dantzig [60] reported the occurrence of a multi-cellular flow structure in the liquid region. Other publications [38,61,62], however, reported that only a single cellular structure, or a weak secondary cell in addition to the primary cell, was observed during the melting process. Recently, Hannoun et al. [59], based on the enthalpy formulation, carried out studies on very fine grids using high-end computers to resolve this issue. They observed multiple convection rolls in the liquid region during the melting process, which confirms the observations of Dantzig [60]. They attributed the reasons for non-observance of multiple rolls in most of the earlier studies to insufficient grid resolution. Although this paper is primarily meant to present a numerical model for solving melting and solidification problems, an investigation was also made to ascertain the model’s capabilities to capture complex flow structures as observed by Hannoun et al. [59] using very fine grids for gallium melting.

Fig. 9 shows a comparison of the interface locations obtained from present computations with the experimental results of Gau and Viskanta [58] at two different times. The present predictions, particularly at shorter times, clearly show that the shape of the interface has distinct bulges, which were otherwise absent in the experimental observations. This may be attributed to the presence of multiple convective rolls in the flow structure. At longer times (10 min), better agreement was obtained with the experimental results. As shown in Fig. 10, four distinct rolls (at 90 s) were observed early in the simulation process. Further, as the solution time proceeds, with the change in the aspect ratio of the liquid region the number of convective cells decreases and, through a process of merging, only two major rolls are observed at 600 s. This observation of multiple rolls is in qualitative agreement with that of Hannoun et al. [59] obtained with very fine numerical grids. However, as mentioned earlier in this section, since the present solutions are not strictly grid independent solutions, the quantitative comparison of exact number of rolls and their locations cannot be made with those obtained by Hannoun et al. [59].

![Fig. 9. Gallium melting: validation of interface positions at 6 and 10 min with experimental results of Gau and Viskanta [58].](image-url)
3.3. Axisymmetric Czochralski crystal computations

In computations of crystal growth processes e.g. those utilizing the Czochralski method or the floating zone method, and also in computations of chemical vapor deposition, correct computations of liquid–solid interfacial phenomena play an important role. At presently, 95% commercial bulk single crystals are grown with the Czochralski crystal growth technique, hence the numerical modelling of this process is of the utmost importance. A literature survey showed that the essential difficulties in numerical modelling of the solidification process result from the available non-optimally operating boundary tracking methods. The correct incorporation of the crystal pulling process adds further to the computational difficulties.

In this section, the capability of the present formulation to handle the movement of the solid–liquid interface to simulate a complex crystal growth process, such as Czochralski crystal growth, is demonstrated. Only the case of buoyancy-driven flow, without any crystal or crucible rotation, was considered for simplicity. The movement of the solid–liquid interface was observed with respect to the changes in flow and temperature field within the melt and the crystal. All the data relevant to the present computation are provided in Table 3. This table contains information on the chosen material properties, details of the crystal and crucible geometries, the boundary and initial conditions employed and the dimensionless numbers defining the present crystal growth process.

Fig. 11 shows the typical Czochralski configuration used in this study. The geometric and physical details were obtained from Kopetsch [63]. A fixed pulling velocity of 8 mm/h was used and details of the pulling model are available in Kopetsch [41]. Several assumptions were made in the present simulations:

- The thermophysical properties of the crystal and melt were kept constant.
- The free surface was assumed to be planar and hence the deformation due to free surface forces were not within the scope of study.
- A constant pulling velocity was imposed on the interface where the crystal was located.
- The contact angle between crystal and the melt was kept constant and equal to 90°.
- Flow was assumed to be axisymmetric.

In the present computations, 40 × 60 control volumes were chosen in the melt and the crystal consisted 20 × 40 control volumes. The initial grid used is shown in Fig. 12(a). For unsteady computations the fully
The implicit Euler scheme was employed with a time step of 0.1 s. Flux blending was used for convective terms with an 80% contribution from the central differencing scheme ($\gamma = 0.8$). Iterations were continued until the absolute residual sources became less than $1 \times 10^{-6}$. The entire crystal growth process was simulated until 60 s of process time.

