A level set simulation of dendritic solidification with combined features of front tracking and fixed domain methods

Lijian Tan, Nicholas Zabaras [1]

Materials Process Design and Control Laboratory, Sibley School of Mechanical and Aerospace Engineering, 188 Frank H.T. Rhodes Hall, Cornell University, Ithaca, NY 14853-3801, USA

Abstract

A method combining features of front tracking methods and fixed domain methods is presented to model dendritic solidification of pure materials. To explicitly track the interface growth and shape of the solidifying crystals, a front tracking approach based on the level set method is implemented. To easily model the heat and momentum transport, a fixed domain method is implemented assuming a diffused freezing front where the liquid fraction is defined in terms of the level set function. The fixed domain approach, by avoiding the explicit application of essential boundary conditions on the freezing front, leads to an energy conserving methodology that is not sensitive to the mesh size. To compute the freezing front morphology, an extended Stefan condition is considered. Applications to several classical Stefan problems and two- and three-dimensional crystal growth of pure materials in an undercooled melt including the effects of melt flow are considered. The computed results agree very well with available analytical solutions as well as with results obtained using front-tracking techniques and the phase field method.

Key words: Level set method; Crystal growth; Dendritic solidification; Front tracking methods; Fixed domain methods; Fluid flow

1 Introduction

Simulation of solidification and crystal growth processes has been of interest for a number of years not only because of its scientific value in understanding

[1] Corresponding author: Fax: 607-255-1222 Email: zabaras@cornell.edu

pattern formation in nature but also because of its importance in many technological applications. The classical Stefan problem with a sharp interface is well accepted for modelling the solidification of pure materials. Front-tracking techniques have been widely used for such sharp front solidification processes. Recently these techniques have been used successfully to reproduce the complex dendritic structure in crystal growth in undercooled melts including effects such as liquid trapping, tip-splitting, side branching and coarsening [1–3] Successful three-dimensional front-tracking implementations including the effects of melt flow have been presented [4]. The advantages of front-tracking methodologies lie in their ability to directly enforce the freezing interface temperature (Gibbs-Thomson) relation and energy balance (Stefan condition). Unfortunately, many of the current implementations of these conditions do not allow global energy conservation even though they may satisfy the Stefan condition pointwise. In addition, such formulations appear to result in high sensitivity to mesh size and orientation.

During the last two decades, significant progress has been made in the simulation of microstructure evolution in solidification processes using phase-field methods [5–7]. These approaches, by considering a diffused-interface and a fixed-grid, avoid the need for applying temperature boundary conditions on the moving interface. A review of recent progress in phase field methods as applied to solidification processes is given in [5]. The basic idea of the phase-field method is to employ a phase-field variable Φ that varies smoothly from zero to unity between the two phases over the diffused-interface region, which has a small but numerically resolvable thickness. The phase field variable serves to distribute the interfacial forces over the diffused freezing region. It is governed by a phase-field control equation derived from the thermodynamics of phase transition [5]. Important physical mechanisms, such as curvature, anisotropy and kinetics effects, are implicitly incorporated in the phase-field control equation. This leads to many computational advantages. For example, one does not need to compute interfacial geometric quantities such as interface curvature and outward normal vector.

The phase field method can be shown to reduce to the standard sharp interface formulation in the limit of vanishing interface thickness [8]. The quality of the solution deteriorates with increasing interface thickness. This necessitates the grid spacing to be of the order of or smaller than the interfacial thickness. It has been shown that the interface thickness must be smaller than the capillary length for the solution to converge to the sharp interface limit [9,10]. Improved asymptotic coefficients have been derived for the thin-interface limit of the phase field equations which facilitate the use of a coarser grid [11]. Phase-field analysis for unequal solid/liquid diffusivities leads to computational models which require finer grid resolution and hence lead to slower computational performance [12]. One of the drawbacks of the phase field method is the significant computational effort required, especially when

investigating dendritic growth in the presence of convection and multiple array dendritic growth. However, the techniques of adaptive gridding and parallel computing have resulted in assuaging this drawback to some extent [13,14]. Another drawback of the phase field methods is related to the large number of parameters involved in the solution of the evolution equations. These parameters are difficult to determine for accurate physical crystal growth simulation of real world materials.

The level set method is an alternative method to handle the sharp interface front directly and to avoid the asymptotic analysis needed in phase field models [15]. It has been shown to be a promising mathematical tool for tracking the interface with low computational cost. It is widely used in various applications such as two-phase flow, crack propagation, computer vision and image processing. In this method, interfacial geometric quantities such as curvature and outward normal can be easily calculated using the level set variable ϕ. The method was first applied to Stefan problems in [16]. Level set calculations for dendritic growth were reported in [17]. In these works [16,17], the heat flux at nodes near the interface is interpolated in order to calculate the interface velocity via the Stefan condition. During this interpolation, the temperature on the interface is fixed at the equilibrium temperature defined from the Gibbs-Thomson relation. However, like front-tracking methods, the direct application of temperature boundary conditions on the interface and the computation of heat fluxes from the temperature nodal values usually lead to energy conservation issues associated with the discretization error [18]. This may result in large variation of the computed crystal shapes if meshes of different sizes and orientations are used.

The method presented here combines features of front-tracking and fixed domain methods. The level set method is implemented to allow a direct calculation of the growth and shape of the solidifying crystals. An energy conserving implementation of the level set method is used without the need to apply the temperature boundary condition explicitly at the freezing interface. To easily model the heat and momentum transport, a fixed domain method is implemented assuming a diffused freezing front where the liquid fraction is defined in terms of the level set function. The fixed domain approach, by avoiding the need to explicitly apply essential boundary conditions on the freezing front, leads to an energy conserving methodology that is not sensitive to the mesh size or orientation. The present implementation is similar to that of the phase field method, except that the 'phase field variable' Φ is no longer governed by the phase-field control equation but is defined using the level set (signed distance) variable ϕ. The freezing front morphology is computed as a post-processing operation using an extended Stefan condition.

This work provides a new approach for modelling dendritic solidification using a fixed finite element grid. It avoids the need for adaptive or moving

griding while at the same time provides an explicit and accurate tracking of the interface front. It allows a rather simple integration of level set techniques with stabilized volume-averaged based finite element techniques for thermal and momentum transport. The interface conditions are accurately enforced on an implicit manner while maintaining energy conservation for the overall system. The melt flow is modelled using equal-order velocity-pressure interpolation that has been shown to lead to better convergence rates and accuracy [19] than the fractional step method commonly used in dendritic solidification models. The present methodology will be shown to be computationally efficient and accurate for both two- and three-dimensional problems. We will demonstrate that the methodology performs very well in benchmark dendritic growth problems including growth under low-undercooling conditions for which other numerical approaches do not work [20].

The plan of the paper is as follows. Section 2 introduces the sharp interface model including the effects of fluid flow. Section 3 discusses a diffused-interface approximation of the Stefan problem. The level set formulation is briefly reviewed in Section 4. Section 5 provides details on the implementation of the method including modelling of the underlying transport phenomena. The overall methodology of the energy conserving scheme is discussed in Section 6. Section 7 presents several examples and compares their results with those obtained from analytical and other numerical approaches. Finally, Section 8 provides a summary of the work.

2 Definition of the Stefan problem

Let us consider the classical Stefan problem with a sharp interface including the effects of melt flow. The liquid phase of a pure material at initial temperature $T_0^\ell(x, y)$ is assumed to occupy a region Ω_0^ℓ. This liquid phase is either alone ($\Omega_0^\ell = \Omega$) or in an unstable equilibrium with the neighboring solid phase Ω_0^s ($\Omega = \Omega_0^\ell \bigcup \Omega_0^s$) at initial temperature $T_0^s(x, y)$. In the latter case, the regions Ω_0^ℓ and Ω_0^s are separated by the solid/liquid interface Γ_0. The superscripts ℓ and s are used here to denote quantities corresponding to the liquid and solid phases, respectively, whereas the subscript 0 is used to denote quantities at time $t = 0$. We assume that solidification starts at time $t = 0$. The domains Ω^ℓ and Ω^s are time-dependent and the solid-liquid interface Γ is moving with normal velocity V. The normal $\boldsymbol{n}$ to the interface Γ is defined as pointing away from the solid region Ω^s. The domain $\Omega = \Omega^\ell \bigcup \Omega^s$ containing these two phases and its external boundary $\partial\Omega$ are assumed constant (time-independent). Figure 1 presents a typical schematic of the problems considered.

We assume constant thermo-physical and transport properties, including viscosity μ, density ρ, thermal conductivity k, heat capacity c and latent heat L.

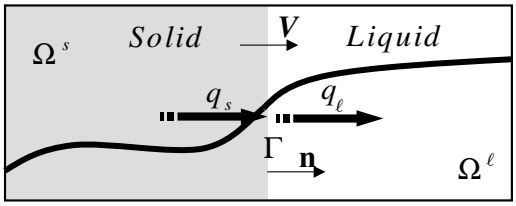


Fig. 1. Schematic of the sharp interface model of the solidification of a pure material (Stefan problem). The heat fluxes q_s and q_l are defined on the freezing front.

The melt flow is assumed to be a laminar flow caused by temperature-induced density variations (Boussinesq flow). Following standard notation, the governing equations in the presence of fluid flow are given as follows:

$$\rho_s c_s \frac{\partial T(\boldsymbol{x},t)}{\partial t} = k_s \nabla^2 T(\boldsymbol{x},t), \quad \boldsymbol{x} \in \Omega^s, \tag{1}$$

$$\rho_l c_l \left(\frac{\partial T(\boldsymbol{x},t)}{\partial t} + \boldsymbol{v} \cdot \nabla T(\boldsymbol{x},t) \right) = k_l \nabla^2 T(\boldsymbol{x},t), \quad \boldsymbol{x} \in \Omega^\ell, \tag{2}$$

$$\rho_l \left(\frac{\partial \boldsymbol{v}}{\partial t} + \nabla \boldsymbol{v}(\boldsymbol{x},t) \boldsymbol{v}(\boldsymbol{x},t) \right) = -\nabla p(\boldsymbol{x},t) \boldsymbol{I} + \nabla \cdot \mu \left[\nabla \boldsymbol{v}(\boldsymbol{x},t) + (\nabla \boldsymbol{v}(\boldsymbol{x},t))^{\mathrm{T}} \right] + \boldsymbol{b}, \quad \boldsymbol{x} \in \Omega^\ell, \tag{3}$$

$$\nabla \cdot \boldsymbol{v}(\boldsymbol{x},t) = 0, \quad \boldsymbol{x} \in \Omega^\ell, \tag{4}$$

where $\boldsymbol{v}$ is the melt flow velocity and $\boldsymbol{b}$ is the buoyancy body force.

The temperature on the interface Γ denoted as T_I is equal to the equilibrium temperature T^*. This equilibrium temperature is given from the Gibbs-Thomson relation as follows:

$$T^* = T_m + \varepsilon_c \kappa + \varepsilon_V V, \tag{5}$$

where T_m is the pure material melting temperature, κ is the curvature of the freezing interface, and ε_c and ε_V are the curvature and kinetic undercooling coefficients, respectively.

The motion of the interface velocity V is dictated by the classical Stefan equation (energy balance at the freezing front) as follows:

$$\rho_s L V = [q] = q_s - q_l, \tag{6}$$

where [q] denotes the jump of the heat flux $q = k \nabla T \cdot \boldsymbol{n}$ at the freezing front.

To describe the interface, we construct a field ϕ such that at any time t, the

interface is equal to the zero level set of ϕ, i.e.

$$\Gamma(t) = \{\boldsymbol{x} \in \Omega \ : \ \phi(\boldsymbol{x}, t) = 0\}\,. \tag{7}$$

Initially, ϕ is set equal to the signed distance function from the interface Γ_0,

$$\phi(\boldsymbol{x}, 0) = \begin{cases} +d(\boldsymbol{x}, 0), & \boldsymbol{x} \in \Omega_0^\ell, \\ 0, & \boldsymbol{x} \in \Gamma_0, \\ -d(\boldsymbol{x}, 0), & \boldsymbol{x} \in \Omega_0^s, \end{cases} \tag{8}$$

where $d(\boldsymbol{x}, 0)$ is the normal distance of a point $\boldsymbol{x}$ from the interface.

The idea behind the level set method is to move ϕ with the correct speed V at the interface which is extracted from Eq. (6). The interface position is thus implicitly stored in ϕ. The equation of motion governing ϕ is given as follows:

$$\phi_t + V|\nabla\phi| = 0. \tag{9}$$

This equation moves ϕ with correct speed at the interface so that Γ will always be equal to the zero level set of ϕ [15].

With the above introduction of the level set function, we will re-write the Stefan condition for the classical Stefan problem as follows:

$$\rho_s L V = [q], \quad q_s = \lim_{\phi \to 0^-} q, \quad q_l = \lim_{\phi \to 0^+} q, \tag{10}$$

where the notation 0^+ and 0^- is used here for notational simplification to denote the values of ϕ as we approach the freezing front from the solid and liquid sides, respectively.

Equations (1)-(5) and (10) together with appropriate initial temperature and velocity conditions, boundary thermal conditions on Γ and the no-slip condition in all solid boundaries completely define what we here refer to as the *Stefan problem*. Note that in the literature the classical Stefan problem does not include melt flow. These effects have been included herein to allow us to study their importance in crystal growth of pure materials in undercooled melts.

