Numerical Heat Transfer, Part B, 42: 1–21, 2002

1040-7790/02 $12.00 + .00
DOI:

A FINITE-VOLUME SHARP INTERFACE SCHEME FOR DENDRITIC GROWTH SIMULATIONS: COMPARISON WITH MICROSCOPIC SOLVABILITY THEORY

H. S. Udaykumar and L. Mao
Department of Mechanical Engineering, University of Iowa, Iowa City, Iowa 52242, USA

R. Mittal
Department of Mechanical Engineering and Aersopace Engineering, George Washington University, Washington, DC 20052, USA

We present and validate a numerical technique for computing dendritic growth of crystals from pure melts. The solidification process is computed in the diffusion-driven limit. The governing equations are solved on a fixed Cartesian mesh and a mixed Eulerian-Lagrangian framework is used to treat the immersed phase boundary as a* sharp *solid–fluid interface. A conservative finite-volume discretization is employed which allows the boundary conditions to be applied exactly at the moving surface. The results from our calculations are compared with two-dimensional microscopic solvability theory. It is shown that the method predicts dendrite tip characteristics in good agreement with the theory. The sharp interface treatment allows discontinuous material property variation at the solid–liquid interface. Calculations with such discontinuities are also shown to produce results in agreement with solvability and with other sharp interface simulations.

1. INTRODUCTION

In this article, a method for computing the evolution of dendritic phase boundaries on fixed Cartesian grids is presented and validated. The phase front is treated as a sharp boundary that runs through the mesh, and a conservative finite-volume method is applied for solution of the field equations. The numerical solutions are validated against the widely accepted microscopic solvability theory [1, 2].

In previous work [3] we described a finite-difference technique for simulation of diffusion-controlled growth of unstable phase boundaries. For the case of dendritic solidification of pure materials from the melt, we demonstrated that a *sharp interface* approach can be developed that yields globally second-order accurate solutions to the field equations. The interface position was computed with first-order accuracy.

Received 31 August 2001; accepted 19 April 2002.
This work was supported by a National Science Foundation Career Award (CTS-0092750) to H.S. Udaykumar.
Address correspondence to Prof. H.S. Udaykumar, University of Iowa, 2408 Seamans Center, Iowa City, IA 52242, USA. E-mail: ush@icaen.uiowa.edu

In subsequent work [4, 5], we devised a finite-volume approach for solving the incompressible Navier-Stokes equations to compute incompressible flows around fixed and moving immersed solid bodies. The switch from finite-difference to finite-volume method was prompted by the fact that the former failed to produce adequate satisfaction of the zero-divergence constraint on the velocity field at points adjoining the immersed boundaries. This was because of the lack of explicit flux conservation in the finite-difference method at the irregular stencils that arise when a solid boundary cuts through the Cartesian mesh. In contrast, the finite-volume technique enforced explicit flux conservation across the common faces of the control volumes, as described later in this article. Thus, stable divergence-free velocity calculations were facilitated.

This article advances the work presented by Udaykumar et al. [3] in several aspects. As in the case of the Navier-Stokes solutions presented by Udaykumar et al. [5], fluxes (of heat in the present case) are explicitly conserved across the control-volume faces. Using a compact linear-quadratic interpolant [4] to evaluate the fluxes at the control-volume faces that are cut by the immersed interface, a second-order-accurate flux evaluation procedure is devised. The technique is applied to study dendritic growth in undercooled melts. The present work places the finite-volume scheme on sound footing by directly comparing the results of the computations with two-dimensional microscopic solvability theory. The method is shown to predict the correct physical behavior in the dendritic growth of pure materials. This validation effort augments the demonstration of grid size and orientation independence and convergence studies for the sharp interface method, presented by Udaykumar et al. [3].

2. SHARP VERSUS DIFFUSE INTERFACE METHODS

In recent years there has been increased activity in the area of simulation of solidification microstructures, particularly dendritic growth. Various techniques, including phase fields [6–8], level sets [9, 10], finite elements [11, 12], and finite difference front tracking [3, 13], have been employed to simulate this pattern-forming phenomenon. The phase-field method is currently the most popular approach and was the first approach used [14, 15] to determine whether the quantitative measures, such as tip radius, velocity, and the selection parameter of the numerically grown dendrites, could be validated against microscopic solvability theory, which is generally accepted as describing the physics of dendritic crystal growth. The phase-field method falls under the class of Eulerian (i.e., fixed-grid) methods called ''diffuse interface'' methods [16]. In such methods, as shown in Figure 1*a*, the interface is not a sharp phase boundary but is given a certain thickness or spread on the computational mesh. Typically, the interface thickness occupies a few mesh cells. Thus, both delta (singular sources residing on the interfaces, such as latent heat and capillarity) and Heaviside (such as jump discontinuities in material properties) functions are replaced by smoothed numerical approximations (Figure 1*b*) in diffuse interface methods. This is in contrast to the class of ''sharp interface'' methods [3, 9, 17], where these functions are retained as discontinuities (Figure 1*c*). Comparison of a sharp interface method with solvability in two dimensions was performed by Dantzig and co-workers [10, 18, 19]. They used a level-set technique for tracking the front and a finite-element scheme for solution of the diffusion equation. Here we use

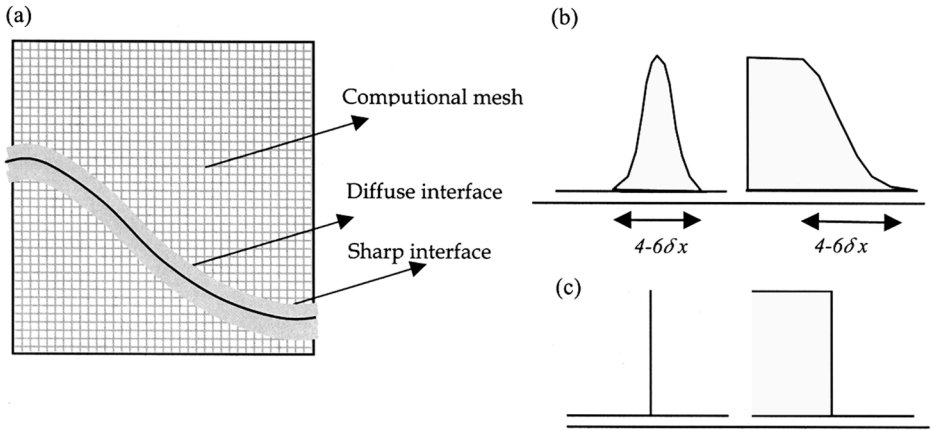

Figure 1. (*a*) Illustration of the difference between sharp and diffuse interface representation on a fixed mesh. (*b*) Smoothed forms of delta and Heaviside forms seen by diffuse interface methods. (*c*) Delta and Heaviside functions are maintained as discontinuities in sharp interface methods.

a finite-volume method and explicit front tracking to compute the evolution of sharp dendritic fronts. In combination with the Navier-Stokes solver for moving boundaries, developed by Udaykumar et al. [5], the present method will enable computation of fluid flow effects on dendrites as demonstrated in preliminary investigations by Udaykumar et al. [20].

3. SOLVABILITY THEORY

Microscopic solvability is the currently accepted theory for the growth of dendritic structures in solidification from the melt. Extensive reviews on the subject are available (e.g., Kessler et al. [2]). For solidification from pure melts, the basic features of the theory are as follows.

