JOURNAL OF COMPUTATIONAL PHYSICS **127,** 110–117 (1996)
ARTICLE NO. 0161

Algorithms for Phase Field Computation of the Dendritic Operating State at Large Supercoolings

SHUN-LIEN WANG* AND ROBERT F. SEKERKA

Department of Physics, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213

Received August 2, 1995; revised February 6, 1996

This paper deals with numerical solutions of the phase field model of solidification in two dimensions in the context of dendritic growth. Finite difference methods associated with vectorized algorithms are employed to solve the phase field equations. The temperature equation is solved by an alternating direction implicit scheme and the equation for the phase field by an explicit Euler scheme. We identify a region of parameter space, namely large supercoolings, where we can compute dendritic morphologies that display steady-state dendrite tip operating conditions that are independent of computational parameters. © 1996 Academic Press, Inc.

I. INTRODUCTION

The classical approach to modeling a first-order phase transition involves finding solutions to the governing partial differential equations in the domains of the growing phase and nutrient phase, subject to boundary conditions on the unknown moving boundary that separates these phases. This is a difficult free boundary problem to implement numerically because it involves tracking of the boundary through a fixed grid, or the implementation of solution-adaptive grids that move with the boundary.

Another approach is to use the phase field model [1, 2], which has recently become a popular computational tool for modeling complicated solidification problems. This model avoids explicit boundary tracking by replacing the PDE for the temperature field and its boundary conditions on the moving boundary by two coupled PDEs involving the temperature field and a new field, the phase field ϕ that keeps track of the phase. The variable ϕ assumes a representative value in each phase, say, 0 in the solid and 1 in the liquid, and changes rapidly but continuously in the vicinity of the sharp phase boundary of the classical model. In the asymptotic limit of a suitably thin transition layer, the classical boundary conditions at the crystal-melt interface are satisfied.

For many years, the phase field model did not lead to significant computational results in more than one dimension. Recently, increased computing power has enabled the computation of quite complex morphologies, such as dendritic growth in two dimensions [3–9]. There has even been some success with computations in three dimensions [10].

In our previous research [11], we introduced a phase field model that is based on positive local entropy production. Furthermore, as shown in [12], the model is capable of incorporating arbitrary anisotropies of surface tension and interface kinetics, believed to play important roles in pattern selection during the growth of a dendrite. By means of this improved phase field model, our attention is directed toward the quantitative determination of the dendrite operating state for dendritic growth from a pure melt in two dimensions. Specifically, we seek to compute the dendritic tip growth velocity, v, and the tip radius of curvature, ρ, for various dimensionless supercoolings, $S = (T_M - T_\infty)c/L_o$, where T_M is the absolute melting point, T_∞ is the absolute far field temperature, c is the volumetric heat capacity, and L_o is the latent heat per unit volume. Extensive results of this study are published elsewhere [13]. Here we present the details of the numerical algorithms that we have implemented for computations.

First of all, we have achieved an enhanced vectorized algorithm which reduces computational time by not attempting to update the value of the phase field in the bulk liquid and solid. The considerations that we use to improve the performance of the code are addressed in this paper. Moreover, we have repeated the computations done in [6] in two dimensions and shown that, for the range of dimensionless supercooling $S \sim 0.5$ that they used, the results are not independent of computational parameters. In particular, the Peclet number, $P = v\rho/2\kappa$, where κ is the thermal diffusivity of the melt, can be calculated with reasonable accuracy, whereas the individual values of v and ρ are found to vary considerably with the numerical grid size and another computational parameter, $\bar{\varepsilon}$, that characterizes the diffuse interface width. (Some evidence for this behavior is also apparent from Table II and Table III in [6].) The sensitivity of results to computational parameters is mainly due to the large differences, for $S \sim 0.5$, among the relevant lengths involved in dendritic

---

* Current address: Technology Modeling Associates, Inc., Sunnyvale, CA 94086.

0021-9991/96 $18.00

growth, e.g., capillary length, interface thickness, dendrite tip radius, and system size. By analyzing these length scales, we have discovered that quantitative results independent of computational parameters can be obtained only at large supercoolings, $S \sim 1$, for reasons that will be explained below.

This paper is organized as follows. The phase field equations used for computation are presented in Section II. The numerical implementation for the resulting equations is discussed in Section III. In Section IV, we discuss the proper range of physical parameters for which results independent of numerical parameters can be obtained.

II. ANISOTROPIC DYNAMICAL EQUATIONS IN TWO DIMENSIONS

The dynamical equations in two dimensions are [11, 12, 14]

$$\frac{\partial u}{\partial t} + \frac{30\phi^2(1-\phi)^2}{S}\frac{\partial \phi}{\partial t} = \nabla^2 u \tag{1}$$

for the dimensionless temperature, $u = (T - T_M)/(T_M - T_\infty)$, which reduces to the heat equation in the region where $\phi = 0$ and $\phi = 1$, and, for the phase field,

$$\frac{q(\theta)}{\bar{m}}\frac{\partial \phi}{\partial t} = F(\phi, u) + \nabla \cdot (B(\theta)\nabla\phi), \tag{2}$$

where

$$B(\theta) = \begin{pmatrix} r^2(\theta) & -r(\theta)r'(\theta) \\ r(\theta)r'(\theta) & r^2(\theta) \end{pmatrix}. \tag{3}$$

Equation (2) can be viewed as a second-order parabolic PDE that contains a "source" term

$$F(\phi, u) = \frac{1}{\varepsilon^2}\phi(1-\phi)\left[\phi - \frac{1}{2} + 30\bar{\varepsilon}\alpha S\frac{u}{1 + (L_o/cT_M)u}\phi(1-\phi)\right]. \tag{4}$$

Here, lengths are scaled in units of the characteristic domain size, w, and time in units of w^2/κ.