The material used in this study was Al$_2$O$_3$, which has a high Prandtl number. At 1 s the isotherms near the circumference of the crucible were almost vertical and showed a slight curvature, pressing against the iso-

---

**Table 3**

Parameters used in Czochralski crystal growth computations

<table>
<thead>
<tr>
<th>Material properties</th>
<th>$\rho_l = 3.05$ g/cm$^3$, $c_{pl} = 1.25$ J/g K</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$k_l = 0.1$ W/cm K, $\nu = 0.009$ cm$^2$/s</td>
</tr>
<tr>
<td></td>
<td>$\beta = 3 \times 10^{-7}$/K</td>
</tr>
<tr>
<td></td>
<td>$\rho_s = 3.97$ g/cm$^3$, $c_{ps} = 1.43$ J/g K</td>
</tr>
<tr>
<td></td>
<td>$k_s = 0.058$ W/cm K, $T$(melting point) = 2316.0 K</td>
</tr>
<tr>
<td></td>
<td>$\epsilon = 0.1$ (melt and crystal)</td>
</tr>
</tbody>
</table>

| Geometric details   | Crystal diameter = 1.5 cm                       |
|                     | Initial crystal height = 5.0 cm                 |
|                     | Crucible diameter = 5.0 cm                      |
|                     | Melt depth = 3.0 cm                             |

| Boundary conditions | $T$(crucible wall) = 2444.5 K, $T$(crystal top) = 1066.0 K |
|                     | $T$(ambient) = 350 K                             |
|                     | No slip condition at walls and solid–melt interface |
|                     | Stefan condition at the interface               |
|                     | Free surface condition at the melt surface ($v = 0$, $\partial u/\partial x = 0$) |
|                     | Radiation loss at the free surface and the side walls of the crystal |

| Initial conditions  | $T$(melt) = 2356.0 K                            |
|                     | $T$(crystal) = 1500.0 K                         |
|                     | $u = v = 0$                                    |

<table>
<thead>
<tr>
<th>Non-dimensional parameters</th>
<th>$Gr = 5.84 \times 10^6$ based on crucible diameter</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$Ra = 4.36 \times 10^5$ based on melt depth</td>
</tr>
<tr>
<td></td>
<td>$St = 0.15$</td>
</tr>
<tr>
<td></td>
<td>$Pr = 0.34$</td>
</tr>
</tbody>
</table>

---

**Fig. 11.** Schematic of the physical model used to simulate the Czochralski process (the dotted centre line represents the axis of symmetry assumed in the computations).
therms below the crystal and the free surface of the melt, as shown in Fig. 12(b). This also shows that flow due to natural convection was yet to be developed and the interface is more or less flat. As time progressed, the

Fig. 12. (a) Initial grid distribution. Subsequent diagrams show streamlines and isotherms at (b) 1, (c) 5, (d) 20, (e) 30 s and (f) 60 s. The Isotherms are in Kelvin and in the melt plotted with $\Delta T = 10$ K, in the crystal plotted with $\Delta T = 100$ K.
buoyancy-driven flow became stronger. This distorted the temperature field, as shown in Fig. 12(c)–(f), and a strong thermal gradient was observed at the centre of the crucible. The solid–liquid interface assumed a distinct convex and conical shape at later times. In this problem, the major cooling was generated owing to the withdrawal of the crystal. Since the pulling velocity was very small and, as a result, the melt height was lowered more slowly, the overall buoyancy force and hence the flow structure in the melt remained almost unchanged at sufficiently later stages. Therefore, no major effect on the shape of the interface was observed and hence one may assume that a quasi-steady feature was approached. The present predictions are also qualitatively in good agreement with those of Kopetsch [63].

4. Conclusions, final remarks and outlook

Melting and solidification processes occur in many fields of engineering and their importance is stressed in this paper by referring to the existing literature. Numerous experimental studies exist to quantify the change of location and shape of interfaces. The present paper underlines the importance of numerical computations of flows with heat and mass transfer that occur when melting and solidification processes are present in materials. It is important for correct numerical predictions that the formation of the interface between the liquid and solid faces is predicted in a thermodynamically correct way. This is occasionally not the case, because it is usually felt that this requires complicated and time-consuming grid generations after each time step. The present paper shows that this grid generation can easily be carried out and an interesting method has been developed to adapt the grid after each time step to the new interface location, taking the space conservation law fully into account. It also ensures that the numerical grid does not become distorted because of strong interface deformations. Hence the paper describes a method well suited to numerically predict melting and solidification processes such as occur with sharp interfaces in many fields of engineering.