The Mullins-Sekerka instability [1] of a planar solidification front occurs in the case when the melt is undercooled, i.e., the temperature of the melt T_∞ is depressed below the melting temperature of the planar front T_m. Linear stability analysis of phase fronts under such conditions predicts a continuum of solutions in the form of paraboloidal needle crystals, which were obtained by Ivantsov [21] as

$$\Delta = \sqrt{\pi\,\mathrm{Pe}}\,\exp(\mathrm{Pe})\,\mathrm{erfc}(\sqrt{\mathrm{Pe}}) \tag{1}$$

where Δ is the dimensionless undercooling [$\Delta = (T_m - T_\infty)/(L/C_p)$, L is the latent heat and C_p is the specific heat at constant pressure], Pe is the tip Peclet number given by

$$\mathrm{Pe} = \frac{\rho V_t}{2\alpha} \tag{2}$$

where ρ is the radius of the paraboloidal dendrite tip, V_t is the steady-state tip velocity, and α is the thermal diffusivity as shown in Figure 2. However, in nature, one observes unique crystal patterns that are selected by the growth conditions and

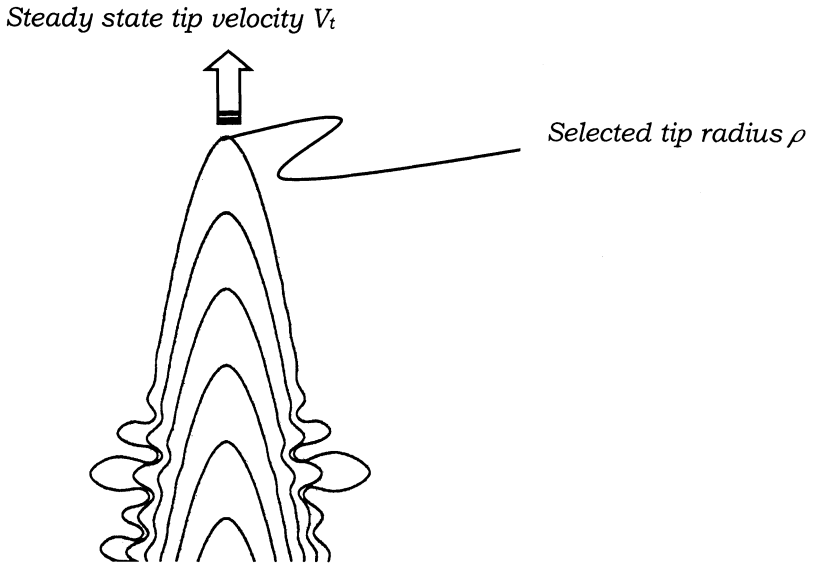

Figure 2. Illustration of important tip characteristics for a dendritic crystal. The tip is a paraboloid of revolution and propagates at a selected steady velocity and constant tip radius.

material properties. The selection of a pattern from the continuum of unstable wavelengths becomes possible by introduction of surface tension, which provides a smoothing or restabilization mechanism and picks out a spectrum of solutions. The influence of the surface tension appears through the modulation of the interfacial temperature for solidification fronts with curvature by means of the Gibbs-Thomson condition (without including interfacial kinetics effects):

$$T_i = T_m - \frac{\gamma(\theta) T_m}{L} \kappa \tag{3}$$

In the above, κ is the interfacial curvature and the surface tension $\gamma(\theta)$ is a function of the crystalline anisotropy (θ being the angle made by the normal to the interface with the x axis). The strength and directional dependency of the surface tension are determined by the structure of the solid formed. For a typical fourfold symmetric crystal (such as the popular transparent organic model material succinonitrile [22]), this function could assume the form [18]

$$\gamma(\theta) = \gamma_0(1 - 15\varepsilon \cos 4\theta) \tag{4}$$

A unique operating point of the dendritic tip is then selected from the set of allowed solutions by crystalline anisotropy through the solvability mechanism [2] by stipulating that a material-dependent selection parameter σ^* be related to the tip radius and velocity as follows:

$$\sigma^* = \frac{2\alpha d_o}{\rho^2 V_t} \tag{5}$$

where d_o is a capillary length defined as

$$d_0 = \frac{\gamma_0 T_m C_p}{L^2} \tag{6}$$

The selection parameter σ^* is a function of the crystalline anisotropy ε and is typically of the form

$$\sigma^* = \sigma_0 \varepsilon^{7/4} \tag{7}$$

The unique geometric features that are displayed by growing dendrites when a steady tip propagation state has been reached are the tip radius (ρ) and velocity (V_t), as shown in Figure 2. Eqs. (1), (2), and (5) above provide the necessary relations for obtaining these two unknown selected parameters in terms of the control parameters, namely, the undercooling Δ, capillarity parameter γ_0, and anisotropy strength ε.

4. GOVERNING EQUATIONS

To test the numerics, we grow a dendritic front for specified control parameters, namely, the undercooling (Δ), capillary length (d_o), and anisotropy strength (ε). The predicted selection parameter (σ^*) and selected tip radius (ρ) and velocity (V_t) are then compared with the results of solvability theory.

The following nondimensionalization is adopted: length $X = x/d_0$, time $\tau = t\alpha_l/d_0^2$, velocity $V^* = Vd_0/\alpha_l$, and temperature $T^* = (T - T_m)/L/C_p$. The heat conduction equation is solved in the solid and liquid phases separately, in nondimensional form:

$$\frac{\partial T^*}{\partial t} = \alpha_i \nabla^2 T^* \tag{8}$$

where α_i is the thermal diffusivity in the ith phase, where $i = 1$ in the solid phase and $i = 2$ in the liquid phase. The interface temperature equation, Eq. (3), reads in nondimensional form:

$$T_i^* = -d(\theta)\kappa^* \tag{9}$$

where

$$d(\theta) = d_0(1 - 15\varepsilon \cos 4\theta) \tag{10}$$

For phase change of pure materials the normal velocity at a point on the front is provided by the rate of transport of latent heat away from the solid–liquid interface (Stefan condition). In nondimensional form, this equation is

$$V_i^* = \left[\left(\frac{\partial T^*}{\partial n}\right)_1 - \left(\frac{\partial T^*}{\partial n}\right)_2\right] \tag{11}$$

where subscripts 1 and 2 apply to solid and liquid, respectively. Note that the temperature boundary condition at the far field for a crystal growing in the undercooled melt then becomes, in nondimensional form,

$$T^*(r \to \infty, t) = \Delta \tag{12}$$

5. THE NUMERICAL METHOD

5.1. Discrete Form of the Governing Equations

The present method performs temperature field computations on a fixed Cartesian mesh, while the solid–liquid front evolves through the mesh. The interface is tracked using markers connected by piecewise quadratic curves parametrized by the arclength [3]. In Ye et al. [4] we provided details regarding the interaction of the interfaces with the underlying fixed Cartesian mesh. These include obtaining locations where the interface cuts the mesh, identifying phases in which the cell centers lie, and procedures for obtaining a consistent mosaic of control volumes in the cells crossed by the immersed interface. This results in the formation of control volumes near the interface that are, in general, trapezoidal in shape (see Figure 3).