With θ defined as the angle between a level set $\phi =$ const and a reference direction, presumed to be of crystallographic significance, i.e.,

$$\theta = \arctan\left(\frac{\phi_y}{\phi_x}\right)\Bigg|\ \phi = \text{const}, \tag{5}$$

proper choices for the functions $r(\theta)$ and $q(\theta)$ enable independent control of the anisotropies of surface tension and interface kinetics [12]; this is possible because ϕ is practically constant, except in the interfacial region, which corresponds to $\phi = \frac{1}{2}$ in the sharp interface limit. As an example, which will be used in subsequent computations, we choose

$$r(\theta) = 1 + \delta_\gamma \cos(4\theta), \tag{6}$$

$$q(\theta) = \frac{1 + \delta_\gamma \cos(4\theta)}{1 + \delta_\mu \cos(4\theta)}. \tag{7}$$

This corresponds asymptotically to an anisotropic surface tension, $\gamma(\theta) = \bar{\gamma}[1 + \delta_\gamma \cos(4\theta)]$, and an anisotropic kinetic coefficient, $\mu(\theta) = \bar{\mu}[1 + \delta_\mu \cos(4\theta)]$, where δ_γ and δ_μ are the corresponding strengths of anisotropy, respectively. For the special case when $\delta_\gamma = \delta_\mu = 0$, Eqs. (1)–(2) reduce to Eqs. (36) and (37) in [11].

The parameter $\bar{m}$ is the ratio of the capillary length, $d_o = c\bar{\gamma}T_M/L_o^2$, to the kinetic length, $\rho_k = c\kappa/(\bar{\mu}L_o)$, $\bar{\varepsilon} = \bar{\delta}/w$ is the ratio of the interface thickness parameter, $\bar{\delta}$, to w and $\alpha = w/(6\sqrt{2}d_o)$. Note that $\bar{m}$ depends only on material parameters, whereas both $\bar{\varepsilon}$ and α depend on the computational parameters, $\bar{\delta}$ and w. The domain size, w, is a computational parameter because steady state dendritic growth should be independent of w.

III. NUMERICAL IMPLEMENTATION

In solving Eqs. (1)–(2), we choose to discretize the equations by using a second-order finite difference scheme on a grid of uniform squares with grid spacing δx. For the temporal discretization, a time step δt is introduced. We assume that initially there is a small circular solid located at the corner of a domain with dimensions X_L and Y_L in the x and y directions, respectively. Based on our experience with respect to a square domain, $X_L = Y_L$, and a rectangular domain, $Y_L = X_L/2$, the calculated results of dendritic operating states for dendrites that grow along the X_L direction are found to be insensitive to the two types of domain for the large supercoolings that we use (see Section IV), even though dendrites that grow along the Y_L direction are impeded and distorted by the walls of the rectangular domain. This observation is important because it means that we can save computational time by concentrating our study along the X_L direction in a domain of rectangular shape. As a result, we adopt a rectangular domain for subsequent computations. Vanishing conditions for normal derivatives of both u and ϕ (Neumann conditions) are applied at the boundary; i.e., the dendrite will grow with reflection symmetry with respect to the x and y axes. To ensure that the boundary conditions have negligible influence on the selection of operating state, we make sure that the domain size is sufficiently large to simulate a free growth environment. This is done by monitoring the operating state as the dendrite grows along the X_L direction: If a well-defined steady state is not achieved before the distance between the dendrite tip along the X_L direction and the domain boundary is already smaller than the corresponding thermal diffusion length (equal to the ratio of thermal diffusivity to the tip velocity), a bigger computational domain is adopted. A steady state at the dendrite tip is defined when the tip velocity varies less than 5% over a distance of 10 tip radii.

Since solving Eq. (2) by implicit time-differencing is very difficult due to the implicit dependence of $q(\theta)$ and $B(\theta)$ on ϕ, we employed an explicit Euler scheme for Eq. (2). The allowable time step is thus subject to some stability criterion which may be more restrictive than that for the linear part of the equation: $\delta t \leq (\delta x)^2/(4\bar{m}/q(\theta))$. Therefore, the actual size of the time step is determined by numerical experimentation. With explicit time-differencing, however, Eq. (1) would be subject to a more restrictive time step requirement than Eq. (2) because the parameter $\bar{m}$ usually has a value less than one. Therefore, an implicit time-differencing that is unconditionally stable, specifically, the alternating direction implicit method [15], is employed for Eq. (1). Rather than solving the phase field equations over all grid points, as was done in [6], we next show that the computational algorithm can be further enhanced by considering the physical nature of the phase field, ϕ.