The method described here is characterized by the following specific features:

- The present algorithm considers two (or more, if required) different domains for solid and liquid media and solves the corresponding conservation equations on generalised curvilinear coordinates, with a collocated variable arrangement. This feature is common for the other phase change algorithms that employ moving grids.
- The movement of the interface is calculated from the Stefan condition and hence the algorithm allows a sharp interface between the solid and liquid phases. This algorithm is, therefore, thermodynamically consistent for phase change problems with pure substances.
- Once the interface has been moved, the algorithm allows for sliding the nodes on the interface in order to avoid grid clustering at some regions of the interface. This also allowed the generation of the numerical grid in both phases by simple algebraic mapping and hence the computational effort generally associated with the moving grid formulation is greatly reduced.
- The volume (and hence the mass) fluxes due to the movement of the numerical grid are calculated explicitly by satisfying the space conservation law and thereby the present algorithm avoids the inclusion of artificial mass sources (or sinks) in order to satisfy the mass conservation equation. Although the moving grid algorithms gained popularity in crystal growth processes owing to their applicability, to the best of the authors knowledge, this feature was not clearly shown in any of the earlier papers in this field.

The performance of the developed grid-generating method is demonstrated and predictions were carried out for a solidification process, a melting process and the prediction of crystal growth utilizing the Czochralski method. The results clearly demonstrate the advantages of the present grid generation and grid adaptation in attempts to yield reliable predictions of the location of the interface between the liquid and solid faces when melting and solidification processes occur. The results obtained are encouraging and suggest the extension of the present method to three-dimensional melting and solidification processes. Such work is currently being carried out at the author’s Institute in an attempt to develop a general prediction procedure, based on the computer code FASTEST-3D, generally applicable to fluid flow problems with heat and mass transfer that result in melting and solidification processes.
Acknowledgements

Dr. Subhashis Ray, thanks the authorities of Jadavpur University for granting leave to carry out this research work. He also thanks the Alexander von Humboldt Foundation for financing his research stay at the Institute of Fluid Mechanics (LSTM-Erlangen). Mr. Santhanu Jana also thanks the authorities of BMBF for financing his study and research stay at LSTM-Erlangen through a grant under Project No. IND 00/002.

Appendix A

A.1. Algorithm to slide grid points on the solid–liquid interface

In this Appendix, the algorithm for sliding the grid nodes on the interface is presented. In order to prevent clustering of the nodes on the interface arising out of non-uniform phase changes that occurs mainly due to the buoyancy-induced convective flow in the melt. The first constraint that the algorithm must satisfy is that the physical location of the interface must remain unaltered after the redistribution of the nodes. In addition, one would also like to place the nodes at the same proportional distances with respect to each other as they were placed for the initial configuration.

For two-dimensional problems, the interface is located at one of the ends of the computational domain, where one of the curvilinear coordinates (either \( \xi_1 \) or \( \xi_2 \)) remains constant. Such an interface is defined by a set of coordinates \((x_j, y_j)\), where \( j = 1, 2, \ldots, N \). The main job is, therefore, reduced to finding new set of coordinates \((x_{nj}, y_{nj})\) on the original curve (interface), such that the proportional distances between two successive points remain same as that of the original configuration. In order to satisfy these criteria, the normalized distance of the given points along the interface was calculated, which was subsequently used as the parametric variable for interpolation. From the given data points \((x_j, y_j)\), the distances between two consecutive points \(D_{sj}/C_{213}\) were first calculated as follows:

\[
D_{sj} = \left( (x_{j+1} - x_j)^2 + (y_{j+1} - y_j)^2 \right)^{1/2}.
\]  

(24)