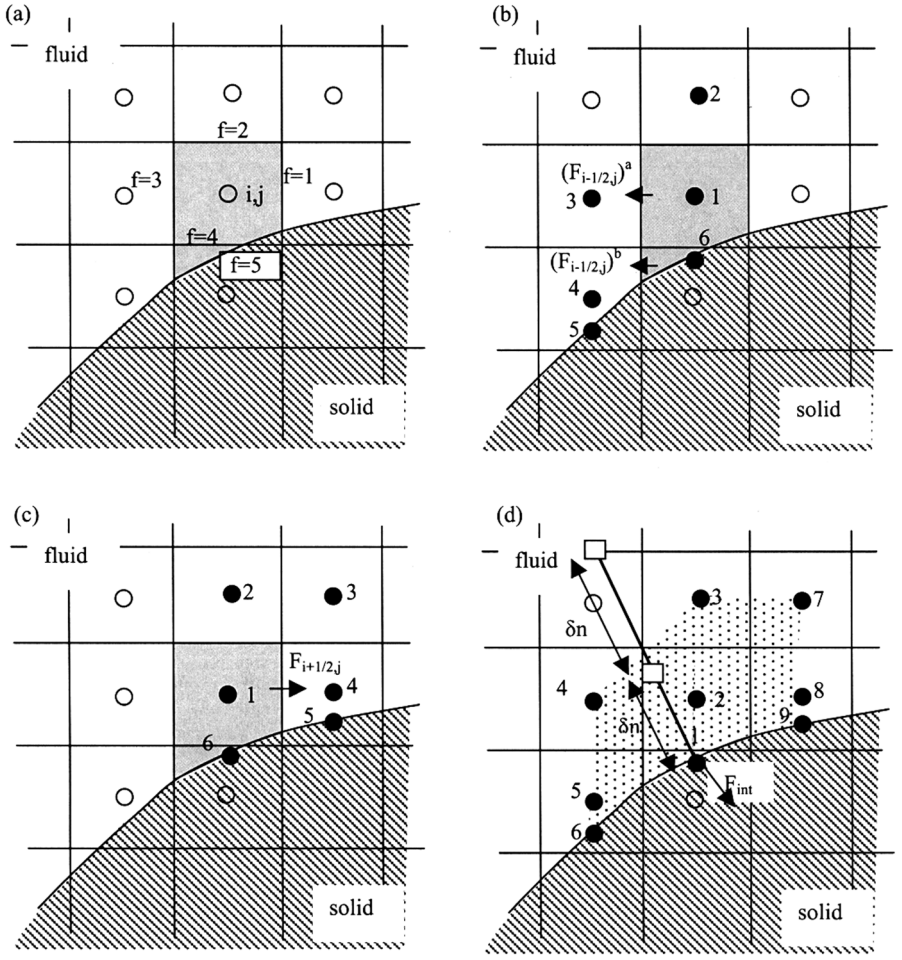

Figure 3. Illustration of stencils for evaluation of cell face fluxes. (*a*) Interfacial cell nomenclature showing for numbering the faces of the (possibly) five-sided control volume. (*b*) Stencil points for linear-quadratic interpolation to obtain the flux $F_{i-1/2,j}$. (*c*) Stencil points to calculate flux $F_{i+1/2,j}$. (*d*) Stencil points to calculate flux F_{int}.

The finite-volume discretization is then performed over the regular Cartesian grid cells in the bulk of the computational domain and a lower-dimensional set of irregularly shaped cells that adjoin the interface.

The energy equation, Eq. (8), is written in semidiscrete form as (after dropping the asterisk on the nondimensional quantities)

$$\int_{\mathrm{v}} \frac{T^{n+1} - T^n}{\delta t}\, dV = \frac{\alpha_i}{2} \oint (\nabla T^{n+1} + \nabla T^n) \cdot \hat{n}\, dS \tag{13}$$

The above Crank-Nicolson scheme provides nominal second-order temporal accuracy.

In discrete form Eq. (13) is written, for a control volume in the Cartesian mesh indexed (i, j), as

$$\frac{\Delta V_{ij}}{\delta t}(T_{ij}^{n+1} - T_{ij}^{n}) = \frac{\alpha_i}{2} \sum_{f=1}^{5} \left(\frac{\partial T^{n+1}}{\partial n} + \frac{\partial T^n}{\partial n} \right)_f \Delta S_f \tag{14}$$

In the above, the subscript f for the summation runs over the faces of the control volume (see Figure 3*a*).

In Eq. (14), the summation runs over the sides of the irregularly shaped (four- or five-sided) control volumes. The finite-volume discretization requires evaluation of the diffusive fluxes at the faces of each control volume, viz.,

$$F_d = \nabla T \cdot \boldsymbol{n} \tag{15}$$

For a uniform Cartesian mesh, the fluxes on the face centers can be computed to second-order accuracy with a linear profile for the temperature field between neighboring cell centers. This is not the case for a trapezoidal boundary cell, since the centers of some of the faces of such a cell may not lie halfway between neighboring cell centers. This is seen from Figures 3*b* and 3*c*, where the points at which fluxes are evaluated are indicated by the filled arrows. A linear approximation would not provide a second-order-accurate estimate of the gradients. Furthermore, some of the neighboring cell centers do not even lie on the same side (i.e., phase/material) of the immersed boundary and therefore cannot be used in the differencing procedure. Thus, not only do we need an accurate procedure for computing these face-center quantities, we also require that the procedure be capable of systematically handling reshaped boundary cells with a wide range of shapes. This is done using a compact two-dimensional polynomial interpolating function, described in [4], which allows us to obtain a second-order-accurate approximation of the fluxes and gradients on the faces of the trapezoidal boundary cells from available neighboring cell-center values. The current interpolation scheme coupled with the finite-volume formulation guarantees that the accuracy and conservation property of the underlying algorithm is retained even in the presence of arbitrary-shaped immersed boundaries. This has been demonstrated in [4] for stationary immersed boundaries and in [5, 20] for moving solid boundaries embedded in flows. We now employ the method described in [4] to obtain the discretization of Eq. (13) in control volumes that lie adjacent to the moving boundary.

First, consider the fluxes at the face $(i - 1/2, j)$ of the cell (i, j) shown in Figure 3*a*. To conserve heat fluxes between adjoining cells, we split the flux at this face into two

parts as shown in the figure. Thus, $F_{i-1/2,j} = F^a_{i-1/2,j} + F^b_{i-1/2,j}$. Approximation of $F^a_{i-1/2,j}$ to second-order accuracy is quite straightforward and is done in the same way as for the face of a nonboundary cell. For instance, if $F^a_{i-1/2,j}$ corresponds to a diffusive flux at the cell face $(i-1/2, j)$, one obtains the gradients for the diffusive flux from

$$\left(\frac{\partial T}{\partial x}\right)^a_{i-1/2,\,j} = \frac{T_i - T_{i-1}}{x_i - x_{i-1}} \tag{16}$$

For a uniform mesh, this yields second-order-accurate evaluation for the flux at that cell face. For cell (i,j) in Figure 3, evaluation of $F^b_{i-1/2,\,j}$ or $F_{i+1/2,\,j}$ to second-order accuracy is somewhat more complicated. Expressions like Eq. (15) cannot be used, since in many instances some of the neighboring nodes lie in the opposite phase at the immersed boundary. For instance, for the situation shown in Figure 2*a*, the node $(i,j-1)$ is inside the immersed boundary and cannot be used in the evaluation of $F^b_{i-1/2,j}$. Even if neighboring nodes are available, as they are for the east face, $F_{i+1/2,j}$ is to be evaluated not on the line joining the neighboring cell centers (i,j) and $(i+1,j)$, but at the center of that face as shown in Figure 3*c*. A linear-quadratic interpolant is used to evaluate this flux as detailed in [4]. For instance, in order to approximate $F^b_{i-1/2,j}$, we approximate the temperature field $T(x,y)$ in the shaded trapezoidal region shown in Figure 2*b* in terms of a function that is linear in x and quadratic in y:

$$T(x,y) = c_1xy^2 + c_2y^2 + c_3xy + c_4y + c_5x + c_6 \tag{17}$$

where c_1–c_6 are six unknown coefficients. Once the interpolating function is obtained, the desired gradient of T at the cell face can be calculated from

$$\frac{\partial T}{\partial x} = c_1y^2 + c_3y + c_5 \tag{18}$$

In Ye et al. (1999), we demonstrated numerically that the linear-quadratic interpolating function in Eq. (17) does indeed result in second-order-accurate evaluation of values and derivatives for fluxes such as $F^b_{i-1/2,j}$. It can be seen in Figure 3*b* that the points from which the values are drawn to evaluate the coefficients c_1–c_6 in Eq. (17) correspond to four nodal points (1–4) and two boundary points (5 and 6). These two boundary points are where the interfacial conditions are supplied in the discretization of the governing equations. The effect of the moving phase boundary is felt, in part, through the values imposed at these boundary points. Thus, the six unknown coefficients in Eq. (17) can be expressed in terms of the values of T at these six points in the form

$$c_k = \sum_{l=1}^{6} b_{kl}T_l \qquad k = 1,\ldots,6 \tag{19}$$

where b_{kl} are the elements of the inverse of the Vandermonde matrix [4]:

$$V = \begin{bmatrix} x_1y_1^2 & y_1^2 & x_1y_1 & y_1 & x_1 & 1 \\ . & & & & & \\ . & & & & & \\ . & & & & & \\ x_6y_6^2 & y_6^2 & x_6y_6 & y_6 & x_6 & 1 \end{bmatrix}$$