From the physical point of view, the phase field varies rapidly across the interface. In most of the computational domain, either $\phi \simeq 0$, which denotes bulk solid, or $\phi \simeq 1$, which denotes bulk liquid; most of these values will remain unchanged at each iteration. From the computational point of view, F in Eq. (4) has the value zero if ϕ is 0 or 1. Moreover, the *central difference* method is applied to evaluate spatial derivatives, such that a typical grid point is related to its four nearest neighbors only; thus, if the value of ϕ at grid point ($x_i = i\delta x$, $y_j = j\delta x$), $\phi_{i,j}$, and its four nearest neighbors, $\phi_{i+1,j}$, $\phi_{i-1,j}$, $\phi_{i,j+1}$ and $\phi_{i,j-1}$, all have the same value, then all the spatial derivative terms in the equation yield zero. The value of $\phi_{i,j}$ at time step $n + 1$ is, therefore, unchanged from its value at time step n. Conversely, if $\phi \neq 0, 1$ such that $F \neq 0$, then the value of $\phi_{i,j}$ at time step $n + 1$ is very likely to differ from the value at time step n. In addition, if $\phi_{i,j} \neq 0, 1$ then $\phi_{i,j}$ will contribute non-zero values when calculating the derivative terms with respect to any one of its four nearest neighbors. As a consequence, the values of ϕ at the four nearest neighbors of a point at time step $n + 1$ will possibly differ from their values at time step n. We conclude that only those points (diffuse interface) for which $\phi \neq 0, 1$ and their four nearest neighbors are required to be updated at each iteration. The number of such points are significantly smaller than the number of total grid points because the interfacial region is much smaller than the total bulk domain. Therefore, computational time will be saved by not attempting to update the values of ϕ for points in the bulk liquid and solid. This results in computational advantages that are similar to those obtained by Elliott and Schätzle who used a "double-obstacle" potential [16].

We have therefore improved the algorithm, originally developed by researchers at NIST [6, 7] for solving Eq. (2) by further vectorizing the code as follows: In addition to two components containing the values of ϕ and u, respectively, the data structure associated with each grid point is expanded to include a boolean variable, called *mark*. If the value of ϕ at the grid point needs to be updated, *mark* = TRUE; otherwise, *mark* = FALSE. To identify the initial diffuse interface for computation, we first assign *mark* = FALSE for all of the grid points. Next, a logic operation, if ($\phi \neq 0, 1$) (within some tolerance less than the order of truncation error of the discretized method), is performed throughout all of the grid points. If the logic operation is true, then *mark* = TRUE is assigned to this grid point and to the nearest neighbors of this point. In the time iteration loop, those points with *mark* = TRUE are listed as a one-dimensional array (vectorization), allowing the phase field equation to be solved for only these points, rather than the entire grid. This results in great efficiency because it takes full advantage of the architecture of a highly vectorized supercomputer. Note that this vectorization process requires a logic operation, if (*mark* = TRUE), throughout all of the grid points. However, to identify the diffuse interface for the next iteration, it is necessary to perform a logic operation, if ($\phi \neq 0, 1$), only for the points within the present diffuse interface. (Of course, those grid points with *mark* = TRUE must first be reverted to be *mark* = FALSE.)

Although two additional logic operations are needed and more memory is required to store *mark* and the one-dimensional array, this modification has significantly sped up the running performance as follows:

On a rectangular domain with a total 1400 × 700 grid points, we monitor the CPU time consumed by updating the phase field and temperature at each time step. As shown in Fig. 1, about 0.085 s on the Cray C90 are required to update the values of the phase field parameter at all grid points; yet, using our enhanced algorithm to update the values of ϕ only inside the diffuse interface requires a computing time of the order of 0.015 s. As a dendrite grows, the total number of grid points that need to be updated increases due to the growth of total interface length, with the width of the interface being nearly constant. Therefore the computing time is accordingly increased as well.

In Fig. 2, we plot the CPU time needed to update points in the diffuse interface as a function of the number of grid points in the diffuse interface. As expected from our algorithm, the CPU time increases roughly linearly with

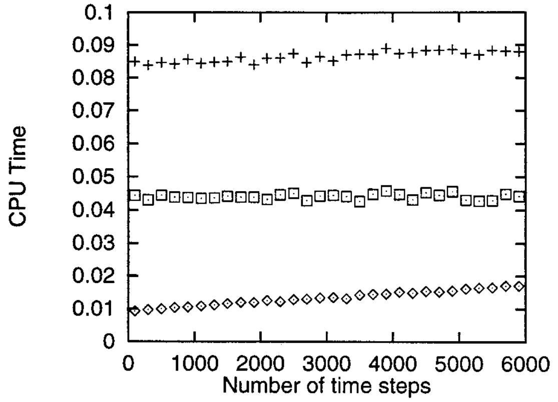


FIG. 1. CPU time (per time step) consumed for updating the phase field equations. The cross marks are for updating the values of ϕ for all of the grid points (1400 × 700). The diamond marks are for updating the values of ϕ only in the diffuse interface. The square marks are for updating the values of u at all grid points.

the total size of the diffuse interface. Therefore, the CPU time would be about 0.085 s (the NIST value) if there were 250,000 grid points in the diffuse interface, which is about *one quarter* of the total grid points for the whole domain. Such a morphology would represent a very convoluted body. Up until such time as a dendrite shape becomes highly convoluted in the above sense, our algorithm will be superior. In our calculations, a dendrite reaches a steady state when there are about 60,000 grid points in the diffuse interface; hence it takes about 0.027 s to update ϕ at each iteration. Therefore, our algorithm has improved the performance by at least three times for the computation of the phase field alone. On the other hand, because the calculation of Eq. (1) takes about 0.045 s for each iteration, the overall computing time needed to solve both of the partial differential equations is decreased with respect to the NIST algorithm by about a factor of $(0.085 + 0.045)/(0.027 + 0.045) \approx 2$. At earlier stages of the computation, the relative efficiency is even greater, close to a factor of 2.5.

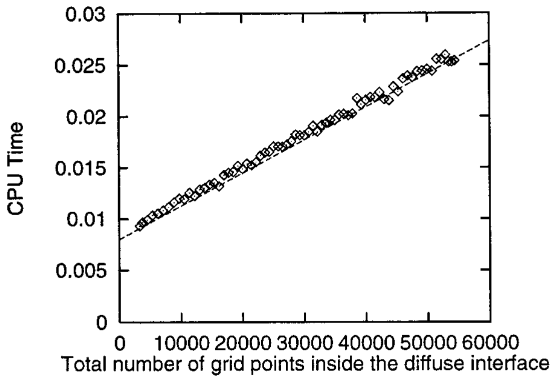


FIG. 2. CPU time (per time step) versus the total number of interface grid points for the phase field, according to the enhanced algorithm. The dashed straight line through the computed data is a least squares fit.