3 Introducing the extended Stefan problem

Many of the difficulties in the implementation of the Stefan problem defined earlier are related to the application of the essential boundary condition given in Eq. (5). Front-tracking techniques attempt to apply this condition directly to the moving front, often leading to schemes that are not energy conserving. On the other hand, phase field methods consider a diffused-interface model in an attempt to avoid a direct application of this condition. However, phase field models require proper parameter selection through an asymptotic analysis in order to model the sharp interface solidification problem.

To take advantage of the front-tracking capability of the level set methods and of the ability of phase field methods to avoid directly applying Eq. (5), we introduce in this section the *extended Stefan problem* that uses features of both methods. Figure 2 shows a schematic of the new problem.

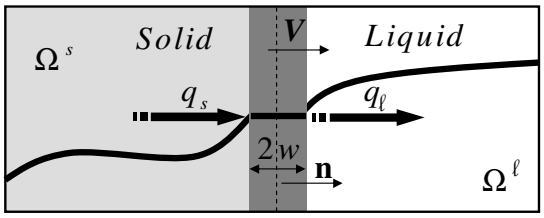


Fig. 2. Schematic of the diffused-interface model of the solidification of a pure material (extended Stefan problem). The heat fluxes q_s and q_l are here defined at distances $\pm w$ from the actual sharp freezing front.

Assumption 1: We assume that solidification occurs in a diffused zone of width $2w$ that is symmetric around $\phi = 0$ (see Fig. 2). The half-width w is not related with the underlying physics of the solidification of a pure material (which usually happens on a thickness of the order of atomic distances). In this work, the half-width w is selected based on the size of the grid used in the discretization of the problem.

A diffused-interface model will be used in the solution of the energy and momentum equations. Following ideas similar to those in diffused-interface models, let us define the function $\Phi(\boldsymbol{x}, t)$ as follows:

$$\Phi(\boldsymbol{x},t) = \begin{cases} 1, & \phi(\boldsymbol{x},t) > w, \\ 0, & \phi(\boldsymbol{x},t) < -w, \\ \phi(\boldsymbol{x},t)/_{2w} + 0.5, & \phi(\boldsymbol{x},t) \in [-w, w]. \end{cases} \tag{11}$$

Since solidification occurs in a diffused-interface, following a volume-averaging approach as in [19], we can write the energy equation (applicable in the whole domain Ω) as follows:

$$[\rho_l \Phi c_l + \rho_s(1-\Phi)c_s]\frac{\partial T}{\partial t} + \rho_l c_l \boldsymbol{v}\cdot\nabla T = \nabla\cdot(k\nabla T) - \rho_s[L + (c_s - c_l)(T - T_m)]\dot{\Phi}, \quad (12)$$

where Φ plays the role of the phase field variable in phase field models and $k \equiv k_l\Phi + k_s(1-\Phi)$. Similarly, the volume-averaging momentum and continuity equations take the following forms:

$$\frac{\partial(\rho\boldsymbol{v})}{\partial t} + \nabla\cdot(\frac{\rho^2}{\rho_l}\frac{\boldsymbol{v}\boldsymbol{v}}{\Phi}) = -\nabla p + \frac{p}{\Phi}\nabla\Phi + \nabla\cdot[\mu(\nabla(\frac{\rho}{\rho_l}\boldsymbol{v}) + \nabla^T(\frac{\rho}{\rho_l}\boldsymbol{v}))] - \mu\frac{(1-\Phi)^2}{\Phi^2}\frac{\rho}{\rho_l}\frac{\boldsymbol{v}}{K_0} + \Phi\rho_l g\boldsymbol{e_g}, \quad (13)$$

$$\nabla\cdot\boldsymbol{v}(\boldsymbol{x},t) = 0, \quad (14)$$

where $\rho \equiv \rho_l\Phi + \rho_s(1-\Phi)$, $\boldsymbol{e_g}$ is the unit vector in the direction of gravity, g is the gravity constant and $\boldsymbol{v}$ from now on denotes the volume-averaged velocity equal to $\Phi\boldsymbol{v}_l$ [19]. For simplicity of the model, we assume that the solid/liquid densities and specific heats are the same but we allow for different conductivities.

The Kozeny-Carman approximation for the permeability $K(\Phi)$ has been used directly in Eq. (13) with K_0 denoting the permeability constant. Note that the governing conservation Eqs. (12), (13) and (14) are applicable in the whole domain Ω.

The transport equations for the extended Stefan problem are now defined from Eqs. (12)-(14) together with appropriate initial temperature and velocity conditions, boundary thermal conditions on Γ and the no-slip condition. Note that these equations completely define the temperature and melt flow, but since they are based on a volume-averaging formulation, they do not directly involve the motion of the interface zone. The freezing front motion will be computed by the level set function as it will be discussed below.

To compute the front velocity V (velocity of the interface $\phi = 0$) we cannot simply utilize Eq. (10), which is based on the sharp front model. Instead, a modified energy balance is needed at the freezing zone as follows (see Fig. 2):

$$2\rho_s\tilde{c}w\frac{\Delta\bar{T}_I}{\Delta t} = \rho_s LV + (q_l - q_s), \quad q_s = \lim_{\phi\to -w^-} q, \quad q_l = \lim_{\phi\to w^+} q, \quad (15)$$

where $\bar{T}_I$ is the average temperature within the diffused-interface, and $\Delta\bar{T}_I$ is the increase of $\bar{T}_I$ at the time interval Δt. Also,

$$\tilde{c} \equiv 0.5(1 + \frac{\rho_l c_l}{\rho_s c_s})c_s, \quad (16)$$

and $w^+, -w^-$ are defined following similar notation to that introduced earlier for Eq. (10). The term $2\rho_s\tilde{c}w\Delta\bar{T}_I$ on the left hand side of Eq. (15) is the energy change in the diffused-interface due to change of temperature. Also note that the temperature on the zero level set denoted as T_I is conceptually different from $\bar{T}_I$, which is the average temperature within the diffused mushy zone. Since the temperature varies only slightly within the thin interface, in this work we will use T_I to approximate $\bar{T}_I$. The temperature T_I can be easily computed from interpolating temperature to zero level set. However, in order to maintain the generality of the formulation, we will maintain the notation $\bar{T}_I$ for the average temperature within the diffused-interface.

The Gibbs-Thomson constraint $T_I = T^*$ with T^* given in Eq. (5) ensures that the interface temperature is the equilibrium temperature at each instant. To explicitly enforce this constraint, various methods like a penalty method or a Lagrange multiplier approach can be used. In this work, we take an alternative approach. Since the numerical simulation provides solutions only at discrete time levels, we ensure that the interface temperature approaches the equilibrium temperature at these time levels.

Assumption 2: The mean interface temperature $\bar{T}_I$ in the freezing zone of width $2w$ is allowed to vary from the equilibrium temperature T^* in such a way that

$$\frac{\mathrm{d}\bar{T}_I}{dt} = -k_N(\bar{T}_I - T^*), \tag{17}$$

where k_N controls the rate with which $\bar{T}_I$ is designed to approach the desired equilibrium temperature.

Assuming for example that the equilibrium temperature remains constant, integration of Eq. (17) leads to the following:

$$\bar{T}_I(t) - T^* = (\bar{T}_I(0) - T^*)\exp(-k_N t). \tag{18}$$

A relaxation time that controls how fast $\bar{T}_I$ approaches T^* can then be defined as $\tau = \frac{1}{k_N}$. The relaxation parameter τ (or equivalently k_N) is selected such that the interface reaches the desired equilibrium temperature exponentially fast. An explicit integration of Eq. (17) over the time step Δt results in the following:

$$\bar{T}_I(t_n) - T^* \approx (\bar{T}_I(t_{n-1}) - T^*)(1 - k_N\Delta t), \tag{19}$$

where the subscripts refer to time levels ($n = 1, 2, \ldots$). To make sure that

the interface temperature stably converges to the equilibrium temperature, we require the following:

$$|1 - k_N \Delta t| < 1 \Rightarrow \Delta t < \frac{2}{k_N} = 2\tau. \tag{20}$$

If we further require that $\bar{T}_I - T^*$ does not change sign as it approaches zero ($T^* = \bar{T}_I$), then $\Delta t \leq \tau$. In our algorithms, for a given Δt, we select $k_N = \frac{1}{\Delta t}$ so that the condition $\Delta t \leq \tau$ is satisfied automatically. The selection of Δt will be discussed in Section 4.

Substitution of the constraint of Eq. (17) to Eq. (15) results in the following modified energy balance at the freezing front:

$$V = \frac{[q]}{\rho_s L} + \frac{2\tilde{c}w}{L} k_N (T^* - \bar{T}_I), \; q_s = \lim_{\phi \to -w^-} q, \quad q_l = \lim_{\phi \to w^+} q, \tag{21}$$

where the heat fluxes q_l and q_s are computed at the boundaries of the control volume. Note that in the above formulation, we use the average temperature $\bar{T}_I$ instead of the temperature T_I on the exact interface ($\phi = 0$). Equation (21) will be used as the energy balance at the diffused-interface and will allow us to accurately compute the interface velocity V. Note that the above scheme is only to ensure that the interface temperature relaxes to the equilibrium temperature in a given time step.

Remark 1: Note that Eq. (17) is not used directly in the analysis but it is embedded in the modified Stefan condition given in Eq. (21). To compute a finite front velocity when $k_N \to \infty$, Eq. (21) requires that $T_I = T^*$ in which case it is simplified to Eq. (10). It can be shown that, using Eq. (21), the numerical scheme for the thermal problem of Eq. (12) leads to a discretized form of Eq. (17) thus weakly enforcing $T_I = T^*$. The choices $w = \Delta x$ and $k_N = 1/\Delta t$ are sufficient to ensure satisfaction of both Eqs. (5) and (10) and lead to an accurate estimate of the front velocity V.

Remark 2: From a numerical point of view, the second term on the right hand side of Eq. (21) can be thought of as the constraint $T^* - T_I = 0$ numerically enforced via a Lagrange multiplier method. A graphical demonstration of an iterative process for the satisfaction of the constraint $T^* - T_I = 0$ is given in Fig. 3. In this figure, we assume that the temperature field away from the freezing zone remains the same during iterations. Figure 3 shows the iterative process as T_I approaches T^* from below. This iterative procedure for computing V was not needed in the calculations reported in Section 7 and one step calculation was sufficient to evaluate V within the desired accuracy.

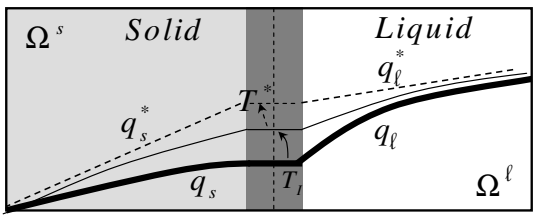


Fig. 3. Schematic of applying $T_I = T^*$ with the correction of Eq. (21).

Remark 3: The volume-averaging based energy Eq. (12) weakly accounts for the Stefan condition on the freezing front by treating its contribution as a source term. This is typically the case with phase field and other diffused-interface models. The key element of the present formulation is that the phase variable Φ (liquid volume fraction) is defined in terms of the level set function ϕ that is used to track the freezing front explicitly. In classical volume-averaging models of solidification, Φ is defined based on thermodynamic update formulas using the computed temperature and equilibrium temperature at each point within the diffused-interface. This last calculation is embedded in the calculation of ϕ.

Remark 4: Curvature or kinetic undercooling effects play a significant role in solidification through the Gibbs-Thomson relation Eq. (5). In the traditional phase field method, an asymptotic analysis is required to determine simulation parameters from the values of ε_c and ε_V used to model sharp front solidification. The present level set method solves the extended Stefan problem directly without any need for an asymptotic analysis.

4 The level set method

In this work, we consider finite difference approximations for the level set function calculation and volume-averaging based stabilized finite element techniques for modelling the thermal and fluid flow problems to capitalize on recent advances in the implementation of the level set method with finite difference techniques. Only structured grids are considered in this work. For the finite element discretization, four-node bilinear elements (in 2D) and eight-node brick trilinear elements (in 3D) are used.

Let ϕ be a signed distance variable (minimum distance to the interface between the two phases) satisfying $|\nabla\phi| = 1$. Then the normal direction of the interface (pointing from the solid to the liquid) is calculated as follows:

$$\boldsymbol{n} = \frac{\nabla\phi}{|\nabla\phi|}. \tag{22}$$

The curvature κ of the interface in terms of ϕ is computed as discussed in [21]. Equation (9) is solved for the level set function in a narrow band near the interface. For this extension, the interface velocity is calculated from the extended Stefan condition Eq. (21). Details of numerical schemes for the solution of Eq. (9) can be found in [15,22].