To evaluate $T^b_{i-1/2,j}$ and its derivative at the required point (x_o, y_o) on the face $(i-1/2,j)$ of the control volume, one uses Eqs. (17) and (18) to write

$$T^b_{i-1/2,j} = \sum_{l=1}^{6} \alpha_l T_l \tag{20}$$

where

$$\alpha_l = b_{1l}x_0y_0^2 + b_{2l}y_0^2 + b_{3l}x_0y_0 + b_{4l}y_0 + b_{5l}x_0 + b_{6l} \tag{21}$$

Also,

$$\left(\frac{\partial T}{\partial x}\right)^b_{i-1/2,j} = \sum_{l=1}^{6} \beta_l T_l \tag{22}$$

where

$$\beta_l = b_{1l}y_0^2 + b_{3l}y_0 + b_{4l}y_0 + b_{5l} \tag{23}$$

Note that α and β are purely geometry-dependent interpolation coefficients that depend on the mesh, and the location and orientation of the immersed boundary. Once the coordinates of the six points in Figure 3*b* are identified, the coefficients α_l and β_l are obtained directly from Eqs. (21) and (22), respectively. As such, α_l and β_l are nothing but the weight coefficients applying to the nodal values in the discrete form of the governing equation. A similar interpolation procedure is also used for approximating $F_{i+1/2,j}$. In general, there are also boundary cells that have their north or south faces cut by the immersed boundary. The approach described above applies to the evaluation of the fluxes on such faces, the only difference being that the interpolating function is then linear in y and quadratic in x. Using this procedure, the discrete form with strict flux conservation and second-order-accurate flux evaluations can be obtained for transport equations involving both advection and diffusion[4].

Now we turn to the calculation of the flux on the cell face 5 which lies on the immersed boundary as shown in Figure 3*a*. The value of the transported quantity at the interface, T_{int}, is usually available from a specified boundary condition at the interface and hence no interpolation is required for the value of the variable itself. Here we describe the approximation procedure for the normal derivative, given on the interface segment by

$$\left(\frac{\partial T}{\partial n}\right)_{\text{int}} = \left(\frac{\partial T}{\partial x}\right)_{\text{int}} n_x + \left(\frac{\partial T}{\partial y}\right)_{\text{int}} n_y \tag{24}$$

where n_x and n_y are the components of the unit vector normal to face 5. Therefore computation of the normal flux requires estimation of $\partial T/\partial x$ and $\partial T/\partial y$ at the center of that face. For the cell (i,j) in Figure 3, $\partial T/\partial y$ is computed to second-order accuracy by expressing the variation along the vertical line in terms of a quadratic in y as follows:

$$T = a_1y^2 + a_2y + a_3 \tag{25}$$

As in Eq. (21), the coefficients in the quadratic can be expressed in terms of the values of ϕ at the three points indicated in Figure 3*d*. Note that the boundary point

incorporates the appropriate interfacial condition expressing the effect of the moving boundary. Then the y derivative at the center of face 5 is evaluated as

$$\left(\frac{\partial T}{\partial y}\right)_{\text{int}} = 2a_1 y_{\text{int}} + a_2 = \sum_{l=1}^{3} \beta_l T_l \tag{26}$$

where again the β_j are interpolation coefficients which depend solely on the geometry of the boundary cell.

To obtain $(\partial T/\partial x)_{\text{int}}$, an approach consistent with the computation of the fluxes at the faces of the control volume is devised to estimate the value of this derivative to desired accuracy. Consider the trapezoid shown in Figure 3*d*. Expressing the variable in this trapezoidal cell in terms of an interpolating function which is linear in x and quadratic in y allows us to obtain a second-order-accurate approximation to $(\partial T/\partial x)_{\text{int}}$ at the center of the cell face 5. The procedure for this follows along lines similar to that shown for $(\partial T/\partial x)^{b}_{i-1/2,j}$ and we get the following form for the x derivative on the interface:

$$\left(\frac{\partial T}{\partial x}\right)_{\text{int}} = \sum_{l=4}^{9} \beta_l T_l \tag{27}$$

The index j in the summation runs over the points 4–9 in Figure 2*d*. Finally, using Eqs. (26) and (27) in Eq. (24), we write

$$\left(\frac{\partial T}{\partial n}\right)_{\text{int}} = \sum_{l=1}^{9} \beta_l T_l \tag{28}$$

for the normal gradient, where β_l is again geometry dependent. Thus we obtain a nine-point stencil for the gradient at the interface and the points in this stencil are shown in Figure 3*d*. As can be seen from Figure 3*d*, of these nine points, three points lie on the immersed boundary. The values of the temperature at such points are available from the prescribed boundary condition at the moving boundary, Eq. (9).

Note that the procedure for discretization above enables the formulation of fluxes using the general forms:

Gradients (for diffusive fluxes) at the noninterfacial sides of the control volume $F_f = (\sum_{j=1}^{6} \beta_j T_j)_f$

Gradients (for diffusive fluxes) at the interfacial sides of the control volume $F_{\text{int}} = (\sum_{j=1}^{9} \beta_j T_j)_{\text{int}}$

In the above, subscript f stands for the face of the control volume (faces 1–4, Figure 3*a*), and subscript "int" for the interfacial side (side 5, Figure 3*a*). Substitution of these expressions in the Eq. (14) results in a general discrete form:

$$\left(\frac{T_{ij}^{n+1} - T_{ij}^{n}}{\delta t}\right)\delta V_{ij} = \alpha_i \sum_{f=1}^{4} \frac{1}{2}\left[\left(\sum_{l=1}^{6} \beta_l T_l^{n+1}\right)_f + \left(\sum_{l=1}^{6} \beta_l T_l^{n}\right)_f\right] dS_f + \alpha_i \left(\sum_{l=1}^{9} \beta_l T_l\right)_{\text{int}} dS_{\text{int}} \tag{29}$$

which can be written as

$$\sum_{l=1}^{l_{\max}} a_l T_l^{n+1} = S(T^n, T_{\text{int}}^{n+1}) \tag{30}$$

where the explicit terms, boundary and interface contributions, and the accompanying interpolation coefficients are absorbed in the source term $S(.)$. The summation runs over all the $l_{\max}$ computational points that are included in the stencils for the cell-face flux evaluations. The current computational point ij is of course also included in the $l_{\max}$ stencil points. In cells away from the interface, as usual, $l_{\max} = 5$, while for the interfacial cells, $5 \leq l_{\max} \leq 9$, and depends on the interface orientation and shape of the irregular cell. Equation (30) is solved using the standard line-SOR procedure, with alternate sweeps in the i and j directions and a standard Thomas algorithm for the solution of the resulting tridiagonal matrix. The use of a Cartesian grid allows for the use of these fast solution procedures.

From the above discussion of the discretization strategy for interfacial cells, it is clear that the discretization can be constructed based on the geometry of each cell and depends on the (x, y) locations of the stencil-points chosen to represent the local linear-quadratic interpolant in the interfacial cells. In terms of implementation of the stencil-point choice and calculation of the weights α, β, a systematic procedure can be developed in a fairly straightforward way once the geometry of the cell is known. These procedures can also be extended to three dimensions. With all of these features, the current solver can, in principle, handle arbitrarily complex moving geometries. Furthermore, as demonstrated in [4], multiple immersed boundaries can be handled just as easily as a single boundary, because the control-volume reconfiguration is performed locally in the interfacial cells and does not affect the rest of the domain. Finally, since the inside of the immersed boundary is treated in the same manner as the outside, it is a straightforward matter to entertain arbitrarily large jumps in transport properties (without smoothing them) across the phase boundary or even to solve a different set of equations inside the immersed boundary. In this article we will use this feature to compute the diffusion of heat with discontinuities in transport coefficients across the solid–liquid interface.