Although our algorithm requires tracking of the diffuse interface, it need not be so precise as the classical algorithm for the sharp interface model. In the free boundary problem, the moving boundary conditions *explicitly* involve geometric properties of the boundary itself, such as the local curvature and the normal direction. The calculations of the first- and second-order spatial derivatives are therefore needed to be as accurate as possible in order to represent the boundary condition. As a result, a much finer mesh is needed to resolve the interface geometry, especially for a complicated boundary. This requirement, however, is not so restrictive for the phase field model since the interfacial boundary condition is included *implicitly* in the phase field model.

IV. SELECTION OF PARAMETERS

In the discretized form of Eqs. (1) and (2), there are four physical parameters: S, $\bar{m}$, δ_γ, and δ_μ. In addition, there are four computational parameters: ε, α, the mesh size δx and the time step δt.

As discussed in [11, 12], a free boundary problem is recovered in the limit that $\varepsilon \to 0$ with S, α, and $\bar{m}$ of order one. That is, the dimensionless temperature at the interface becomes [12],

$$u = -1 \Big/ S\left\{L_o/(cT_M) + \left([1 - 15\delta_\gamma \cos(4\theta)]\frac{1}{\tilde{\rho}} + \frac{\tilde{v}/\bar{m}}{[1 + \delta_\mu \cos(4\theta)]}\right)^{-1}\right\}, \tag{8}$$

where $\tilde{v}$ is the dimensionless tip growth velocity (in units of κ/d_o) and $\tilde{\rho}$ is the dimensionless tip radius (in units of d_o). Therefore, in order to get this model to give results that mimic those suggested by the asymptotics, proper selection of the computational parameters is necessary for practical numerical computations.

For a fixed set of physical parameters and for a given value of δx, we choose δt small enough to obtain numerical stability for the ϕ equation, as discussed above. Then, for $\varepsilon = \frac{1}{200}, \frac{1}{400}$, and $\frac{1}{800}$, respectively, we conducted a series of computations with various $\delta x = \frac{1}{200}, \frac{1}{400}, \frac{1}{800}$, and $\frac{1}{1000}$, while keeping the other parameters fixed. The calculated results of $\tilde{v}$ and $\tilde{\rho}$ are shown to be sensitive to the choice of δx for $\delta x > \varepsilon$, but insensitive to δx for $\delta x \leq \varepsilon$. Therefore, we choose $\delta x = \varepsilon$ for all subsequent computations. In practical terms, this means that there will be about eight grid points along any interval where the phase field varies from zero to one. Next, we discuss the choices of ε and α.

TABLE I

The Calculated Tip Velocity $\tilde{v}$, Tip Radius $\tilde{\rho}$, and Tip Temperature u for Four Different Values of $\bar{\varepsilon}$

$\bar{\varepsilon}$	α	$\tilde{v}$	$\tilde{\rho}$	u	$\lvert u - u_{GT}\rvert / u$
$\frac{1}{200}$	70	0.00955	89.1	−0.2200	7.7%
$\frac{1}{400}$	70	0.01034	79.9	−0.2245	2.3%
$\frac{1}{600}$	70	0.01038	78.1	−0.2233	1.4%
$\frac{1}{800}$	70	0.01035	78.6	−0.2219	1.0%

Note. u_{GT} is calculated from the modified Gibbs–Thomson equation, Eq. (8), using the values of $\tilde{\rho}$ and $\tilde{v}$. The last column provides a quantitative measure of the closeness of the phase field model to the sharp interface model. The results were computed for parameters $S = 0.9$, $\bar{m} = 0.05$, and $\delta_\gamma = \delta_\mu = 0.03$ on a domain with $X_L = 3.5$ and $Y_L = 1.75$. The tip velocity is scaled by κ/d_o and the tip radius by d_o.

In Table I, we list some calculated results (after reaching a steady state) for $\tilde{v}$, $\tilde{\rho}$, and u at the interface for several different values of $\bar{\varepsilon}$. The quantity u_{GT} is the interface temperature calculated from the modified Gibbs–Thomson equation, Eq. (8), by using the computed tip radius and tip velocity. Hence, the term $\lvert u - u_{GT}\rvert / u$ provides a quantitative measure of the closeness of the phase field model to the sharp interface model.

As illustrated in Table I, when $\bar{\varepsilon}$ is refined from $\frac{1}{200}$ to $\frac{1}{800}$, the results change very little and are in better agreement with the asymptotic formula for the tip temperature, which is valid in the limit $\bar{\varepsilon} \to 0$. Since the product of α and $\bar{\varepsilon}$ is $\bar{\delta}/(6\sqrt{2}d_o)$, the values of $\bar{\delta}/d_o$ in Table I correspond to about 3, 1.5, 1, and 0.7, respectively.

Additionally, in Table II, we list some results which correspond to different values of α with the other parameters remaining constant. The values of $\bar{\delta}/d_o$ in Table II correspond to 2.8, 1.7, and 1, respectively.