After an update according to the level set Eq. (9), ϕ does not in general remain a signed distance function. It is thus necessary for re-initialization where the following equation is iterated until reaching steady-state [16]:

$$\phi_t = \frac{\phi_0}{\sqrt{\phi_0^2 + \varepsilon^2}}(1 - |\nabla\phi|), \tag{23}$$

where ϕ_0 is the initial level set value to be re-initialized. After ϕ reaches steady-state, $|\nabla\phi| = 1$, i.e. ϕ is a signed distance. The parameter ε in Eq. (23) takes some small value and is needed for the formulation to remain well-posed as $\phi \to 0$. We use $\varepsilon = 2\Delta x$ in our calculations [16]. The time step in this re-initialization process is taken as $\Delta t = \frac{\Delta x}{5}$ and the number of iterations in the re-initialization process is taken to be $200\lambda_{CFL}$, where λ_{CFL} is the CFL condition coefficient ranging from 0 to 1. Note that the solid/liquid interface is advanced within a time step with a distance $\lambda_{CFL}\Delta x$. Thus based on the current re-initialization process, the more distance the interface is advanced forward within a time step, the more iterations will be required to rebuild the signed distance. Unless it is otherwise stated, the CFL coefficient in all the examples of Section 7 is selected as $\lambda_{CFL} = 0.3$.

5 Energy conserving level set method

The level set method has been successfully applied to Stefan problems in [16,17,23]. In this work, we apply the level set method to the extended Stefan problem. Because the Dirichlet temperature boundary condition is not applied directly on the interface, we can use energy conserving numerical schemes for the implementation of the heat transfer problem [18]. While the front-tracking method in [3] and the level set methods in [16,17,23] are all analytically energy conserving, when applying Dirichlet temperature boundary condition on the interface, numerically they do not conserve energy. For the method presented in this work (as well as in phase field methods), energy is not only conserved analytically but also numerically. This is the reason we refer to the present methodology as an 'energy conserving level set method'.

5.1 Stability analysis: Selection of the time step and k_N

To simulate sharp-front solidification, one will theoretically be required to consider a very high value for k_N. Such a choice will of course lead to prohibitively small time steps via the stability condition $\Delta t \leq \tau = 1/k_N$. In this work, our choice of Δt is based on the CFL condition for the level set function calculation i.e.

$$\Delta t \leq \lambda_{CFL} \frac{\Delta x}{V_{\max}}, \tag{24}$$

where λ_{CFL} is the CFL coefficient [21], and $V_{\max}$ is the maximum interface nodal velocity.

The subsequent choice of k_N is such that $\Delta t \leq 1/k_N$ to allow the interface temperature T_I to asymptotically converge to the equilibrium temperature T^*. As discussed earlier, our selection of k_N that satisfies the above condition is $k_N = \frac{1}{\Delta t}$.

In summary, the scheme to select Δt and k_N is the following:

(1) Choose λ_{CFL} between 0 and 1. In the 3D diffusion crystal growth under low-undercooling conditions example examined in Section 7.5.2, λ_{CFL} is selected as 0.1. In all other examples, we use $\lambda_{CFL} = 0.3$.
(2) Select a time step size as $\Delta t = \lambda_{CFL} \frac{\Delta x}{V_{\max}^{n-1}}$, where $V_{\max}^{n-1}$ is the maximum interface nodal velocity at the previous time level.
(3) Select $k_N = \frac{1}{\Delta t}$.
(4) Calculate the interface velocity according to Eq. (21) (see Section 5.2)
(5) Use the level set method to update ϕ.

5.2 Interface velocity calculation

In the level set method, the interface velocity V should be defined on the whole domain (or a narrow band near the interface). In the present algorithm, V is first computed on the nodes near the interface (depicted as empty circles in Fig. 4) using Eq. (21). A node is marked as being near the interface if at least one of its neighboring nodes has a different sign of ϕ. Equation (21) involves heat fluxes q_s and q_l, equilibrium temperature T^* and average temperature within the diffused interface $\bar{T}_I$. All these variables are computed on the nodes near the interface to obtain V using the methodology discussed in Sections 5.2.1 and 5.2.2. After V is computed on these nodes, it is extended to other nodes using the algorithm discussed in Section 5.2.3.

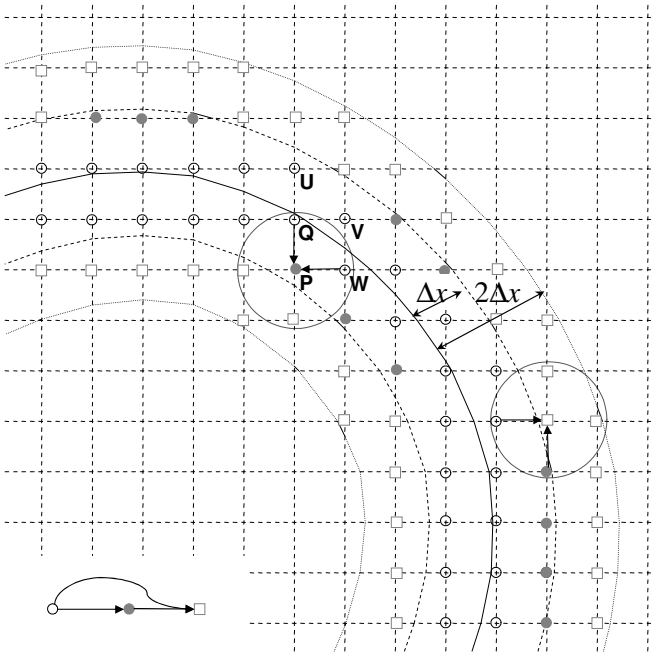


Fig. 4. Extending the interface velocity away from the freezing interface. Note that the velocity V is first computed at the points depicted with empty circles, then at the solid circle points and finally at the points shown with squares.

5.2.1 Calculation of the heat flux jump

We use the following two steps to compute q_s and q_l at the nodes near the interface:

(1) For nodes with $\phi \leq -w$, one computes $q_s = k_s \nabla T \cdot \boldsymbol{n}$, where ∇T is the temperature gradient approximated using central differencing or upwind differencing. Similarly, for nodes with $\phi \geq w$, one computes $q_l = k_l \nabla T \cdot \boldsymbol{n}$.
(2) Solving $\frac{\partial q_s}{\partial t} + \boldsymbol{n} \cdot \nabla q_s = 0$ in the region $\phi \in [-w, w]$ using the values of q_s in the region $\phi \leq -w$ as boundary condition extrapolates q_s in the normal interface direction. Similarly, solving $\frac{\partial q_l}{\partial t} - \boldsymbol{n} \cdot \nabla q_l = 0$ in the region $\phi \in [-w, w]$ using the values of q_l in the region $\phi \geq w$ as boundary condition extrapolates q_l in the opposite normal direction to the interface. This one-way extrapolation method is referred to as the 'ghost fluid method' in [21,24].

Note that after the second step above, q_s and q_l are computed on all nodes belonging to the region $\phi \in [-w, w]$. Since all nodes near the interface (empty circles in Fig. 4) are included in this region, q_s and q_l are now computed on all nodes near the interface.

5.2.2 Interpolation of the interface temperature

The equilibrium temperature T^* can be computed from the Gibbs-Thomson relation as follows:

$$T^* = T_m + \varepsilon_c \kappa + \varepsilon_V V^{n-1}. \tag{25}$$

As discussed earlier in Section 3, we will use T_I (temperature at $\phi = 0$) to approximate $\bar{T}_I$ (average temperature within the diffused-interface). However, in general, the nodes near the interface will not satisfy $\phi = 0$. So interpolation is necessary to obtain T_I. This can be easily computed using the following equation:

$$T_I = T - (\nabla T \cdot \boldsymbol{n})\phi, \tag{26}$$

where ∇T is calculated using simple differencing techniques.

5.2.3 Extending the interface velocity away from the interface

With q_s, q_l, T^* and T_I computed on the nodes near the interface, the interface velocity can be calculated at these nodes using Eq. (21). One can then perform a two-way extrapolation to extend the interface velocity to every other nodal point as described in [21,25]. In the present simulation, we use a slightly faster method defined by the following steps (see Fig. 4):

(1) Extend the interface velocity to band within Δx away from the interface. For all the points within the band Δx but not near the interface (for example, at point P in Fig. 4), draw a circle with radius $1.2\Delta x$, and calculate the average interface velocity for all the points near the interface and within this circle as its velocity. For example, $V_P = \frac{V_Q + V_W}{2}$.
(2) Extend the interface velocity to a band within $2\Delta x$ away from the interface. For all the points within the band $2\Delta x$ and outside the band Δx, draw a circle with radius $1.2\Delta x$, and calculate the average interface velocity for all the points within band Δx and within this circle as its velocity.

Using this method, one can extend the interface velocity $n\Delta x$ away from the interface with just n iterations. In the simulations of Section 7, the narrow band used to extend the velocity for solving the level set equation was $3\Delta x$ on each side of the interface. Although accuracy is sacrificed, this method is faster than the ghost fluid method, which is constrained by the CFL condition. After extension of the interface velocity, the velocity needed for the level set calculation can be defined within a narrow band. As an alternative method, one can use the fast-marching method to extend the interface velocity using a

heap data structure to achieve time complexity of only $O(NlogN)$, where N is the number of nodes where the velocity is extended [15].

5.3 Incorporating melt convection

It has been shown from numerical simulations [2,4,26–28] and experiments [29] that fluid flow has an important effect on crystal growth. There are a variety of ways to incorporate convection using the phase-field method or front-tracking methods. In [27], the solid is treated as a highly-viscous liquid by letting the viscosity depend on the phase field variable in the standard Navier-Stokes equations. In [26], the no-slip condition between the melt and the solid was realized via a drag resistivity in the diffused interface region. In [2], the Navier-Stokes equations are solved in two steps (with the first step considering only the advection and viscous terms and a trial pressure, and the second step considering only the pressure gradient). In these two steps, the unprojected velocity and the pressure gradient are multiplied by an index function to set the velocity in the solid to zero.

In this work, we treat the diffused-interface as a narrow 'mushy zone'. Volume-averaging is then applied to the whole region. The velocity in the solid region is set to zero, so that no-slip condition is applied at the solid/liquid interface. The formulation is briefly summarized below with more details provided in [19]. The flow equations are first re-cast in dimensionless form as follows:

$$\begin{aligned}\frac{\partial \boldsymbol{v}(\boldsymbol{x},t)}{\partial t} + \nabla\cdot\left(\frac{\boldsymbol{v}(\boldsymbol{x},t)\boldsymbol{v}(\boldsymbol{x},t)}{\Phi}\right) &= -\nabla p(\boldsymbol{x},t) + \frac{p(\boldsymbol{x},t)}{\Phi}\nabla\Phi \\ &\quad +\nabla\cdot[Pr(\nabla\boldsymbol{v}(\boldsymbol{x},t) + (\nabla\boldsymbol{v}(\boldsymbol{x},t))^T)] \\ &\quad -\frac{(1-\Phi)^2}{\Phi^2}\frac{Pr}{Da}\boldsymbol{v}(\boldsymbol{x},t) - \Phi PrRa_T\,\theta(\boldsymbol{x},t)\,\boldsymbol{e_g},\end{aligned}$$

where Pr is the Prandtl number defined as $\frac{\nu_l}{\alpha_l}$, Da is the Darcy number defined as $\frac{\alpha_l}{D_l}$ and Ra_T is the thermal Rayleigh number defined as $\beta_T|g|(T_0 - T_m)L^3/\nu_l\alpha_l$. The function spaces $S_{\boldsymbol{v}}$ and S_p are then introduced as follows:

$$\begin{aligned}S_{\boldsymbol{v}} &\equiv \{\boldsymbol{v}|\boldsymbol{v}\in L_2^{nsd}, \mathrm{div}\boldsymbol{v}\in L_2, \boldsymbol{v}=0 \text{ on } \partial\Omega\},\\ S_p &\equiv \{p|p\in L_2, \int_\Omega p d\Omega = 0\}.\end{aligned}$$

The classical Galerkin formulation for the flow problem can then be stated as: Find $\boldsymbol{V}\equiv\{\boldsymbol{v},p\}\in S_{\boldsymbol{v}}\times S_p$ such that for all $\boldsymbol{W}=\{\boldsymbol{w},q\}\in S_{\boldsymbol{v}}\times S_p$ $B(\boldsymbol{W},\boldsymbol{V}) = L(\boldsymbol{W})$ holds, where

$$B(\boldsymbol{W},\boldsymbol{V}) = \int_\Omega \boldsymbol{w}\cdot\left(\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v}\cdot\nabla\left(\frac{\boldsymbol{v}}{\Phi}\right) + \frac{(1-\Phi)^2}{\Phi^2}\frac{Pr}{Da}\boldsymbol{v}\right)d\Omega - \int_\Omega p\nabla\cdot\boldsymbol{w}d\Omega$$
$$+ \int_\Omega Pr\nabla\boldsymbol{w}\cdot(\nabla\boldsymbol{v}+\nabla\boldsymbol{v}^T)d\Omega + \int_\Omega q\nabla\cdot\boldsymbol{v}d\Omega,$$
$$L(\boldsymbol{W}) = \int_\Omega \frac{p}{\Phi}\nabla\Phi\cdot\boldsymbol{w}d\Omega - \int_\Omega \boldsymbol{w}\cdot\Phi PrRa_T\theta\boldsymbol{e_g}d\Omega.$$