For the first grid point \((j = 1)\), the distance along the interface is set to zero, i.e., \(s_1 = 0\). Using Eq. (24), the new parametric coordinate along the curve \(s_j\) was obtained as

\[
s_j = s_{j-1} + \Delta s_{j-1}.
\]  

(25)

Therefore, the total length along the interface was given by \(s_{tot} = s_N\). Using this total length \(s_{tot}\), all the values of \(s_j's\) were then normalized as

\[
s_j = \frac{s_j}{s_{tot}} \text{ for } j = 1, N.
\]  

(26)

The cubic spline fitting routine was then called upon and both \(x_j\) and \(y_j\) were fitted with respect to the normalized parametric variable \(s_j\). For example, while fitting \(x_j\) with \(s_j\), the following expression was used:

\[
x = Ax_j + Bx_{j+1} + Cx''_{j} + Dx''_{j+1},
\]  

(27)

where \(A\), \(B\), \(C\) and \(D\) are given as follows:

\[
A = \frac{s_{j+1} - s}{s_{j+1} - s_j}
\]  

(28a)

\[
B = 1 - A = \frac{s - s_j}{s_{j+1} - s_j}
\]  

(28b)

\[
C = \frac{1}{6} (A^3 - A)(s_{j+1} - s_j)^2
\]  

(28c)

\[
D = \frac{1}{6} (B^3 - B)(s_{j+1} - s_j)^2
\]  

(28d)
The second derivatives appearing in Eq. (27) may be obtained by solving a set of simultaneous equations, which are generated from the requirement that the first derivatives, computed from Eq. (27), be continuous across the boundary between two intervals (for details, see [64]). For linear interpolation, the values of $C$ and $D$ were set to zero.

![Graph showing original curve and interpolated points](image1)

![Grid distributions for tin solidification and Czochralski crystal growth processes](image2)

Fig. 13. New grid points on the original Sine function curve equation (29). The constants selected are $A_{\text{mean}} = 2.5$, $L = 5.0$, $A_{\text{amp}} = 2.5$ and number of grid nodes $NI = 42$ (a). Grid distribution for tin solidification after 1.462 h of process time without, (b) and (c) with using sliding algorithm. Grid distribution for the Czochralski crystal growth process without, (d) and (e) with sliding algorithm.
The original normalized proportionate distances along the interface were already known and let them be denoted by $s_{nj}$. Once, the piecewise cubic (or, linear) functions were fitted for both $x_j$ and $y_j$, the new interpolated values of $x_{nj}$ and $y_{nj}$ were then easily obtained by using the respective cubic spline (or linear) fitting and the variable $s_{nj}$ as the new location of the parametric variable on the interface. The use of parametric variable ensures that the newly determined points lie on the original curve, defined by $(x_j, y_j)$. Moreover, this procedure not only avoids tedious iterations for finding out the new locations for the grid points, it also ensures that the original proportionate distances along the curve is maintained by the new locations $(x_{nj}, y_{nj})$.

In order to validate the algorithm, first an example, where, the interface is represented by a Sine curve in the following form:

$$x_j = A_{\text{mean}} + A_{\text{amp}} \sin(2.0\pi y_0_j/L)$$  \hspace{1cm} (29)

is shown in Fig. 13(a). The figure clearly shows that the new locations of nodes lie exactly on the original interface. Therefore, it may be concluded that the present algorithm for sliding the nodes on the interface works satisfactorily.

In addition to this, two other examples are also presented in Fig. 13: one, where the solidification of tin inside a rectangular cavity is considered and the other, where the crystal growth problem by Czochralski method is solved. In these Fig. 13(b) and (d) are presented for the cases where no precaution is taken and the grid distortion is allowed and Fig. 13(c) and (e) are presented for the cases where the nodes on the interfaces are redistributed according to the present algorithm. These figures clearly show that if the nodes are not reallocated on the interfaces then severe grid distortion may occur, particularly in the region where the phase change process is slower, which eventually may lead to divergence of the solution. It may also be mentioned here, that when the numerical grid is heavily distorted, the time step requirement becomes more stringent to achieve convergence and hence, the benefit of using an implicit scheme with moving meshes, as presented in this paper, is considerably reduced.

References