Clearly, Table I and Table II consistently indicate that the average interface thickness parameter $\bar{\delta}$ must be about the size of d_o in order to "resolve" the Gibbs–Thomson equation. This condition is in agreement with results by Braun *et al.* [17] who used the phase field model to do a linear stability analysis. Thus, we proceeded to investigate the effects of $\bar{\varepsilon}$ while keeping the value of $\bar{\delta}$ to be about equal to d_o. We have actually studied the effects of changing many parameters over broad ranges. For example, Fig. 3 shows that the values of tip velocity and tip radius for cases with $\bar{\varepsilon} = \frac{1}{400}$ and $\alpha = 50$ agree satisfactorily with those calculated with $\bar{\varepsilon} = \frac{1}{800}$ and $\alpha = 90, 70, 70$ in curves (1), (2), and (3), respectively. Note that we have chosen $\delta x = \bar{\varepsilon}$ in all cases; therefore, the computational time for $\bar{\varepsilon} = \frac{1}{800}$ is 16 times larger than for the cases with $\bar{\varepsilon} = \frac{1}{400}$. In light of this, we will adopt $\bar{\varepsilon} = \frac{1}{400}$ and $\alpha = 50$ for our subsequent calculations.

Since the tip radius of curvature must be much greater than the width of the interfacial region and the computational domain must be much greater than the tip radius in order for the solutions to reach a steady state before the dendrite hits the domain boundary, the true computa-

TABLE II

The Calculated Tip Velocity $\tilde{v}$, Tip Radius $\tilde{\rho}$, and Tip Temperature u for Three Different Values of α

$\bar{\varepsilon}$	α	$\tilde{v}$	$\tilde{\rho}$	u	$\lvert u - u_{GT}\rvert / u$
$\frac{1}{400}$	130	0.00744	73.5	−0.188	2.5%
$\frac{1}{400}$	80	0.00763	75.4	−0.189	1%
$\frac{1}{400}$	50	0.00769	76.7	−0.190	<1%

Note. The results were computed for the parameters $S = 0.8$, $\bar{m} = 0.05$, and $\delta_\gamma = \delta_\mu = 0.03$ on a domain with $X_L = 3.5$ and $Y_L = 1.75$.

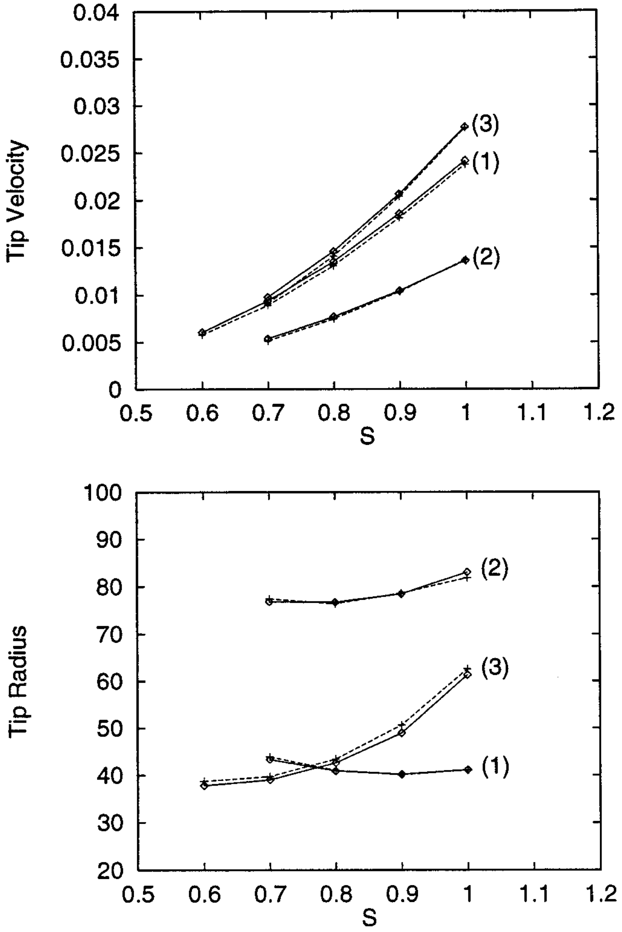


FIG. 3. Comparison of calculated tip velocity and tip radius for two values of $\bar{\varepsilon}$. The diamond marks are for $\bar{\varepsilon} = \frac{1}{400}$ and $\alpha = 50$. The cross marks are for $\bar{\varepsilon} = \frac{1}{800}$ and $\alpha = 90, 70, 70$ in curves (1), (2), (3), respectively. In curve (1), $\delta_\gamma = \delta_\mu = 0.03$ and $\bar{m} = 0.1$. Curve (2) is the same as curve (1) except $\bar{m} = 0.05$. In curve (3), $\delta_\gamma = 0.04$, $\delta_\mu = 0$, and $\bar{m} = 0.1$. The anisotropies are assumed to have fourfold sinusoidal form. The tip velocity is scaled by κ/d_o and the tip radius by d_o.

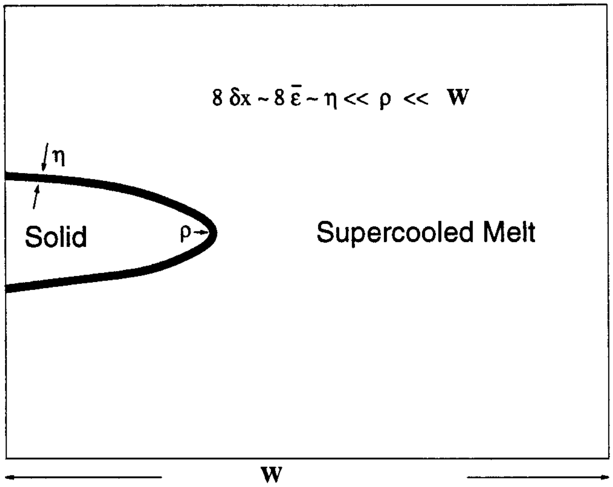


FIG. 4. A sketch to illustrate selection of parameters for numerical computations of dendritic growth by the phase field model.

tional domain is chosen to be $X_L = 3.5$ and $Y_L = 1.75$ in units of w. As a result, there are at least $(3.5 \times 400) \times (1.75 \times 400) = 980{,}000$ grid points in our computations for $\bar{\varepsilon} = \frac{1}{400}$. Typically, about one CPU hour on Cray C90 was required to detect a well-defined steady state at a dendrite tip.