5.2. Computing the Interface Velocity

The interface velocity is obtained as described in [3]. The procedure can be summarized with the aid of Figure 3*d*. The temperature gradient in each phase is obtained by the normal probe technique, where a normal from the interface marker is extended into each phase. The temperature values at two nodes on the normal, placed at distances equal to the local mesh spacing δx, are interpolated from the background mesh. The gradient is then obtained as

$$\frac{\partial T}{\partial n} = \frac{4T_{n1} - T_{n2} - 3T_{\text{int}}}{2\delta x} \tag{31}$$

where subscripts $n1$ and $n2$ imply evaluations of temperature at the two nodes on the normal probe and subscript "int" implies the value on the interface. Having

calculated the temperature gradients in each phase using Eq. (31), the interface velocities are computed at the markers using Eq. (11). These are then advected to new positions in order to evolve the interface in time.

Once the interface has moved to its new position, the interface markers are redistributed at uniform arclength spacing along the interface, maintaining a spacing $ds = O(dx)$, the local grid spacing. Points are added or deleted on the interface as necessary to maintain adequate interface resolution. The normal and curvature at the interfacial markers are computed as described in [3]. The curvature $\boldsymbol{\kappa}$ and orientation $\theta\,[= \tan^{-1}(ny/nx)]$ are then used in applying the boundary condition, via Eq. (9), in solving the governing equations.

5.3. Overall Solution Procedure

For curvature-driven growth problems, stability of the interface update requires an implicit coupled procedure for obtaining the field solution [23, 24] and the interface position simultaneously at time level t^{n+1}. In the absence of an implicit, coupled treatment of the field solution and interface evolution, the calculations can become very stiff. The stability restriction on an explicit scheme can be very severe $[\delta t = O(dx^3)]$, as demonstrated by Hou et al. [23].

An implicit procedure similar to that employed in [3] is used in the present work. The overall solution procedure with boundary motion is as follows:

0. Advance to next time step $t = t + \delta t$. Iteration counter $k = 0$.
1. Augment iteration counter. $k = k + 1$.
2. Determine the intersection of the immersed boundary with the Cartesian mesh.
3. Using this information, reshape the boundary cells.
4. For each reshaped boundary cell, compute and store the coefficients appearing in discrete form, Eq. (29).
5. Advance the discretized equations in time.
6. Advance the interface position in time.
7. Check whether the temperature field and interface have converged. Convergence is declared if $\max |T_{ij}^k - T_{ij}^{k-1}| \langle \varepsilon_T$ and $\max |X_{ip}^k - X_{ip}^{k-1}| \langle \varepsilon_I$, where k is the iteration number and ε is a convergence tolerance set to 10^{-5} in the calculations so that the solution obtained is independent of the convergence criterion.
8. If not converged, go to 1 for next iteration. If converged, go to step 0 for next time step.

Typically, less that five iterations are required for convergence, since the previous time step solution provides an excellent guess to the solution at the current step. Note that with this implicit iterative approach stable computations of interface evolution can be performed with time step sizes that are controlled by a CFL-type criterion of the form $\delta t = \lambda \delta x / \max(V_{\text{interface}})$, where λ is set to 0.1 in the calculations performed.

5.4. Computational Setup

The numerical method described above is now used in computations of dendritic growth and is shown to provide physically correct solutions for dendritic

growth of crystals. It will be shown that the effects of all the control parameters, namely undercooling (Δ), anisotropy (ε), and property jumps (α_s/α_l) across the solid–liquid front are correctly captured by the algorithm. The results are compared quantitatively with solvability theory as well as with other sharp interface computations.

To start the computations, a circular seed crystal is placed at the origin of the computational domain of dimension $H \times H$. In calculations below, we adopt $H = 800$. The radius of the seed is 15. These dimensions are taken in order to compare with the results of Dantzig and co workers [10, 18, 19]. Due to the symmetry of the problem, computations are performed in the first quadrant only. Thus, the x and y axes are symmetry axes. A fine mesh is placed in an inner region, as shown in Figure 4, while the mesh is coarsened linearly from the edge of the fine-mesh region to the end of the domain. The appropriate fine-mesh spacing for each case was determined from grid-refinement studies to establish that the solutions are grid independent for the chosen mesh size. We have shown in earlier work [3] that the sharp interface approach gives solutions for dendritic growth that are independent of the mesh spacing and mesh orientation. In the following we compare the results of solutions that were deemed grid independent by performing a series of calculations on progressively refined grids until the steady-state results on two successive grid sizes were found to agree adequately with each other and with the solvability results.

The computations are performed starting from an initial condition supplied to be $T = 0$ in the solid seed and $T = \Delta$, the nondimensional undercooling value, elsewhere in the domain. Since this initial temperature field is discontinuous at the interface, the interface velocity in the initial stage of the calculations will be very

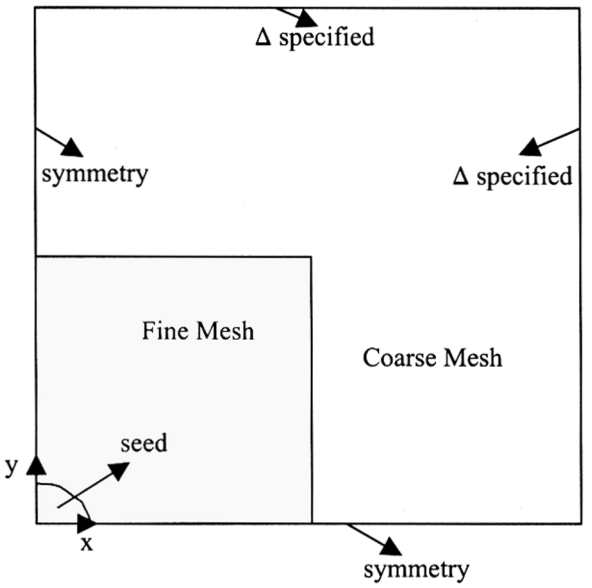

Figure 4. Schematic of the computational setup employed for the dendritic growth simulations.

large. To avoid problems with stability due to the large value of velocity, the seed is held in place (i.e., interface velocity is set to zero) for the first hundred time steps of the calculation. This allows a thermal boundary layer of the extent of a few mesh spacings to form around the seed. Thereafter the seed is allowed to grow with the velocity computed from Eq. (11). Although these initial conditions are somewhat arbitrary, the evolution of the dendrite to the desired theoretical tip shape appears to proceed regardless of the initial condition, providing a strong vindication of the tip selection mechanism. Computations are carried out until the tip velocity, radius, and selection parameter have each hit steady-state values.

The time required to achieve steady-state depends on the parameters assigned for each case, and a typical trend is shown in Figure 5 for the case with $\Delta = 0.65$, $d_0 = 0.5$, and $\varepsilon = 0.05$. The tip velocity V_t, tip radius ρ, and selection parameter σ^* are shown in the figure with appropriate multipliers to fit all the curves on the same plot. The selection parameter appears to be established very quickly in the growth process. The velocity and tip curvature then adjust in time until they approach steady-state values. This was found to be an interesting common aspect in the computations of dendritic growth for the range of parameters explored in this study. It appears that the pattern selection mechanism that determines σ^* operates in the entire growth process, even in the earlier stages of the growth, while the radius and velocity of the tip are "driven" toward appropriate values to yield the established σ^*.