In the finite element implementation of the Navier-Stokes equations, stabilizing techniques are needed to accommodate equal-order interpolation velocity-pressure elements. A stabilized FEM technique for porous media flows is presented in [19] and is briefly discussed below for completeness. After introducing a modified pressure space S_p' as follows:

$$S_p' \stackrel{\text{def}}{=} \{p|p\in H^1(\Omega),\ \int_\Omega p d\Omega = 0\}, \tag{27}$$

the stabilized weak form is the following: Find $\boldsymbol{V}=\{\boldsymbol{v},p\}\in S_{\boldsymbol{v}}\times S_p'$ such that for all $\boldsymbol{W}=\{\boldsymbol{w},q\}\in S_{\boldsymbol{v}}\times S_p'$ the following holds:

$$B_{\text{stab}}(\boldsymbol{W},\boldsymbol{V}) = L_{\text{stab}}(\boldsymbol{W}), \tag{28}$$

with:

$$B_{\text{stab}}(\boldsymbol{W},\boldsymbol{V}) = B(\boldsymbol{W},\boldsymbol{V}) + \int_\Omega \mathcal{F}(\boldsymbol{v},p)\cdot\mathcal{G}(\boldsymbol{w},q)d\Omega + \int_\Omega \tau_5\nabla\cdot\boldsymbol{v}\nabla\cdot\boldsymbol{w}d\Omega, \tag{29}$$
$$L_{\text{stab}}(\boldsymbol{W}) = L(\boldsymbol{W}) \ + \ \int_\Omega \{\frac{p}{\Phi}\nabla\Phi - \Phi\ PrRa_T\ \theta\boldsymbol{e_g}\}\cdot\mathcal{G}(\boldsymbol{w},q)d\Omega, \tag{30}$$

where $\mathcal{F}$ and $\mathcal{G}$ are defined as:

$$\mathcal{F}(\boldsymbol{v},p) = \frac{\partial\boldsymbol{v}}{\partial t} + \boldsymbol{v}_*\cdot\nabla\left(\frac{\boldsymbol{v}}{\Phi}\right) + \nabla p + \frac{(1-\Phi)^2}{\Phi^2}\frac{Pr}{Da}\boldsymbol{v} - Pr\nabla^2\boldsymbol{v}, \tag{31}$$
$$\mathcal{G}(\boldsymbol{w},q) = \tau_1\boldsymbol{v}_*\cdot\nabla\left(\frac{\boldsymbol{w}}{\Phi}\right) - \tau_2\frac{(1-\Phi)^2}{\Phi^2}\frac{Pr}{Da}\boldsymbol{w} - \tau_3 Pr\nabla^2\boldsymbol{w} + \tau_4\nabla q, \tag{32}$$

with $\boldsymbol{v}_*$ a divergence-free velocity, which in the implementation of Eq. (28) at a given time is usually taken as the known velocity at the previous time step. The particular values of the parameters $\tau_1,\ldots,\tau_5$ used in this work are given in [19]. Four-node bilinear finite elements (in 2D) and eight-node brick trilinear finite elements (in 3D) were used for both velocity and pressure interpolations. In closing, we note that in the problems examined in Section 7, $Ra_T = 0$ and the flow is induced by inlet velocity conditions.

6 Summary of the algorithm

A finite difference scheme is used for the level set calculations so that higher-order accuracy (third-order WENO scheme in space and third-order Runge-Kutta in time) can be achieved [22]. The same structured grid is used for both finite difference approximations in the level set calculations and the finite element approximations of the heat and flow problems. A summary of the overall algorithm is provided below.

(1) Update level set variable ϕ.
   (a) Copy ϕ to ϕ^{n-1}.
   (b) Copy V to V^{n-1}.
   (c) Determine the time step Δt and extended velocity V
(2) Copy Φ to Φ^{n-1}. Update Φ using ϕ according to Eq. (11).
(3) Solve the heat equation utilizing Φ^{n-1} and Φ. A fully implicit scheme is used in these calculations. The tolerance for the residual vector is taken as $||\boldsymbol{b}||_2 \leq 10^{-5}$. If no fluid flow effects are incorporated, the discretized equations are only solved once.
(4) Use ϕ to construct an artificial mushy zone and solve the fluid flow equations with tolerance of residual vector $||\boldsymbol{b}||_2 \leq 10^{-5}$, permeability $K_0 = 10^{-5}$, and $Ra_T = 0$.
(5) Set $t = t_{n-1} + \Delta t$. Return to step 1.

7 Numerical examples

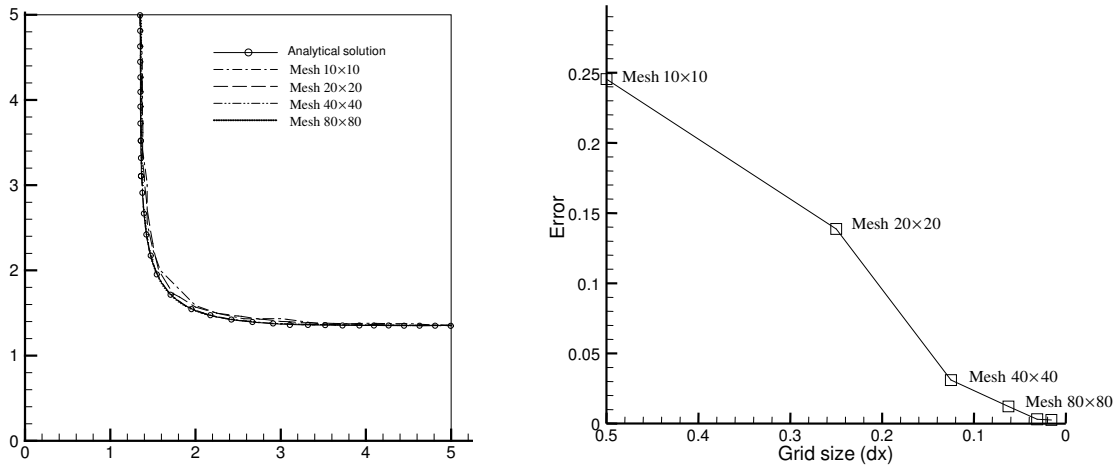
7.1 Solidification in a corner

This example is the solidification of a pure material in an infinite corner region with $k_s = k_l = 1$, $c_s = c_l = 1$, $L = 0.25$ and $\rho = 1$. The melting and initial temperatures are $T_m = 0$ and $T_{in} = 0.3$, respectively. A constant temperature condition $T_0 = -1$ is applied to the two boundary sides of the region. The analytical solution for the non-dimensionless interface position is given in [30].

To simulate this infinite corner Stefan problem, we use a domain of 5×5 discretized with a quadrilateral grid. At the left side and bottom sides of the domain, the temperature is kept at T_0, whereas the top and right sides are assumed to be adiabatic. This is only an approximation of the original problem with a solution that at early times should compare well with the analytical solution of the infinite corner problem.

We considered grids of different sizes. Figure 5(a) shows that the numerical

solution converges to the analytical solution very well. In Fig. 5(b), we define the error as the maximum distance of the calculated interface position from the analytical solution. This is computed by (1) finding all the elements cut by the zero level set, (2) interpolating points which are on element edges and satisfy $\phi = 0$, and (3) calculating distance of the interpolated points to the analytical solution, which is discretized into 100 points. As shown in Fig. 5(b), the error drops almost quadrati

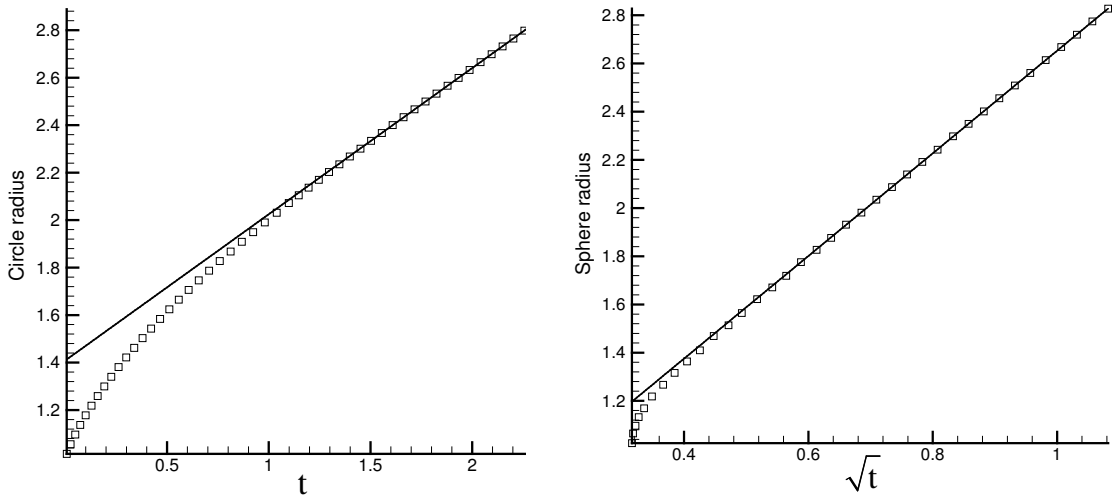


(a) (b)

Fig. 5. Convergence study of the infinite corner problem (time 0.9) (a) front position using various mesh sizes (b) maximum distance of the calculated interface from the analytical solution versus grid size.

7.2 Growing of a circle and a sphere in an undercooled melt

The dynamical evolution away from an unstable steady state was studied in [31,32]. It was found that under conditions favoring rapid solidification in 2D, the radius of the growing circle satisfies $R(t) \propto t$. In 3D, the radius of the growing sphere satisfies $R(t) \propto \sqrt{t}$. Using the presented numerical scheme, we simulated the growth of an initial seed with radius 1 with initial temperature 0 located in the middle of the domain $[-5, 5]^{\text{nsd}}$, where the number of spatial dimensions nsd= 2 for the growing circle example and nsd= 3 for growing sphere. The rest of the domain is at initial temperature $T_0 = -0.5$. The solid/liquid interface is always at temperature $T_m = 0$ without surface tension or kinetic undercooling. A constant temperature boundary condition $T = -0.5$ is applied at the boundary of the domain. Other parameters in the calculation are $k_s = k_l = 1$, $c_s = c_l = 1$, $L = 1$ and $\rho = 1$. We use a 100×100 mesh for the 2D simulation and $100 \times 100 \times 100$ mesh for the 3D simulation. Our results shown in Fig. 6 verify that after a "burn-in" period, the radius of the solidifying circle grows linearly with time in 2D and the radius of the solidifying sphere in 3D grows linearly with the square root of time.



(a) Growing circle (b) Growing sphere

Fig. 6. Radius of a solidifying circle and sphere in an undercooled melt.

7.3 Crystal growth in an undercooled melt: Effects of anisotropy and surface tension

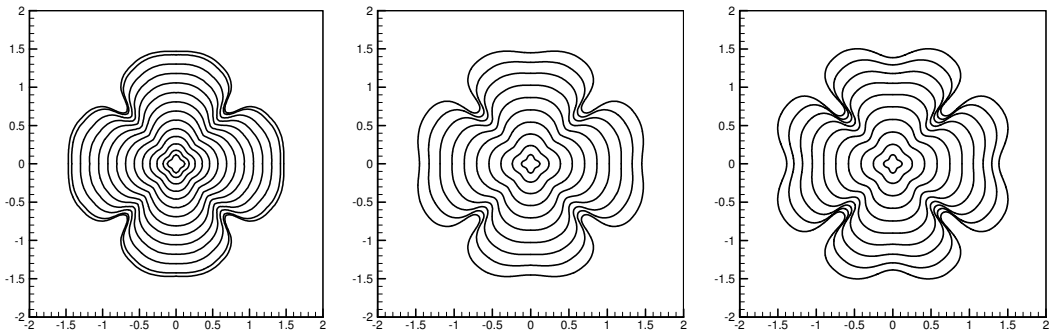
This example was originally addressed in [1] using a front-tracking method and re-examined in [16] using an implementation of the level set method. The material parameters defining the problem are $k_s = k_l = 1$, $c_s = c_l = 1$, $L = 1$, $\rho = 1$ and $T_m = 0$. On the freezing interface Γ we consider the classical Gibbs-Thomson relation given in Eq. (5) with $\varepsilon_c = \varepsilon_V = -0.002$. The computational domain is taken as $[-2, 2] \times [-2, 2]$. Insulated boundary conditions are considered at all sides of the two-dimensional domain.

At time zero, we consider a small solid seed in the middle of the computational domain. Its geometry is described as follows:

$$x(s) = (R + P\cos(8\pi s))\cos(2\pi s), \tag{33}$$
$$y(s) = (R + P\cos(8\pi s))\sin(2\pi s), \tag{34}$$

where $R = 0.1$ and $P = 0.02$. The initial temperature of the seed is taken as 0 and the initial temperature of the undercooled melt as -0.5. In the implementation of this example, we considered three different finite element meshes (100×100, 200×200 and 400×400). These studies allow us to investigate and report on the mesh-dependency of the results obtained with the present methodology.

For these calculations $\lambda_{CFL} = 0.3$. To accelerate our simulation, the time step size is adjusted automatically according to the CFL condition thus the lines shown in Fig. 7 are not evenly separated with time. The growth results



(a) Mesh 100×100 (b) Mesh 200×200 (c) Mesh 400×400

Fig. 7. Crystal growth in the presence of surface tension.

shown in [1,16] are in increments of 0.04 up to a final time of 0.8. Notice in Fig. 7 that at time 0.8, a steady-state has been reached and as expected the area of the computed solid region is exactly half of the total area of the domain. However, the area of the solid region at steady-state (time 0.8) is significantly different for the morphologies with low mesh resolution 100×100 comparing with high mesh resolution 400×400 in both [1] and [16]. The reason is that when Dirichlet boundary condition is applied at the solid/liquid interface, discretization error leads to high mesh-dependency. In our algorithm, we avoided applying Dirichlet boundary conditions on the interface so that energy conservation is satisfied leading to mesh-insensitive results.