As illustrated schematically in Fig. 4, we conclude the following: In terms of the capillary length d_o, good results (reasonable independence of computational parameters and tip temperature close to that given by asymptotic analysis) can be obtained under the following conditions:

- The grid spacing is about d_o.
- The phase field, ϕ, changes from 0.05 to 0.95 in about $8\ d_o$.
- The characteristic size, w, of the computational domain is at about $400\ d_o$; the length along the x direction of the actual computational domain is 3.5 or 7 times w, i.e., $(1400 \sim 2800)\ d_o$.
- The dendrite tip radius must be large compared to $8\ d_o$ but small compared to $1400\ d_o$.

It is this last condition that limits us to large supercoolings, $S \sim 1$, for the reasons to be stated next.

The changes of dendrite tip radius with supercooling can be estimated by consideration of an approximate analytical theory as follows. For infinitely rapid interface kinetics ($\bar{m} \to \infty$) and for a steady state isothermal dendrite of parabolic shape, the supercooling can be related to the Peclet number, P, by the following equation due to Ivantsov (in two dimensions) [18]:

$$S_I = \sqrt{\pi P}\, \exp(P) \operatorname{erfc}(\sqrt{P}), \tag{9}$$

where erfc (U) is the complementary error function of argument U and where we have subscripted S with I to note the restricted nature of this formula. According to marginal stability criterion [19, 20], or to microscopic solvability theory [21–26], the operating state of the dendrite is given by

$$\frac{2\kappa d_o}{\rho^2 v} = \sigma, \tag{10}$$

where σ is a number given by an appropriate theory; e.g., typically, $\sigma \simeq \sigma^* = 0.025$.

Combining Eq. (10) with the definition of the Peclet number gives

$$\tilde{v} = 2\sigma P^2 \tag{11}$$

$$\tilde{\rho} = \frac{1}{\sigma P} \tag{12}$$

which, together with Eq. (9), constitute a parametric representation of the dendritic operating state.

Most experiments to date have been conducted for small values of supercooling, say $S \sim 0.1$, which would lead, through Eq. (9), to small values of P and, hence, from Eq. (12) to values of the tip radius $\tilde{\rho} > 10{,}000$. This is well outside our computationally accessible range. Similarly, even the values $S \sim 0.5$, used by the NIST group, lead to values of $\tilde{\rho}$ that are too large to fall into our computationally accessible range, except for very large anisotropies of surface tension.

For large supercoolings $S_I \to 1$, however, one can show that Eq. (9) leads asymptotically to

$$P \sim \frac{1}{2(1 - S_I)} \tag{13}$$

which for values of S_I sufficiently close to 1 will make P sufficiently large to put $\tilde{\rho}$ into our computationally accessible range.

Actually, for large supercoolings, the situation is more complicated because capillary and especially kinetic corrections to the dendrite tip temperature become large. Under these conditions, we have derived from Eqs. (8) and (9) the approximate modified relationship

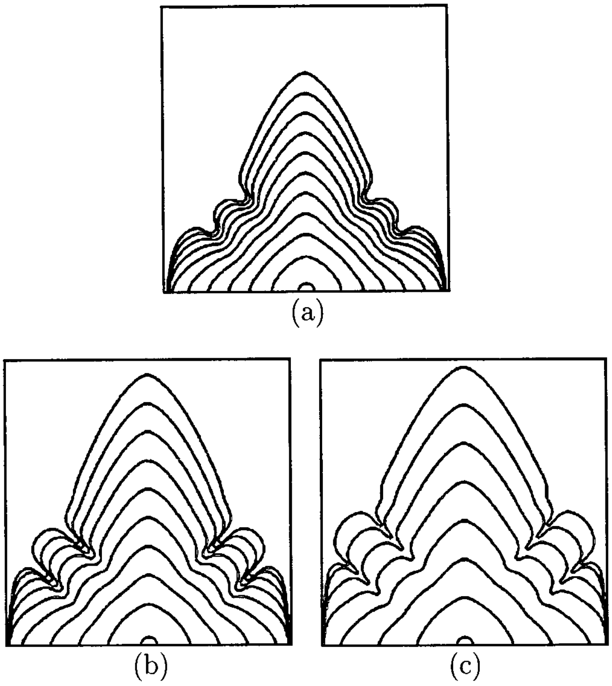
$$S = S_I + \frac{\sigma P(1 + 2P/\bar{m})}{1 + (L_o/cT_M)\sigma P(1 + 2P/\bar{m})} \tag{14}$$

which, together with Eqs. (11) and (12), give parametrically the approximate operating state of the dendrite. Guided by these approximate relationships, we have found that large supercoolings, e.g., $0.7 \le S \le 1.1$ lead to a computationally accessible range for $\tilde{\rho}$. The value of $S = 0.8$ corresponds to an actual supercooling of 347 K for nickel, which is attainable [27].