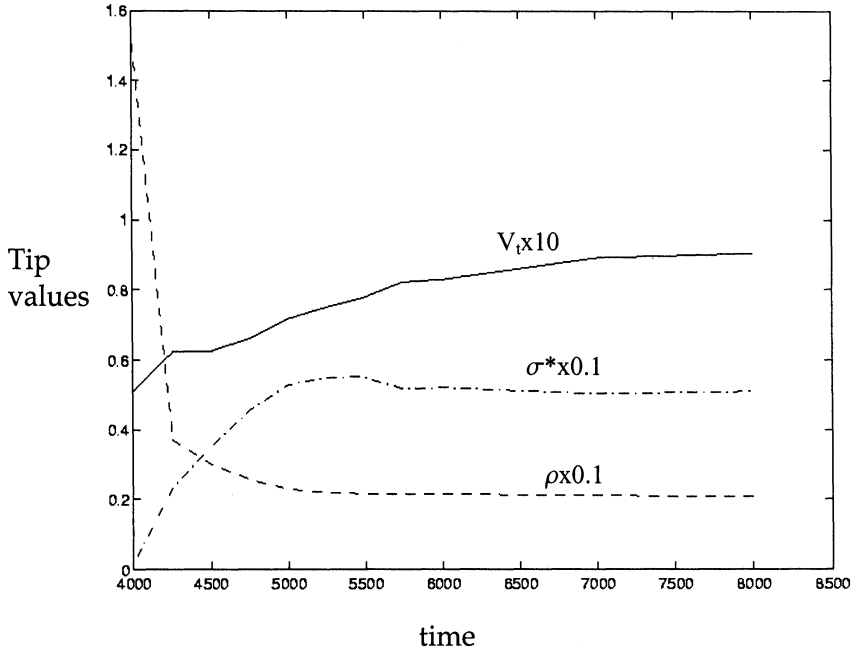

Figure 5. Progression of the tip values to steady state as the dendritic crystal grows. The tip velocity, radius, and selection parameter with indicated multipliers are shown in the figure, plotted against time. The undercooling $\Delta = 0.65$, capillarity parameter $d_0 = 0.5$, and anisotropy strength $\varepsilon = 0.05$.

6. COMPARISON OF NUMERICAL SOLUTIONS WITH SOLVABILITY THEORY

In this section we compare the results of our computations with the solvability theory results quoted in [8, 10, 18, 19]. The growth and material parameters are varied and the dendritic tip details such as radius, velocity, and selection parameter are each compared with theoretical and computational results presented in the above articles. Note that while Tong et al. [8] and Provatas et al. [19] have used the phase-field approach where the interface is diffuse, Kim et al. [10] have used a sharp interface approach based on level-set tracking in order to accurately capture discontinuities in material properties.

6.1. Effect of Undercooling

We first study the effect of growing the dendritic crystal from the seed in the arrangement shown in Figure 4. Two values of undercooling, namely, 0.55 and 0.65, are studied by maintaining the values of capillarity parameter $d_0 = 0.5$ and anisotropy strength $\varepsilon = 0.05$. Figure 6*a* shows the variation of the computed tip velocity

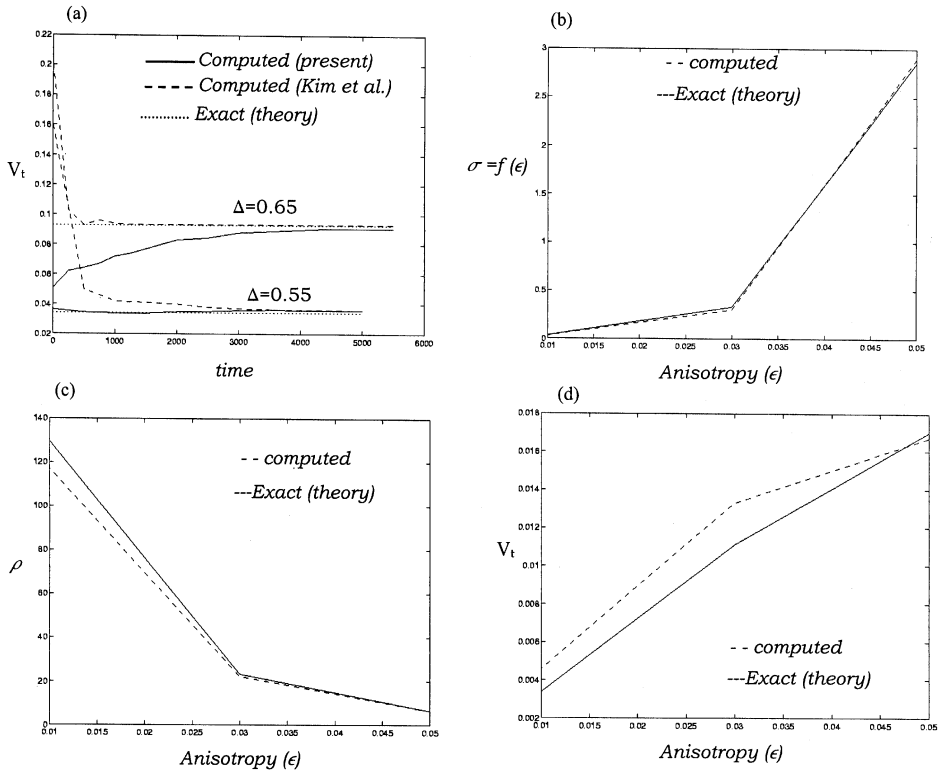

Figure 6. (*a*) Effect of undercooling. The tip velocity for two different undercoolings is compared with results of Kim et al. 2001 and with microscopic solvability theory. (*b*) Tip selection parameter versus anisotropy. (*c*) Tip radius as a function of crystalline anisotropy. (*d*) Tip velocity as a function of crystalline anisotropy.

with time for the dendrites grown at the above two different undercoolings. For these calculations, the grid spacing in the fine-grid region was $\delta x = 0.4$. The fine mesh occupied a region in the computational domain of 200×200 units from the origin. Therefore the mesh size in the fine region was 500×500 mesh cells. The stretching in the coarse region was such that 100 mesh points were used in the coarse region using a linear stretching of the mesh with stretch factor of 1.2. The curves corresponding to each case are indicated in Figure 6*a* along with the results from [10] and the two-dimensional solvability results. Although the initial conditions in our case are different from that in [10], the match in the computed steady-state tip velocity with solvability theory is comparable with that obtained by Kim et al. [10]. Thus, in the present calculations, the dendrite tip velocity is very different from that of [10] in the initial transient. However, in time, the value of the velocity approaches the theoretically predicted one. This behavior is seen for both values of undercooling tested.

6.2. Effect of Anisotropy

Crystalline anisotropy plays a crucial role in selecting the dendritic tip characteristics as indicated by the selection parameter, Eq. (7). Figures 6*b–d* present the computed variation of steady-state tip values at an undercooling of $\Delta = 0.55$ plotted against anisotropy parameter ε. We computed dendritic growth from the seed for this undercooling for three different values of anisotropy, i.e., $\varepsilon = 0.01, 0.03$, and 0.05. These values were chosen to correspond to results from phase-field calculations and results from two-dimensional solvability theory presented in [8]. The current numerical method is seen to adequately compute the tip selection parameter σ^* predicted by solvability theory. This is shown in Figure 6*b*. In Figures 6*c* and 6*d* we show plots of dendrite tip radius and velocity plotted against the anisotropy. As seen in these figures, the computed selection parameter is in excellent agreement with the theoretically predicted value for the range of anisotropies computed. There are some discrepancies in the velocity and radius values, particularly at the lower anisotropy. This is because the sensitivity of the calculations is extremely high at low anisotropies and it is likely that grid-induced anisotropy may be causing the slight deviation of the tip geometry from the theoretical values. Despite this, the selection parameter is accurately computed. These results indicate again that the selection parameter is a robust value that appears to be established to a high degree of accuracy and very early in the dendrite growth process. In Figures 7*a* and 7*b* we show the evolution of the dendrite tip shape in time for the case of $\varepsilon = 0.01$ and 0.05 for an undercooling value of $\Delta = 0.55$. As expected, the tip shape for the higher anisotropy case is seen to be much sharper than for the $\varepsilon = 0.01$ case.

In Figures 7*c* and 7*d* we show the isotherms in the solid and liquid phases for the steady-state dendrites obtained in the cases corresponding to Figures 7*a* and 7*b*. As can be clearly observed from the contours of temperature, the discontinuity in the temperature gradients at the solid–liquid interface is clearly captured by the present sharp interface method. Thus, the singular, moving source of latent heat residing on the phase boundary is correctly captured as a delta function, without smearing over a finite region of the mesh.