We note that the grid refinement results shown in Fig. 7 compare well with those given in [16]. The morphologies obtained in [1] using a front-tracking technique appear to have a much higher mesh-dependency than the results reported here. The difference in published results [1,3,16,33] for this problem suggest that its solution is highly sensitive to perturbations during the solution process and that the problem is indeed a non-trivial one.

The above simulations were repeated but without the effects of surface tension, i.e. with $\varepsilon_c = 0$ and all other conditions in Eq. (5) as before. Figure 8 shows the results of this simulation for a mesh size 400×400. Comparing with the results in Fig. 7, it is seen that the crystal as expected is growing in a much more unstable mode.

7.3.1 Six-fold symmetric growth

We also computed the solution to a crystal growth problem under anisotropy with six-fold symmetry examined previously in [33]. The problem definition is similar to the earlier example.

At time zero, a small solid seed is put in the middle of the computational domain $[-2, 2] \times [-2, 2]$. Its geometry is described from Eqs. (33)-(34). The initial temperature of the seed is taken as 0 and the initial temperature of the under-

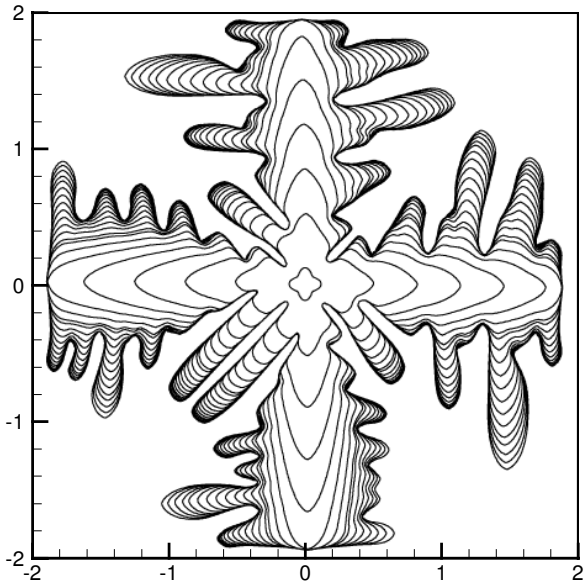


Fig. 8. Unstable crystal growth without curvature effect using a mesh of 400×400.

cooled melt as -0.8. The kinematic undercooling coefficient is constant with value $\varepsilon_v = -0.001$. The surface tension (curvature undercooling coefficient) is specified by the following anisotropic model with six-fold symmetry:

$$\varepsilon_c = -0.001\{1.0 + 0.4[\tfrac{8}{3}\sin^4 3(\theta - \pi/2) - 1.0]\}$$

All other material properties are normalized as 1.

In this example, the initial seed has a four-fold symmetry while ε_c has a six-fold symmetry. This difference allows us to study how initial perturbations affect the crystal growth. Using a grid of 400×400 and 800×800 and at final time 0.036, we obtain the crystal interfaces shown in Fig. 9.

From Fig. 9, we can conclude that the primary dendrite arms are determined by the growth mechanism (anisotropy), while the initial perturbations only affect the formation of secondary dendritic arms.

As pointed out in [33], the formed secondary dendritic arms are different for coarse and fine grid simulations (see Figs. 9 (a) and (b)), while the primary dendrite tips are growing with the same velocity for both grids (they all reach the computational boundary at time about 0.036). However, note that in the work of [33] using a front tracking method with markers, the total time for the crystal to reach the computational boundary varied from 0.035 with grid 800×800 to 0.045 with grid 400×400. The apparent improvement provided by the present methodology may be due to its energy conserving nature.

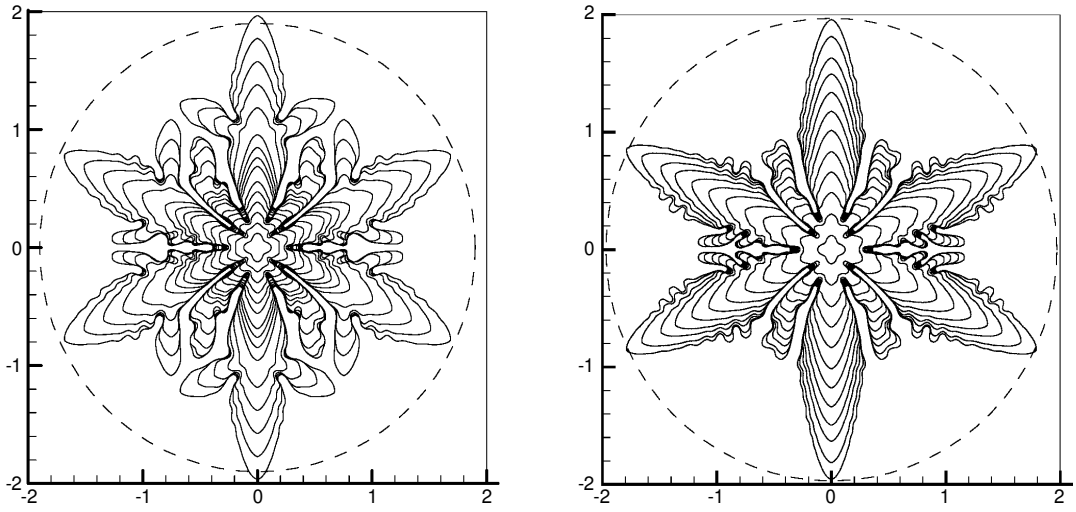


Fig. 9. Evolution of four-fold initial seed with six-fold surface tension anisotropy (a) Mesh 400×400 (b) Mesh 800×800.

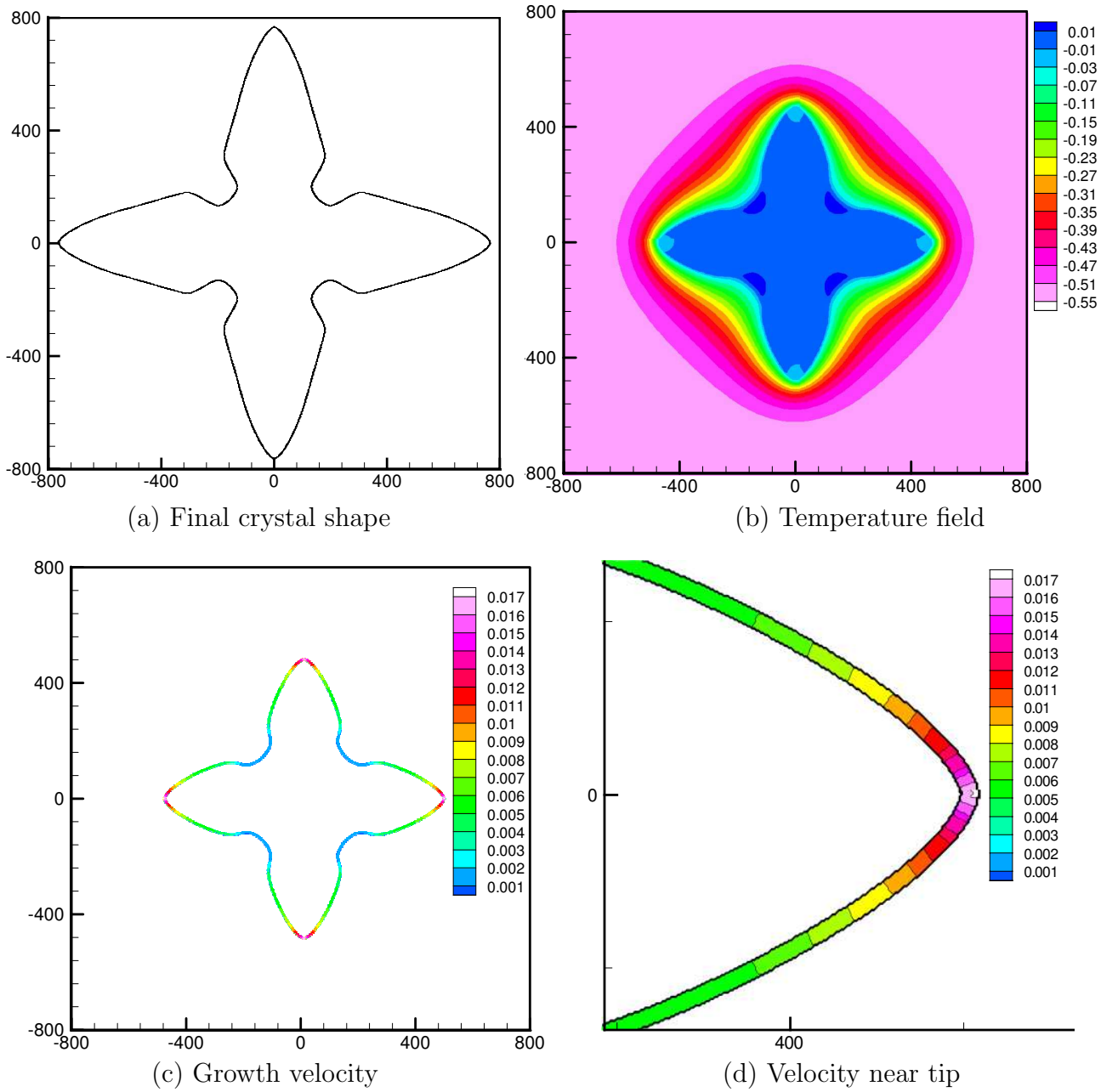
7.4 Two-dimensional steady-state dendritic growth: Comparison with solvability theory

Steady-state features of dendritic growth have been studied extensively using phase-field models [6,7] and the level set method [17]. These calculations have been in good agreement with the predictions of microscopic solvability theory. We will show here that an excellent agreement is also obtained using the developed energy conserving level set method. In the problem considered, the equilibrium temperature in the freezing interface Γ incorporates the effects of anisotropy as

$$T^* = -d_0(1 - 15\epsilon \cos 4\theta)\kappa, \tag{35}$$

with θ the angle between the outward normal and the x direction, $d_0 = 0.5$ and $\epsilon = 0.05$. The initial liquid temperature and boundary temperature (thus the undercooling considered) is $\Delta = 0.55$. The remaining material parameters are selected as $k_s = k_l = 1$, $c_s = c_l = 1$, $L = 1$ and $\rho = 1$. The obtained results are plotted using a normalized velocity $\tilde{V} = \frac{\rho c d_0 V}{k}$, a normalized position $\tilde{x} = \frac{x}{d_0}, \tilde{y} = \frac{y}{d_0}$, and dimensionless time $\tilde{t} = \frac{k}{\rho c d_0^2} t$. These dimensionless variables are also used in all of the following examples.

Our results shown in Figs. 10(a)-(d) obtained with a mesh of 800×800 compare fairly well with the numerical results obtained using the phase field method [6,7] and the level set method [17]. From the plot of the computed temperature field in Fig. 10b, we can observe that there is an undercooling at the dendrite tips due to positive curvature, whereas at places with a negative



(a) Final crystal shape (b) Temperature field

(c) Growth velocity (d) Velocity near tip

Fig. 10. Crystal growth at undercooling $\Delta = 0.55$.

curvature, the melt temperature is greater than zero (the melting point). In these simulations, the interface velocity is extended away from the interface to a distance of $3\Delta x$ on each side for solving the level set equation. At other places, the interface velocity is taken to be zero. From Fig. 10d, we can see that the interface velocity achieves a maximum at the dendrite tips. This value is about 0.016 for the results obtained using a mesh of 400×400 and 0.017 for the results using a mesh of 800×800. According to the solvability theory, the steady-state velocity at the dendrite tips is 0.017 [17].

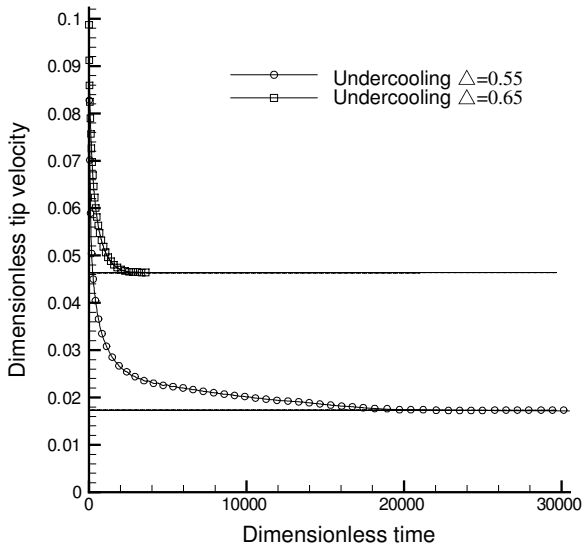


Fig. 11. Time evolution of the dimensionless dendrite tip velocity for various undercoolings.

The dendrite tip velocity is found to be very sensitive to the degree of undercooling. Using a slightly higher undercooling of 0.65, the steady-state dendrite tip velocity increases to about 0.047 as shown in Fig. 11. For the results in this figure, we have increased the computational domain to 1200×1200 using a mesh of 600×600. In this figure, we can find that the dendrite tip velocity settles to a value 0.017 at dimensionless time 22000. In the earlier reported case of a domain of 800×800, the thermal boundary layer eventually interacts with the system boundary and the asymptotic nature of the solution is lost.