Some results for dendritic growth in the range of $0.7 \leq S \leq 1.1$ are shown in this section. Figures 5a–c show the computed evolution of the morphology of a growing dendrite for various large supercoolings. Our results also show the presence of a well-defined steady state near the tip region. An example is shown in Fig. 6 in which the tip growth velocity, tip radius, and tip temperature are plotted as a function of time. After an initial transient, each of the quantities approaches a limiting value. Also apparent in the figures are some oscillations which are attributed to numerical noise. In Fig. 7, we show $\tilde{v}$ and $\tilde{\rho}$ as functions of the supercooling. Studies in investigating the influence of other physical parameters on the values of $\tilde{v}$ and $\tilde{\rho}$ are straightforward.

V. CONCLUSIONS

In this paper, we have applied finite difference schemes to solve the phase field model for dendritic growth on two-dimensional rectangular domains with a uniform square mesh. Specifically, the equation for the temperature is solved by an alternating direction implicit scheme and the equation for the phase field by an explicit Euler scheme. An enhanced vectorized algorithm is achieved since only those grid points inside the diffuse interface need to be updated at each iteration. Computation time is reduced


(a)

(b) (c)

FIG. 5. Influence of supercooling on the evolution of a dendrite with parameters $\bar{m} = 0.1$, $\delta_\gamma = \delta_\mu = 0.03$. The corresponding values of S are 0.8, 0.9, 1.0 for (a) to (c), respectively. The growth shapes are shown at time increments of 0.04 for the following time intervals: (a) 0.0–0.4, (b) 0.0–0.36, (c) 0.0–0.28. At each time level, three contours of $\phi = 0.1, 0.5, 0.9$ are plotted. Note that actual computations were performed only for the right half of the figure and the solutions are reflected about the midplane.

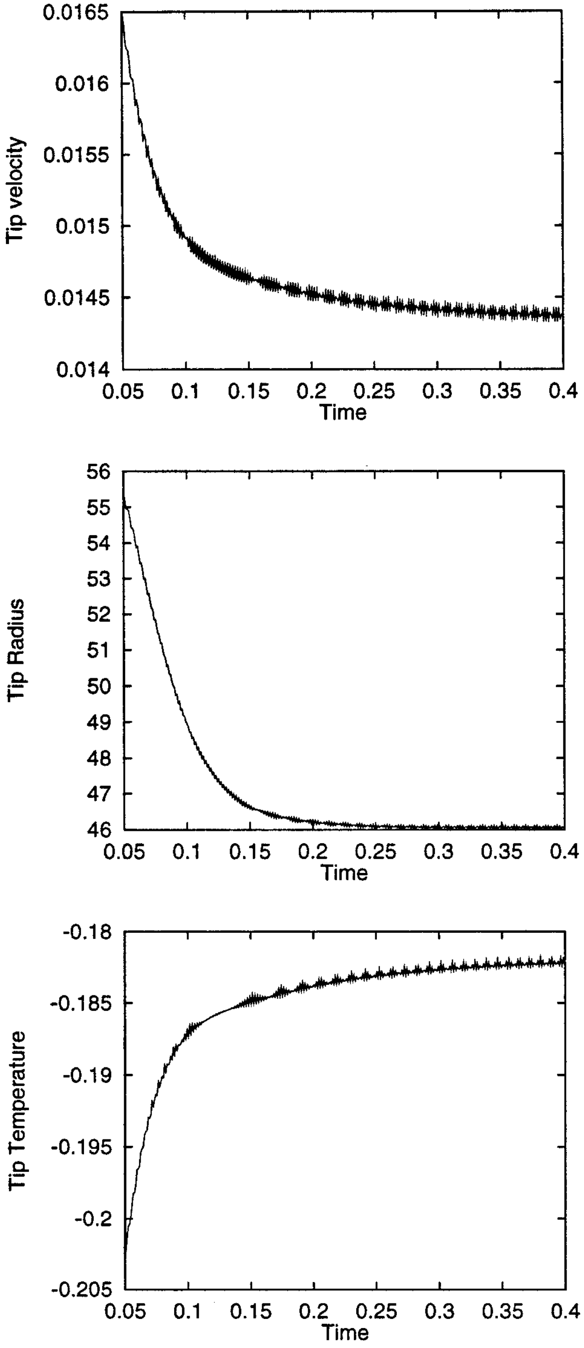


FIG. 6. Calculated values of the tip velocity, tip radius, and tip temperature as a function of time. This is for the case $S = 0.8$, $\bar{m} = 0.1$, $\delta_\gamma = \delta_\mu = 0.03$ for which the associated morphologies were shown in Fig. 5a. The units are κ/d_o for the tip velocity, d_o for the tip radius, and $1.8 \times 10^5\ d_o^2/\kappa$ for time. There are a total of 350 data points distributed at equal increments of time. As a guide to the eye, lines are drawn to connect the points.