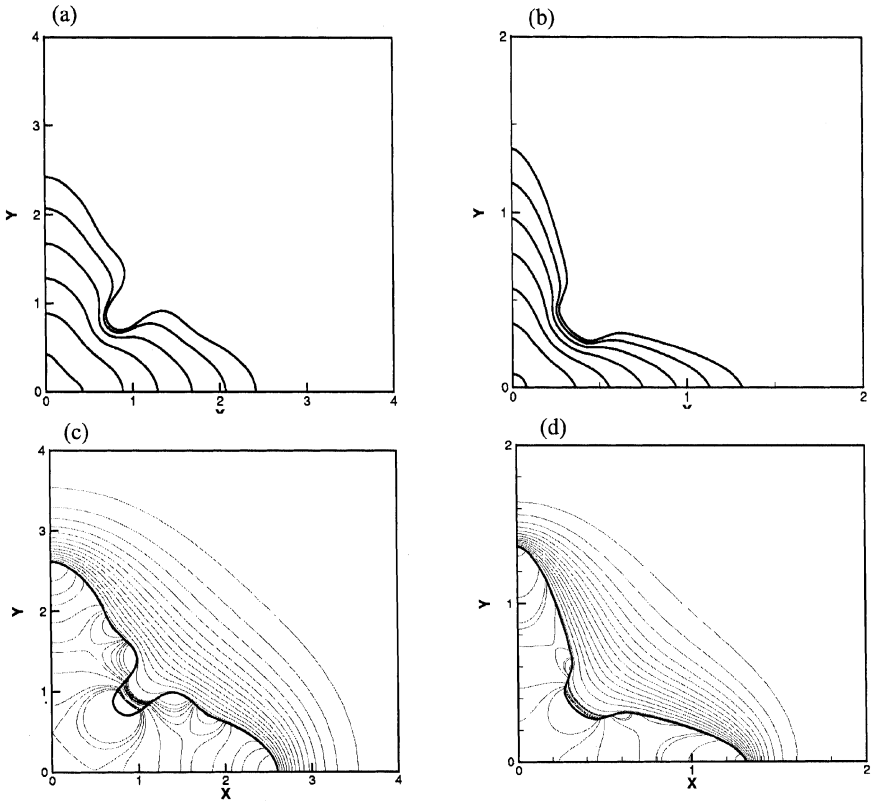

Figure 7. Evolution of interface shape and isotherms for undercooling $\Delta = 0.55$, capillarity parameter $d_0 = 0.5$. (*a*) Interface shape for low anisotropy, $\varepsilon = 0.01$. (*b*) Interface shape for high anisotropy, $\varepsilon = 0.05$. (*c*) Isotherms corresponding to the steady-state shape in (*a*). (*d*) Isotherms corresponding to the steady-state shape in (*b*).

6.3. Effect of Property Jumps

In order to clearly demonstrate the value of the sharp interface approach, we present in Figure 8, the computed results for the selection parameter for the case with parameters $\Delta = 0.65$, $d_0 = 0.5$ and $\varepsilon = 0.05$. The thermal diffusivity ratio α_s/α_l between the solid and liquid is varied and the selection parameter is computed and plotted against this ratio. The results are compared with the computational results of [10] and with the prediction of linearized solvability theory [25], which indicates that, as a function of the thermal diffusivity ratio, the tip radius and velocity are related as follows:

$$\rho^2 V_t \approx \frac{1}{2}\left(1 + \frac{\alpha_s}{\alpha_l}\right)(\rho^2 V_t)_{\alpha_s/\alpha_l = 1} \tag{32}$$

Our computed values for the product $\rho^2 V_t$ [related to the inverse of the selection parameter σ^*, see Eq. (5)] are shown in Figure 8*a*, plotted against the ratio of diffusivities, α_s/α_l. The results are in close agreement with those in [10] and also the

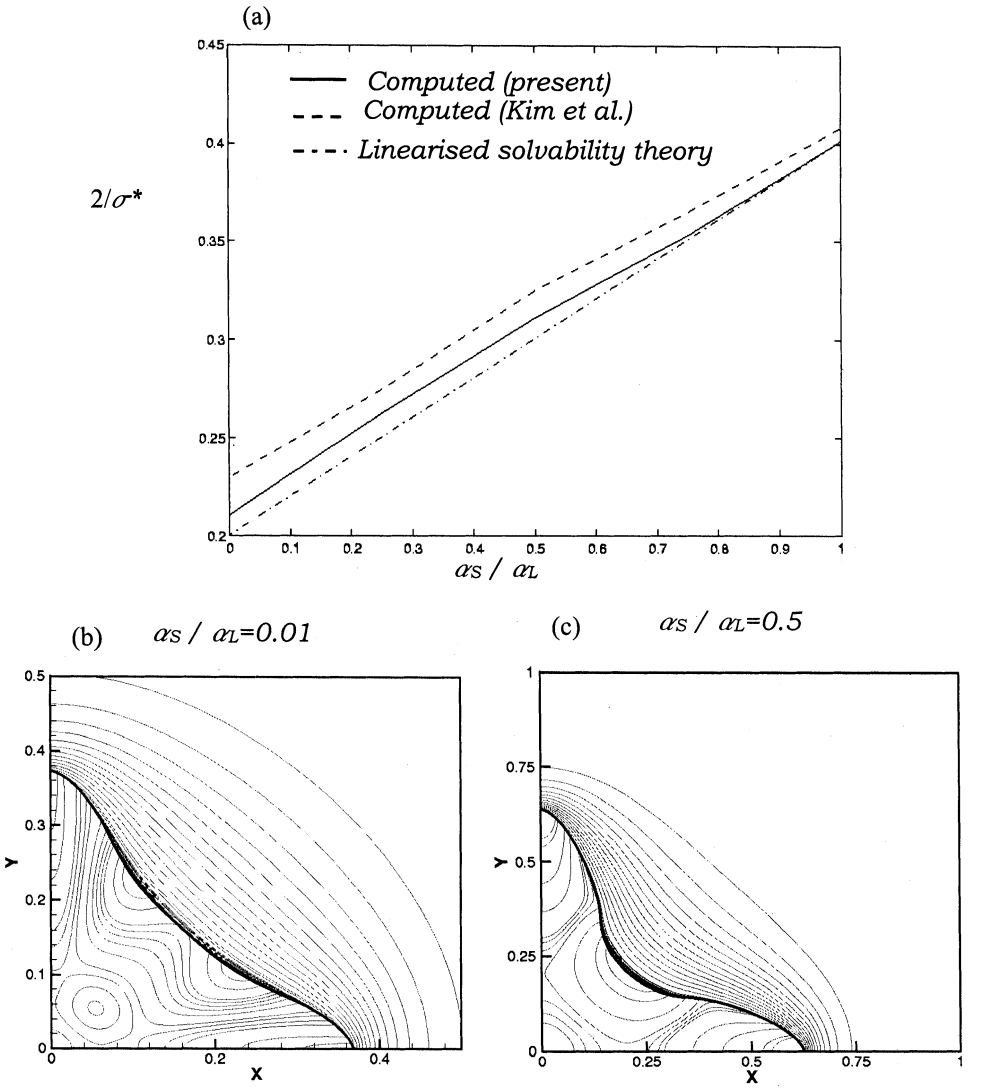

Figure 8. Effect of discontinuous material property on dendritic growth. (*a*) Plot of $\rho^2 V_t = 2/\sigma^*$ against the ratio of diffusivities α_s/α_l. (*b*) Interface shape at steady-state tip condition and isotherms in the two phases for the large diffusivity jump case $\alpha_s/\alpha_l = 0.01$. (*c*) Interface shape and isotherms at steady-state tip condition for $\alpha_s/\alpha_l = 0.5$.

predicted variation is fairly close in slope to the theoretically predicted value according to Eq. (32). In Figures 8*b* and 8*c*, we show the isotherms in the solid and liquid at steady state for the cases where the ratio $\alpha_s/\alpha_l = 0.01$ and 0.5, respectively. The subtle changes in the temperature field in the solid and the resulting changes in the shape of the growing dendrite are clearly seen in the figure. These subtle variations in temperature are responsible for the variation in the selection parameter for discontinuous material properties, as shown in Figure 8*a* and given by Eq. (32).