Finally, in order to demonstrate the ability of the present method to enforce weakly the interface temperature condition, we also documented the equilibrium temperature at the dendrite tips for this example with undercooling 0.55 (Fig. 12).

While as discussed in Remark 2, an iterative process may be needed in general to enforce the interface temperature condition, in this example an explicit non-iterative process was sufficient. The equilibrium temperature falls in the range of $[-0.042, -0.002]$, which only varies about $0.04/0.55 = 7.2\%$ of the total undercooling. This small variation of the equilibrium temperature did not require the use of an iterative process such as the one shown in Fig. 3.

It has been pointed out in [17] that phase-field asymptotics for unequal solid/liquid diffusivities lead to computationally inconvenient forms and require extra grid resolution. The level set method by its nature avoids this difficulty. In order to demonstrate the ability of the present methodology to model unequal diffusivities, we calculated the same 2D crystal growth case using various diffusivities as shown in Fig. 13. From this figure, we can see that an increase of the liquid

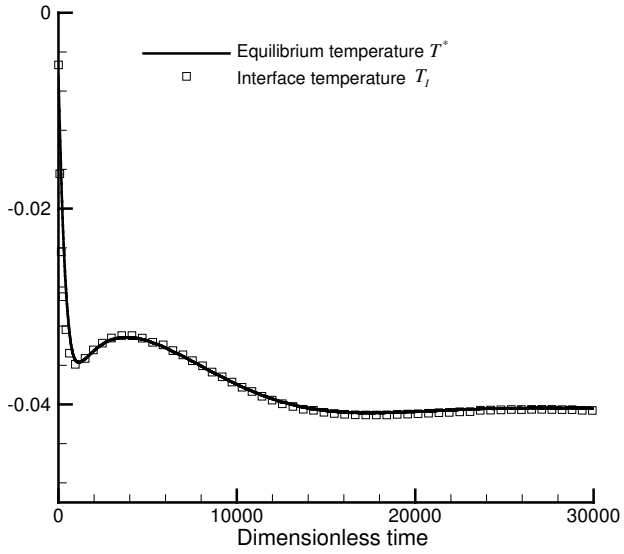


Fig. 12. Equilibrium temperature at dendrite tips of the 2D dendritic growth with undercooling 0.55.

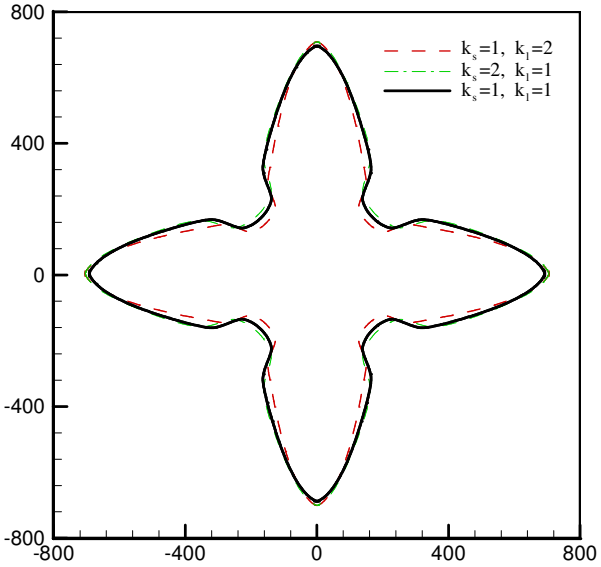


Fig. 13. Dendrite growth with unequal solid/liquid conductivities.

diffusivity tends to make the dendrite tips sharper, while an increase of the solid diffusivity would only make the dendrite tips a little bit fatter. A change of the liquid diffusivity was shown to affect the growth pattern more than a change of the solid diffusivity.

7.4.1 Off-axis solidification growth

In order to demonstrate that the presented algorithm works properly for off-axis growth, we recomputed the above example with a rotated surface tension

anisotropy as follows:

$$T^* = -d_0 \left(1 - 15\epsilon \cos(4(\theta + \frac{\pi}{4}))\right) \kappa. \tag{36}$$

The temperature field (at time about 10000) obtained by the original anisotropy and rotated anisotropy are shown in Figs. 14 (a) and (b). From Figs. 14 (a) and (b), we can conclude that the crystal growth is not affected by the grid orientation.

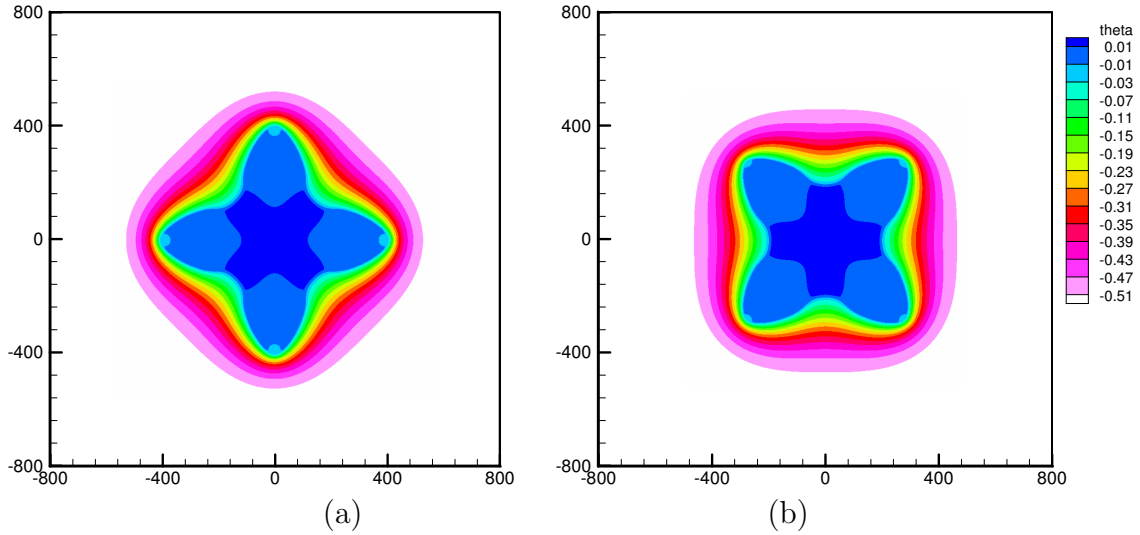
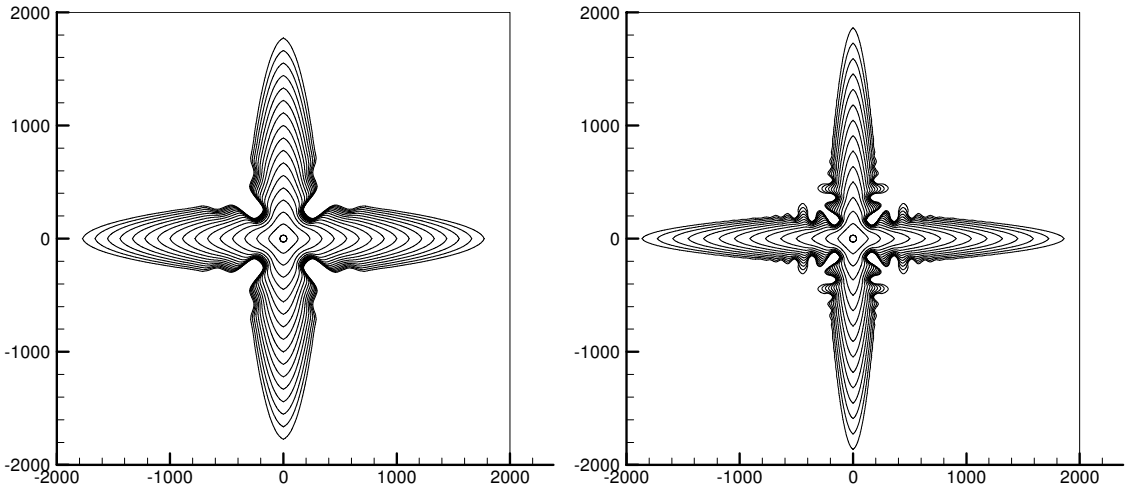


Fig. 14. Temperature field for the 2D solvability case with undercooling 0.55 (a) normal surface tension anisotropy (b) rotated surface tension anisotropy.

7.4.2 Formation of secondary dendrites

The formation of secondary dendrites is of great interest to many researchers, e.g. the mechanical properties are related with secondary dendrite arm spacing. In the solvability problem examined earlier with undercooling 0.55 and a computational domain of 800×800, no secondary arms were formed as shown in Fig. 10. In order to capture the formation of secondary dendritic arms, the computational domain is now increased to 2000×2000 using a grid of 800×800. When the tips are about 800 away from the center, the dendrite shapes are exactly the same as in Fig. 10(a). After that, small perturbations starts growing (see Fig. 15(a)). However, these perturbations do not evolve into secondary dendrites. This is due to the particular undercooling considered. Indeed, we performed another run with the same increased domain and an undercooling of 0.80. In this case, the perturbations eventually evolved into secondary dendrites as shown in Fig. 15(b). Calculations with such increased computational domain are given in [7,14], however only in the adaptive finite element calculations of [14] secondary arms are predicted.



(a) $\Delta = 0.55$ (b) $\Delta = 0.80$

Fig. 15. Front position evolution for the solvability problem using an enlarged domain and two-different undercoolings.

7.5 Three-dimensional dendritic growth

7.5.1 High-undercooling case

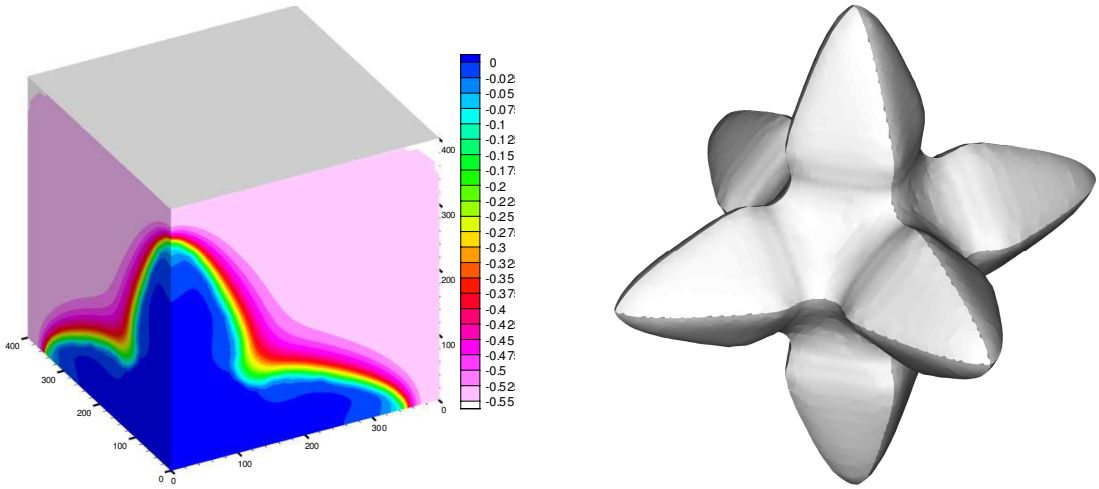
We herein examine the 3D case of the earlier solvability example using a mesh of $60 \times 60 \times 60$ with an undercooling of 0.55. To incorporate anisotropy in this 3D solidification growth example, we consider the following equilibrium temperature [4]:

$$T^* = -d_0(1 - \epsilon(4(n_1^4 + n_2^4 + n_3^4) - 3))\kappa, \quad \text{on } \Gamma, \tag{37}$$

where $d_0 = 0.5$, $\epsilon = 0.05$ and n_1, n_2, n_3 are the components of the normal unit vector along the x, y, z axes, respectively, calculated at each point on the freezing interface. The whole domain considered is $[-400, +400]^3$. Using symmetry, only 1/8 of the whole domain is calculated. Using an IBM T41 with a 1.7GHz CPU and 512 MB memory, the calculation time for the crystal to reach the state shown in Fig. 16 is about 10 hours. In this case, the undercooling is relatively large, so that the temperature field extends spatially as shown in Fig. 16.

7.5.2 Low-undercooling

It was observed that at low-undercoolings, the length scale to model the underlying transport processes is several orders of magnitude the tip radius [20]. However, the mesh step size Δx should be based on the tip radius in order to model the dendrite shape. This requires a very fine 3D grid, and substan-



(a) Temperature field (b) Crystal shape

Fig. 16. Temperature field and crystal shape at time $t = 105$ for 3D crystal growth at an undercooling $\Delta = 0.55$.

tially increases the computational difficulty for such problems. The following case with an undercooling of 0.05 is first presented in [20] to illustrate the computational power of the so called '*multiscale random-walk algorithm*' to address previously computationally unreachable range of low-undercoolings. Other parameters considered are the following [20]:

$$T^* = -d_0(1 - 3\epsilon)[1 - 4\epsilon(n_1^4 + n_2^4 + n_3^4)/(1 - 3\epsilon)]\kappa, \text{ on } \Gamma, \tag{38}$$

where $d_0 = 1$ and $\epsilon = 0.025$. Again using symmetry, we take a computation domain of $[0, +20000]^3$ with mesh 120^3 using 2 V1 nodes in the Cornell CTC center (with 4×500 MHz CPU on each node). Domain decomposition using 8 domains (processes) is used in this calculation.