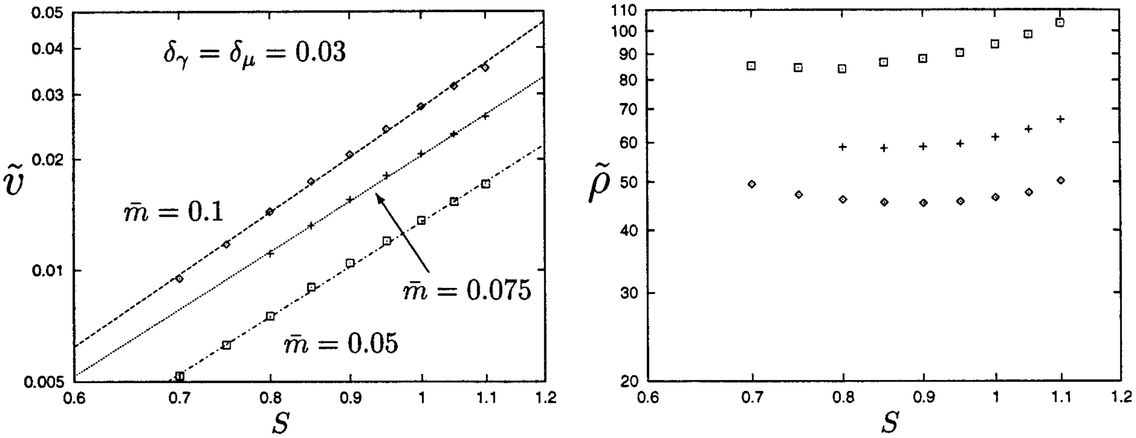


FIG. 7. Log–log plots of computed tip velocity $\tilde{v}$ and tip radius $\tilde{\rho}$ as a function of supercooling S for $\delta_\gamma = \delta_\mu = 0.03$ and three values of $\bar{m}$. The straight lines through the values of computed data points are least squares fits. Diamonds are for $\bar{m} = 0.1$, crosses for $\bar{m} = 0.075$ and squares for $\bar{m} = 0.05$.

by not attempting to update the value of the phase field in the bulk liquid and solid.

Practical computations are found to be possible only for large values of supercooling, which results in a dendrite tip radius that is large compared to the thickness of the diffuse interface but small compared to a domain size. Presumedly, one could employ adaptive moving grids to get results at lower supercoolings, but in some sense such algorithms run counter to the advantage of the phase field model, which is supposed to eliminate interface tracking. This would be a good topic for future research, but success would probably necessitate multidisciplinary cooperation between numerical analysts and materials scientists because of the complexity of the problem.

ACKNOWLEDGMENTS

The authors are grateful for discussions with B. T. Murray, A. A. Wheeler, S. R. Coriell, R. J. Braun, G. B. McFadden, and R. A. Nicolaides. This work was performed with support from the National Science Foundation under Grants DMR-9211276 and MCA94P008P. The services provided by the Pittsburgh Supercomputing Center are also gratefully acknowledged.

REFERENCES

1. J. S. Langer, private communication, August 1978.
2. J. S. Langer, in *Directions in Condensed Matter Physics*, edited by G. Grinsteil and G. Mazenko (World Scientific, Singapore, 1986), p. 165.
3. R. Kobayashi, *Bull. Jpn. Soc. Ind. Appl. Math.* **1,** 22 (1991).
4. R. Kobayashi, in *Computing Optimal Geometries*, Proceedings Amer. Math. Soc. Special Session, edited by Jean Taylor, videotapes, 1991.
5. R. Koyabashi, *Physica D* **63,** 410 (1993).
6. A. A. Wheeler, B. T. Murray, and R. J. Schaefer, *Physica D* **66,** 243 (1993).
7. B. T. Murray, W. J. Boettinger, G. B. McFadden, A. A. Wheeler, and R. F. Sekerka, in *Heat Transfer in Melting, Solidification, and Crystal Growth*, edited by I. S. Habib and S. Thynell (Am. Soc. Mech. Eng., New York, 1993), p. 67.
8. G. Caginalp and E. Socolovsky, *SIAM J. Sci. Comput.* **15,** 106 (1994).
9. R. Kupferman, O. Shochet, and E. Ben-Jacob, *Phys. Rev. E* **50,** 1005 (1994).
10. R. Koyabashi, in *Pattern Formation in Complex Dissipative Systems*, edited S. Kai (World Scientific, Singapore, 1992), p. 121, and associated video tape.
11. S.-L. Wang, R. F. Sekerka, A. A. Wheeler, B. T. Murray, S. R. Coriell, R. J. Braun, and G. B. McFadden, *Physica D* **69,** 189 (1993).
12. G. B. McFadden, A. A. Wheeler, R. J. Braun, S. R. Coriell, and R. F. Sekerka, *Phys. Rev. E* **48,** 2016 (1993).
13. S.-L. Wang and R. F. Sekerka, *Phys. Rev. E*, **53,** 3760 (1996).
14. S.-L. Wang, Doctoral Thesis, Department of Physics, Carnegie Mellon University, 1995 (unpublished).
15. J. Strikwerda, *Finite Difference Schemes and Partial Differential Equations* (Wadsworth & Brooks/Cole, Belmont, CA, 1989.
16. C. M. Elliott and R. Schätzle, Research Report No. 95/05, Center for Mathematical Analysis and Its Applications, University of Sussex, UK, 1995.
17. R. J. Braun, G. B. McFadden, and S. R. Coriell, *Phys. Rev. E* **49,** 4336 (1994).
18. G. P. Ivantsov, *Dokl. Akad. Nauk. SSSR* **58,** 567 (1947).
19. J. S. Langer and H. Müller-Krumbhaar, *J. Crystal Grwoth* **42,** 11 (1977).
20. J. S. Langer and H. Müller-Krumbhaar, *Acta Metall.* **26,** 1681, 1689, 1697 (1978).
21. D. Kessler, J. Koplik, and H. Levine, *Phys. Rev. A* **34,** 4980 (1986).
22. E. A. Brener, *Sov. Phys. JEPT* **69**(1), 133 (1989).
23. E. A. Brener and H. Levine, *Phys. Rev. A* **43,** 883 (1991).
24. B. Caroli, C. Caroli, C. Misbah, and B. Roulet, *J. Phys.* (*Paris*) **48,** 547 (1987).
25. A. Barbieri and J. B. Langer, *Phys. Rev. A* **39,** 5314 (1989).
26. S. Tanveer, *Phys. Rev. A* **40,** 4756 (1989).
27. R. Willnecker, D. M. Herlach, and B. Feuerbacher, *Phys. Rev. Lett.* **62,** 2707 (1989).