In the case of very low thermal diffusivity in the solid, as in Figure 8*b*, the isotherms in the solid are nearly normal to the solid–liquid interface, while for the higher value of solid diffusivity the isotherms are nearly parallel to the interface. Note that the ability of the current methodology to treat discontinuities in both the values (of diffusivities) and gradients (of temperature) at the boundaries has been demonstrated in Figures 8*b* and 8*c* respectively. This has proven useful in computing the solidification of materials with impurities (solutes) in ongoing work on freezing of aqueous solutions. In such problems, there is a vast disparity in the species diffusivities across the interface (solid-to-liquid values in the ratio of 10^{-4}–10^{-5}). In order to accurately capture solute microsegregation, the discontinuity in the material properties has to be respected. The sharp interface approach has been demonstrated in the above test case to be capable of producing physically correct solutions in the presence of such discontinuities.

7. SUMMARY

We have developed a numerical method for the computation of dendritic crystal growth where the solid–liquid interface is treated as a sharp front. Freeing the mesh from conforming to the complex evolving boundaries while maintaining a sharp solid–liquid interface allows for the accurate solution of phase change. The sharp interface nature and the second-order spatial and temporal discretization coupled with a conservative-finite volume scheme allows us to obtain solutions for dendritic crystal growth in good agreement with instability theory. We have demonstrated in the results presented in this article that the effects of undercooling, anisotropy and material property variations are each computed correctly by the method. The uniquely selected features of the dendrite tip, namely, tip radius and tip velocity, and the numerically obtained selection parameter are found to be in agreement with two-dimensional solvability theory. We have also shown the attractiveness of the method in situations where the diffusivity varies discontinuously across the front. Here again, good agreement is shown with theory as well as with other sharp interface calculations. The ability of the method to treat material property jumps as discontinuous is important in the simulation of solidification in alloys and solutions, where the species transport equation comes into play. Solute segregation from the solid into the melt/solution is a critical part of such processes. The large jump in species diffusivity from solid to liquid phase is best handled by treating each phase separately, a capability that is provided by the present sharp interface technique. Ongoing work involves solution of solidification phenomena in the presence of impurities and study of the effect of convection on the microstructure.

REFERENCES

1. W. W. Mullins and R. F. Sekerka, Stability of a Planar interface during Solidification of a Dilute Binary Alloy, *J. Appl. Phys.*, vol. 35, no. 2, pp. 444–451, 1964.
2. D. A. Kessler, J. Koplik, and H. Levine, Pattern Selection in Fingered Growth Phenomena, *Adv. Physics*, vol. 37, no. 3, pp. 255–339, 1988.

3. H. S. Udaykumar, R. Mittal, and W. Shyy, Solid-Liquid Phase Front Computations in the Sharp Interface Limit on Fixed Grids, *J. Comput. Phys.*, vol. 153, pp. 535–574, 1999.
4. T. Ye, R. Mittal, H. S. Udaykumar, and W. Shyy, An Accurate Cartesian Grid Method for Viscous Incompressible Flows with Complex Immersed Boundaries, *J. Comput. Phys.*, vol. 156, pp. 209–240, 1999.
5. H. S. Udaykumar, R. Mittal, P. Rampunggoon, and A. Khanna, A Sharp Interface Cartesian Grid Method for Simulating Flows with Complex Moving Boundaries, *J. Comput. Phys.*, vol. 174, pp. 1–36, 2001.
6. G. Caginalp and P. Fife, Phase-Field Methods for Interfacial Boundaries, *Phys. Rev. B*, vol. 33, pp. 7792–7794, 1986.
7. G. B. McFadden, A. A. Wheeler, R. J. Braun, and S. R. Coriell, Phase-Field Models for Anisotropic Interfaces, *Phys. Rev. E*, vol. 48, no. 3, pp. 2016–2024, 1993.
8. X. Tong, C. Beckermann, and A. Karma, Velocity and Shape Selection of Dendritic Crystals in a Forced Flow, *Phys. Rev. E*, vol. 61, pp. R49–R52, 2000.
9. T. Y. Hou, Z. Li, S. Osher, and H. Zhao, A Hybrid Method for Moving Interface Problems with Application to the Hele-Shaw flow, *J. Comput. Phys.*, vol. 134, no. 2, pp. 236–247, 1997.
10. Y.-T. Kim, N. Provatas, N. Goldenfeld, and J. Dantzig, Universal Dynamics of Phase-Field Models for Dendritic Growth, *Phys. Rev. E*, vol. 59, no. 3, pp. R2546–R2549, 1999.
11. A. Schmidt, Computation of Three-Dimensional Dendrites with Finite Elements, *J. Comput. Phys.*, vol. 125, pp. 293–312, 1996.
12. J. M. Sullivan, D. R. Lynch, and K. O'Neill, Finite Element Simulations of Planar Instabilities during Solidification of an Undercooled Melt, *J. Comput. Phys.*, vol. 69, pp. 81–111, 1987.
13. D. Juric and G. Tryggvasson, A Front Tracking Method for Dendritic Solidification, *J. Comput. Phys.*, vol. 123, pp. 127–148, 1996.
14. A. Karma and W. J. Rappel, Phase-Field Simulation of Three-Dimensional Dendrites: Is Microscopic Solvability Theory Correct?, *J. Crystal Growth*, vol. 174, pp. 54–64, 1997.
15. A. Karma and W. J. Rappel, Phase Field Model of Dendritic Growth in Two and Three Dimensions, *Phys. Rev. E*, vol. 57, pp. 4323–4349, 1998.
16. D. M. Anderson, G. B. McFadden, and A. A. Wheeler, Diffuse Interface Methods in Fluid Mechanics, *Annu. Rev. Fluid Mech.*, vol. 30, pp. 139–165, 1998.
17. R. J. Leveque and Z. Li, The Immersed Interface Method for Elliptic Equations with Discontinuous Coefficients and Singular Sources, *SIAM J. Numer. Anal.* vol. 31, no. 4, pp. 1019–1044, 1994.
18. Y.-T. Kim, N. Provatas, N. Goldenfeld, and J. Dantzig, Computation of Dendritic Microstructure Using a Level-Set Method, *Phys. Rev. E.*, vol. 62, no. 2, pp. 2471–2474, 2000.
19. N. Provatas, N. Goldenfeld, and J. Dantzig, Efficient Computation of Dendritic Microstructures Using Adaptive Mesh Refinement, *Phys. Rev. Lett.*, vol. 80, no. 15, pp. 3308–3311, 1998.
20. H. S. Udaykumar, R. Mittal, and P. Rampunggon, Interface Tracking Finite Volume Method for Complex Solid-Fluid Interactions on Fixed Meshes, *Commun. Numer. Meth. Eng.*, in press, 2001.
21. G. P. Ivantsov, Temperature Field around Spherical, Cylindrical and Needle-Shaped Crystals Which Grow in Supercooled Melt, *Dokl. Akad. Nauk SSSR*, vol. 58, pp. 567–569, 1947.
22. S. C. Huang and M. E. Glicksman, Fundamentals of Dendritic Solidification—I and II, *Acta Metall.*, vol. 29, pp. 701–715 and 717–734, 1981.

23. T. Y. Hou, J. S. Lowengrub, and M. J. Shelley, Removing Stiffness from Interfacial Flows with Surface Tension, *J. Comput. Phys.*, vol. 114, p. 312, 1994.
24. C. Tu and C. S. Peskin, Stability and Instability in the Computation of Flows with Moving Immersed Boundaries: A Comparison of Three Methods, *SIAM J. Sci. Stat. Comput.*, vol. 13, p. 1361, 1992.
25. A. Barbieri and J. S. Langer, Predictions of Dendritic Growth Rates in the Linearized Solvability Theory, *Phys. Rev. A*, vol. 39, no. 10, pp. 5314–5325, 1989.

AUTHOR QUERY

1. Au: ok?
2. Au: update?