For this low-undercooling case, the interface velocity is very small compared to the high-undercooling case. A smaller CFL coefficient of $\lambda_{CFL} = 0.1$ is thus used in this case to reduce the computation time step. The temperature field on the first process and the crystal shape is shown in Fig. 17 with domain $[0, +10000]^3$ using a mesh of 60^3. Since the other 7 processes provide only little additional information on temperature, the results of those processes are not shown here. The dimensionless tip velocity and tip radius as a function of time are shown in Fig. 18. These results are very close to those reported in [20]. The steady-state tip velocity and dendrite tip radius agree well with values reported in [20]. This demonstrates that the present methodology can successfully be applied to undercoolings as low as 0.05 thus bridging the disparity of length scales for modelling the tip radius and the thermal boundary layer. This ability is important since there is direct experimental relevance in this order of undercoolings [20].

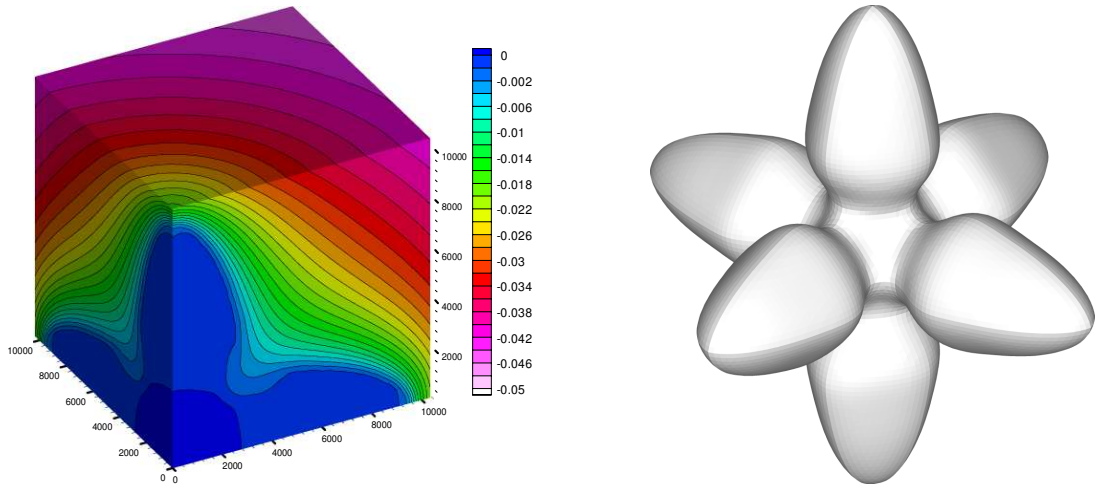


Fig. 17. Temperature field and crystal shape at time $t = 2.4 \times 10^8$ for 3D crystal growth at an undercooling $\Delta = 0.05$.

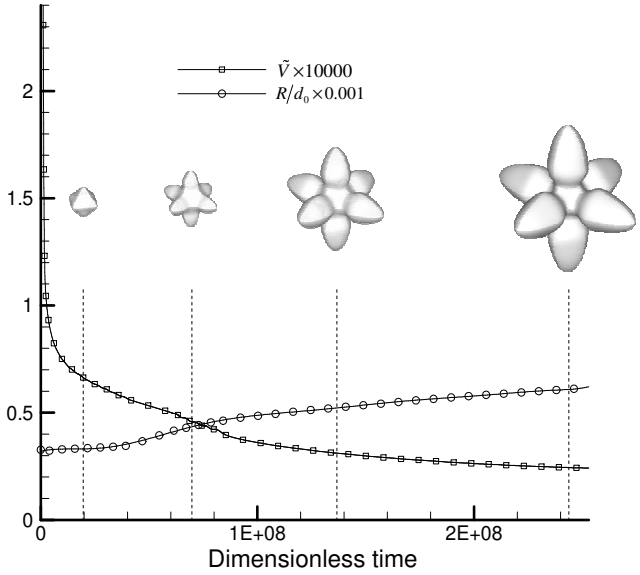
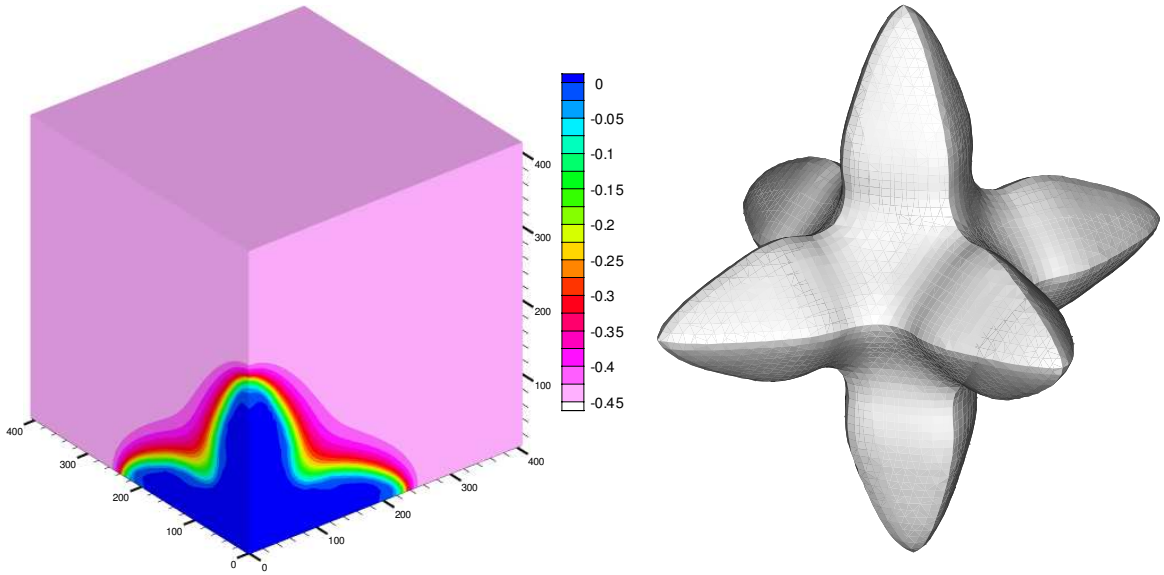


Fig. 18. Time evolution of the dimensionless tip velocity and dimensionless tip radius for 3D crystal growth at an undercooling $\Delta = 0.05$.

We also considered the equilibrium condition in Eq. (38) but with an undercooling of 0.45 and $\epsilon = 0.04$. This case was examined earlier in [34]. The obtained results in Fig. 19 show that the predicted dendrite shapes compare fairly well with those reported in [34] using the phase field method.

7.6 Two-dimensional crystal growth with convection

Beckermann et al. [26] presented the first calculations of dendritic growth in the presence of convection with a diffused-interface using the phase-field method. An implementation of dendritic growth with fluid flow using a front-



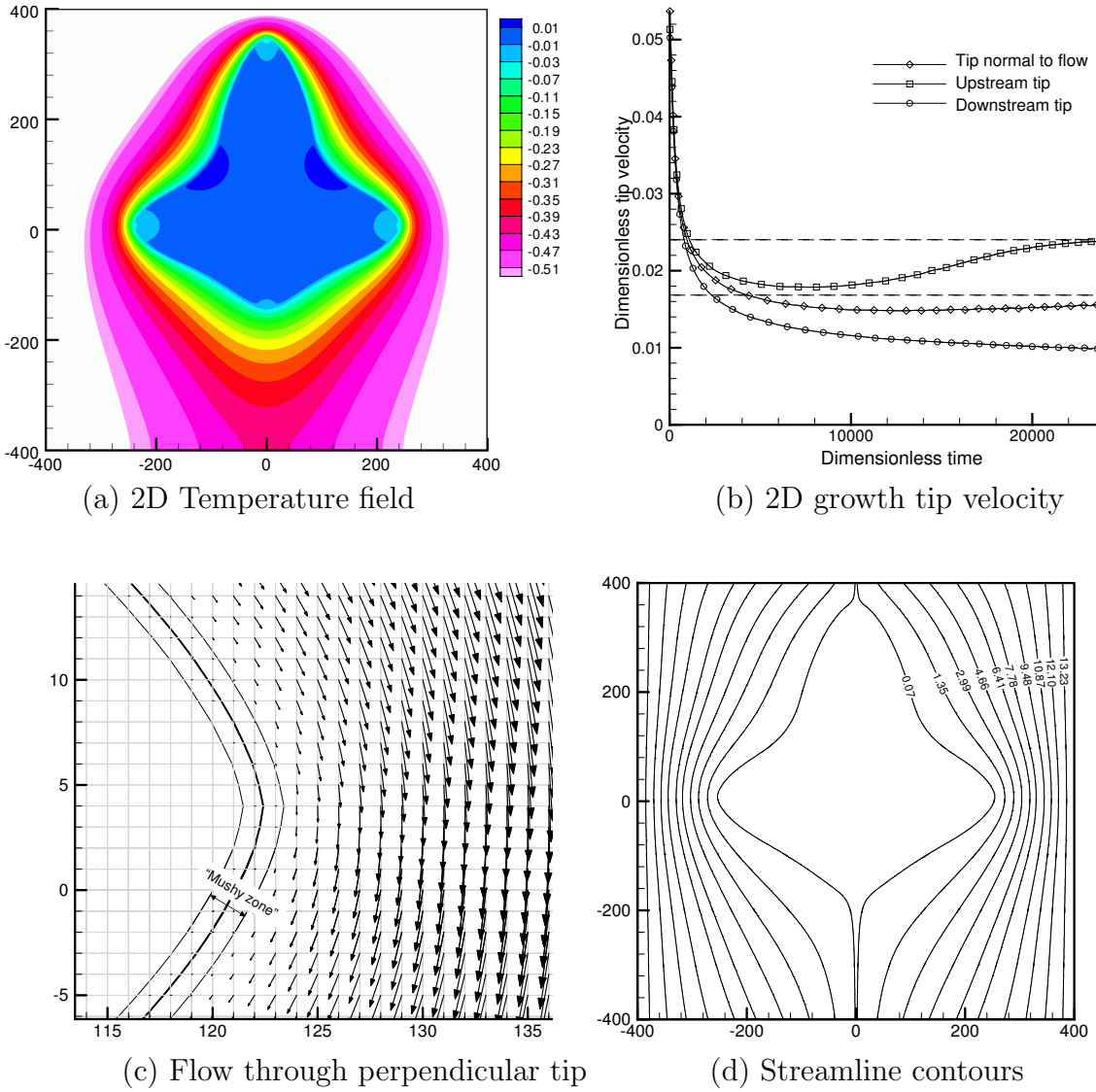
(a) Temperature field (b) Crystal shape

Fig. 19. Temperature field and crystal shape at time $t = 320$ for 3D crystal growth at an undercooling $\Delta = 0.45$. The predicted 3D crystal growth compares well with that obtained in [34] using the phase field method.

tracking method is given in [4]. The 2D case studied here is from [26]. The initial temperature is $T_{in} = -0.55$ and the inlet velocity $V_{inlet} = 0.035$ with Pr= 23.1. The remaining conditions are the same as those reported in the 2D solvability theory case.

The computed results are summarized in Fig. 20. The dendrite 'tilting' shown in Fig. 20(a) is due to the fact that the heat fluxes are higher on the upstream side than on the downstream side. The growth patterns predicted here compare fairly well with those in [26]. As shown in Fig. 20(a), we only calculated the solution on the right half of the domain $[0, 400] \times [-400, 400]$ using a mesh of 200×400. The left half is mirrored to show the results. Domain decomposition using 8 domains is also used in this calculation. The actual grid in each processor includes two rows of additional grid points in each outer surface to facilitate the communication of data at the boundary. This calculation was performed in the Cornell CTC supercomputer using 4 VII nodes (2×2.4 GHz CPU on each node) within about 5 hours. A schematic of the typical dendrite growth in the presence of fluid flow is shown in Fig. 20(a). The dimensionless velocity at the tips is shown in Fig. 20(b).

The flow passing through the perpendicular tip is shown in Fig. 20(c), while the streamline contours are shown in Fig. 20(d). As shown in Fig. 20(c), an artificial 'mushy zone' with width $2\Delta x$ is assumed. We use a permeability constant $K_0 = 10^{-5}$ so that the velocity is almost zero within this zone, as if the no-slip condition is applied at the boundary between the mushy-zone and the melt region. Since the half-width of the mushy zone is only Δx, the



(a) 2D Temperature field

(b) 2D growth tip velocity

(c) Flow through perpendicular tip

(d) Streamline contours

Fig. 20. Two-dimensional crystal growth with convection.

no-slip condition is applied on the position $\phi = \Delta x$. In our calculation, the mushy zone is identified using $\phi \in [-\Delta x, +\Delta x]$. One can also change position of the artificial mushy zone according to the signed distance variable ϕ, e.g. $\phi \in [-\Delta x, 0]$ or $\phi \in [-2\Delta x, 0]$, so that the no-slip condition can be exactly applied at $\phi = 0$. Comparing with the phase-field method [26] where special modelling of the interfacial stress term was used, it can be seen that the present method does not require an asymptotic analysis. Moreover, since the present method is a whole-domain method using volume-averaging, it is easy to implement and accurate when coupling heat transfer with fluid flow [19]. As it has been pointed out in [19], the volume-averaging model with stabilized FEM formulation converges nearly quadratically. Other methods such as the fractional step method do not show such rates of convergence.

Remark 5: Treating the diffused-interface as a porous medium [26] with the Kozeny-Carman approximation for the permeability is of no physical significance in the present calculations. It is simply a numerical tool for applying the no-slip boundary condition on the growing crystal using a fixed-grid. A variant viscosity in the diffused-interface can be applied as well [27]. In our calculations, we found that both of these methods lead to nearly identical results with velocity of very small magnitude within the mushy zone.

7.7 Three-dimensional crystal growth with convection

Two 3D examples are examined with high- and low-undercooling. These examples have been examined earlier in [34] and [4], respectively.

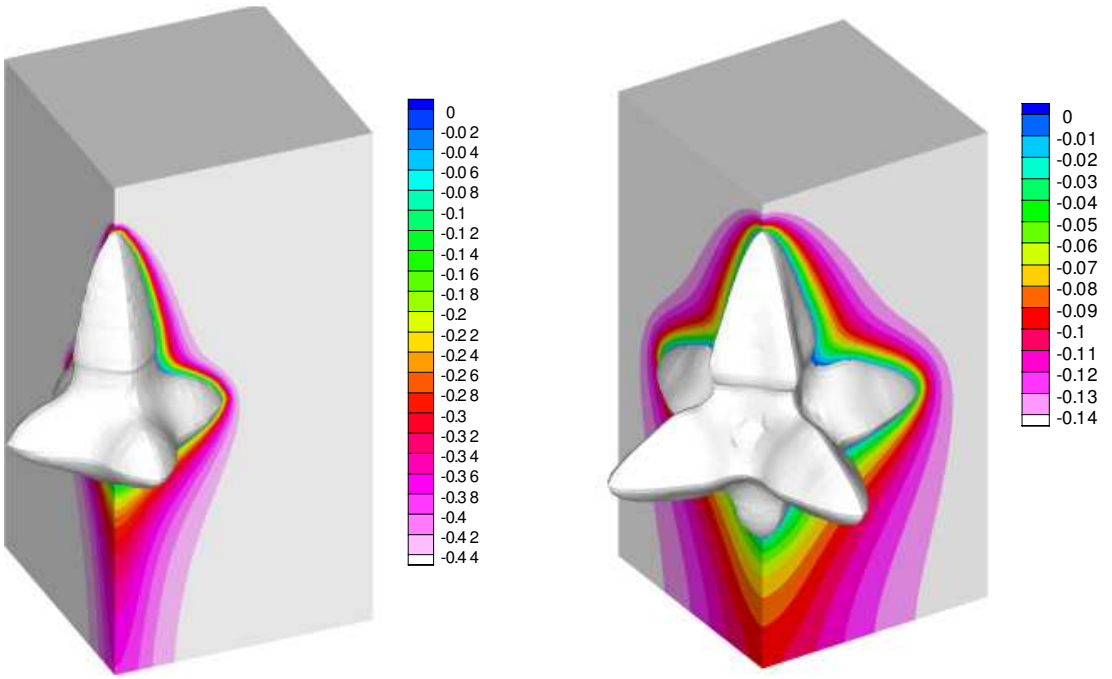
For the high-undercooling case, the undercooling is 0.45 and the inlet velocity is specified as 1.0. The equilibrium temperature on the interface is given from Eq. (38) where $d_0 = 1$ and $\epsilon = 0.04$. The computational domain is taken as $[0, 200] \times [0, 200] \times [-200, 200]$ with mesh size $60 \times 60 \times 120$.

For the low-undercooling case, the undercooling is 0.15, and the inlet velocity is 0.002. The equilibrium temperature on the interface is specified by Eq. (37) where $d_0 = 1$ and $\epsilon = 0.3$. The computational domain is taken as $[0, 12500] \times [0, 12500] \times [-12500, 12500]$ with mesh size $60 \times 60 \times 120$.

The temperature field and crystal shape for both cases are shown in Fig. 21. The computed 3D growth pattern is similar to the 2D growth pattern except that there are four perpendicular arms in 3D. Comparing Figs. 22(a) and 22(b), one can also observe that the crystal growth velocity is greatly affected by the degree of undercooling. The dendrite tip velocity differs by an order of about 10^2 for the two undercoolings considered. These results compare very well with those reported in [4,34].

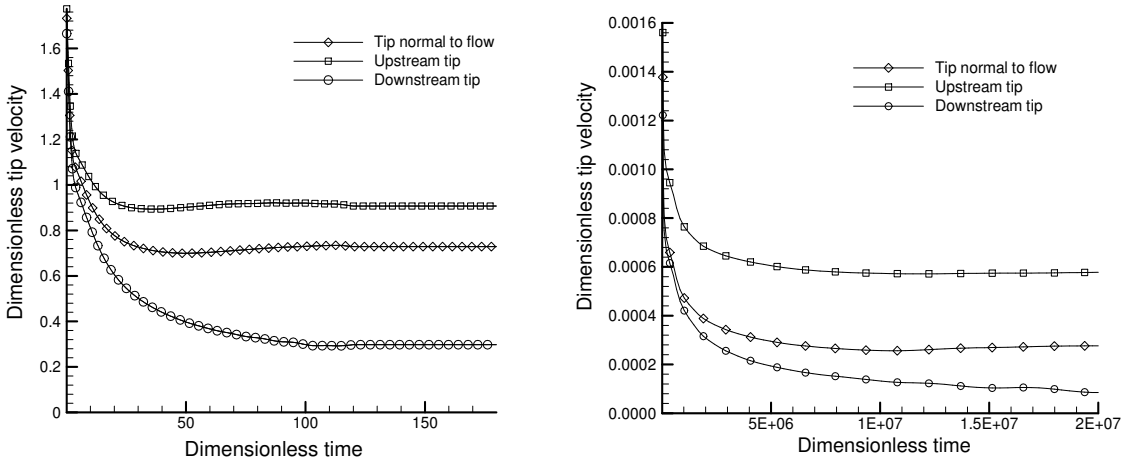
8 Conclusions

A method combining features of front-tracking and fixed domain methods is presented to model dendritic solidification of pure materials. Some of the key features of the presented method include (a) the use of a fixed-grid simulation for heat and momentum transfer, (b) energy conservation by avoiding the explicit application of temperature condition on the freezing front, (c) avoiding the direct application of the no-slip condition on the freezing front, (d) symmetric discretization of the heat equation thus avoiding the problems addressed in [17] and (e) automatic time step selection. The method is sub-



(a) (b)

Fig. 21. Crystal shape and temperature field for 3D growth in the presence of fluid flow (a) $\Delta = 0.45$, $\epsilon = 0.04$, $V_{inlet} = 1$, $Pr = 23.1$ at time $t = 163$ (b) $\Delta = 0.15$, $\epsilon = 0.3$, $V_{inlet} = 0.002$, $Pr = 1.0$ at time $t = 1.89 \times 10^7$.



(a) (b)

Fig. 22. Dendrite tip velocity evolution for 3D crystal growth in the presence of flow (a) $\Delta = 0.45$, $\epsilon = 0.04$, $V_{inlet} = 1$, $Pr = 23.1$ (b) $\Delta = 0.15$, $\epsilon = 0.3$, $V_{inlet} = 0.002$, $Pr = 1.0$.

stantially simpler to implement relative to front tracking or phase field models.

The numerical investigations shown here have demonstrated that the present method can serve as an accurate and computationally effective alternative tool

for modelling dendritic solidification. Current work is in progress to apply this methodology to alloy solidification.

9 Acknowledgements

The work presented here was funded by the NASA Microgravity Materials Science Program (grant NRA-98-HEDS-05) and by the University-Industry Partnerships for Aluminum Industry of the Future Program of the Office of Industrial Technologies of the U.S. Department of Energy (DE-FC07-02ID14396). The computing was supported by the Cornell Theory Center.

References

[1] D. Juric, G. Tryggvason, A front-tracking method for dendritic solidification, J. Comput. Phys. 123 (1996) 127–148.

[2] N. Al-Rawahi, G. Tryggvason, Numerical simulation of dendritic solidification with convection: two-dimension geometry, J. Comput. Phys. 180 (2002) 471-496.

[3] P. Zhao, J. C. Heinrich, Front-tracking finite element method for dendritic solidification, J. Comput. Phys. 173 (2001) 765–796.

[4] N. Al-Rawahi, G. Tryggvason, Numerical simulation of dendritic solidification with convection: three-dimension flow, J. Comput. Phys. 194 (2004) 677-696.

[5] W. J. Boettinger, J. A. Warren, C. Beckermann, A. Karma, Phase-field simulation of solidification, Annu. Rev. Mater. Res. 32 (2002) 163–194.

[6] N. Provatas, N. Goldenfeld, J. Dantzig, Efficient computation of dendritic microstructures using adaptive mesh refinement, Phys. Rev. Letters 80 (1998) 3308–3311.

[7] A. Karma, W.J. Rappel, Quantitative phase-field modelling of dendritic growth in two and three dimensions, Phys. Rev. E 57 (1998) 4323–4349.

[8] W.W. Mullins, R.F. Sekerka, Stability of a planar interface during solidification of a dilute binary alloy, J. Appl. Phys. 35 (1964) 444–451.

[9] A. Wheeler, W. Boettinger, G. Mcfadden, Phase-field model for isothermal phase transitions in binary alloys, Physical Review A 45 (1992) 7424-7440.

[10] S.L. Wang, R. F. Sekerka, A.A. Wheeler, B. T. Murray, S. R. Coriell, Thermodynamically consistent phase field models for solidification, Physica D 69 (1993) 189–200.

[11] A. Karma, W.J. Rappel, Phase field model for computationally efficient modelling of solidification with arbitrary interface kinetics, Phys. Rev. E 53 (1996) 3017–3020.

[12] R. Almgren, Second-order phase field asymptotics for unequal conductivities, SIAM J. Appl. Math. 59 (1999) 2086-2107.

[13] J.A. Warren, W.L. George, A parallel 3D dendritic growth simulator using the phase-field method, J. Comput. Phys. 177 (2002) 264–283.

[14] N. Provatas, N. Goldenfeld and J. A. Dantzig, Adaptive mesh refinement computation of solidification microstructures using dynamic data structures, J. Comput. Phys. 148 (1999) 265–290.

[15] J.A. Sethian, Level set Methods (Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Material Science, published by the Press Syndicate of the University of Cambridge, 1996.

[16] S. Chen, B. Merriman, S. Osher, P. Smereka, A simple level set method for solving Stefan problems J. Comput. Phys. 135 (1997) 8–29.

[17] Y.T. Kim, N. Goldenfeld, J. Dantzig, Computation of dendritic microstructures using a level set method, Physical Review E 62 (2000) 2471–2474.

[18] Y. Jaluria, K. E. Torrance, Computational Heat Transfer, second edition, published by Taylor & Francis, 2003, pp. 49-54.

[19] N. Zabaras, D. Samanta, A stabilized volume-averaging finite element method for flow in porous media and binary alloy solidification processes, International J. for Numerical Methods in Engineering, 60 (2004) 1103–1138.

[20] M. Plapp, A. Karma, Multiscale random-walk algorithm for simulating interfacial pattern formation, Physical Review Let. 84 (2000) 17401-17443.

[21] S. Osher, R. Fedkiw, Level set Methods and Dynamic Implicit Surfaces, Springer-Verlag, New York, Inc. 2003.

[22] S. Osher, R. P. Fedkiw, Level Set Methods: An Overview and Some Recent Results, J. Comput. Phys. 169 (2001) 463–502.

[23] F. Gibou, R. Fedkiw, R. Caflisch, S. Osher's, A level set approach for the numerical simulation of dendritic growth, Journal of Scientific Computing, Vol.19 (2003) 183–199.

[24] R. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (The Ghost Fluid Method), J. Comput. Phys. 152 (1999) 457–492.

[25] H. Zhao, T. Chan, B. Merriman, S. Osher, A Variational level set method for multiphase motion, J. Comput. Phys. 127 (1996) 179–195.

[26] C. Beckermann, H.-J. Diepers, I. Steinbach, A. Karma, X. Tong, Modeling melt convection in phase-field simulations of solidification, J. Comput. Phys. 154 (1999) 468–496.

[27] R. Tonhardt, G. Amberg, Phase-field simulation of dendritic growth in a shear flow, Journal of Crystal Growth, 194 (1998) 406–425.

[28] M. Lai, C. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2000) 705–719.

[29] Y. Lee, R. Ananth, W. Gill, Selection of a length scale in unconstrained dendritic growth with convection in the melt Journal of Crystal Growth, Journal of Crystal Growth 132 (1993) 226–230.

[30] K.A. Rathjen, L.M. Jiji, Heat conduction with melting or freezing in a corner, J. Heat Transfer ASME 93 (1971) 101–109.

[31] D. Jasnow, J. Vinals, Dynamical scaling during interfacial growth following a morphological instability, Physical Review A 40 (1989) 3864–3870.

[32] G. Caginalp, Dynamical renormalization group calculation of a two-phase sharp interface model, Physical Review E 60 (1999) 6267–6270.

[33] P. Zhao, J.C. Heinrich, Numerical approximation of a thermally driven interface using finite elements, International Journal for Numerical Methods in Engineering 56 (2003) 1533–1547.

[34] J. Jeong, N. Goldenfeld, J. Dantzig, Phase field model for three-dimensional dendritic growth with fluid flow, Physical Review E 64 (2001) 041602.