Computer Physics Communications 236 (2019) 95–103

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

Fast and accurate adaptive finite difference method for dendritic growth

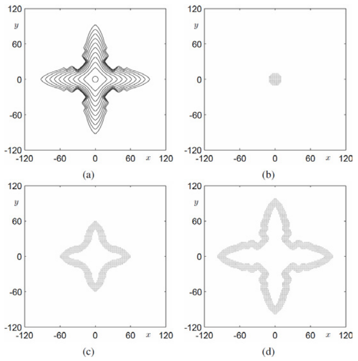
Darae Jeong[a], Junseok Kim[b,*]

[a] Department of Mathematics, Kangwon National University, Gangwon-do 24341, Republic of Korea
[b] Department of Mathematics, Korea University, Seoul 02841, Republic of Korea

HIGHLIGHTS

- Fast and accurate adaptive finite difference method is developed.
- The algorithm implementation is very simple.
- The proposed method can straightforwardly be extended to three-dimensional space.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 20 May 2017
Received in revised form 25 September 2018
Accepted 24 October 2018
Available online 2 November 2018

Keywords:
Phase-field model
Dendritic growth
Crystal morphology
Solidification
Growth from melt
Adaptive numerical method

ABSTRACT

We propose a fast and accurate adaptive numerical method for solving a phase-field model for dendritic growth. The phase-field model for dendritic growth consists of two equations. One is for capturing the interface between solid and melt. The other is for the temperature distribution. For the phase-field equation, we apply a hybrid explicit method on a time-dependent narrow-band domain, which is defined using the phase-field function. For the temperature equation, we apply the explicit Euler method on the whole computational domain. The novelties of the proposed numerical algorithm are that it is very simple and that it does not require the conventional complex adaptive data structures. Our numerical simulation results are consistent with previous results. Furthermore, the computational time required (CPU time) is shorter.

1. Introduction

The dendritic growth of crystals can be observed in nature. Many theories have been proposed to understand dendritic growth kinetics and have been compared with experimental observations [1]. Computer simulations of crystal growth processes are important in many technological applications [2]. The phase-field method is one of the most powerful and popular computational methods for modeling and simulating crystal growth from supersaturated solutions [3,4]. The convective effects on free dendritic crystal growth in a supercooled melt in two dimensions [5–10] (Fig. 1(a)) and three dimensions [11–13] (Fig. 1(b)) have been investigated using the phase-field method.

Mullis [14] studied the effects of the thermal conductivity on the side-branching characteristics of dendrites and found that high conductivity in the solid resulted in extensive side-branching. Demange et al. [15] reported simulations of snow crystal growth in three dimensions using a modified phase-field model. The simulated snowflakes showed excellent agreement with the experimental observations. Ode et al. [16] derived a phase-field model

[*] Corresponding author.
E-mail address: cfdkim@korea.ac.kr (J. Kim).
URL: http://math.korea.ac.kr/~cfdkim (J. Kim).

https://doi.org/10.1016/j.cpc.2018.10.020
0010-4655/© 2018 Elsevier B.V. All rights reserved.

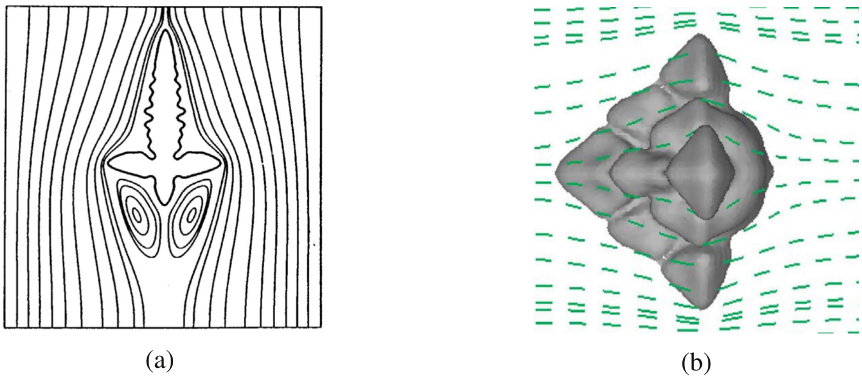


Fig. 1. Dendritic crystal growth with velocity in (a) two dimensions (reprinted from [6] with permission from the American Physical Society) and (b) three dimensions (reprinted from [11] with permission from Elsevier Science).

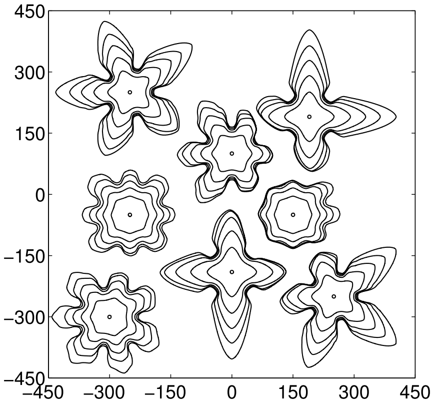


Fig. 2. The growth of eight crystals with different fold numbers and orientations.
Source: Reprinted from [17] with permission from Elsevier Science.

for ternary alloy systems and simulated dendrite growth for Fe-C-P alloys. Using the vector-valued Allen–Cahn equation, Lee and Kim [17] simulated the growth of multiple crystals with different orientation angles and fold numbers (see Fig. 2). For an extensive review of dendritic growth during solidification, see a recent review paper [18].

One of the distinguishing features of simulating dendritic growth is that it involves multi-scale phenomena. It starts from a small seed and a thin interfacial transition layer develops later. Therefore, it is essential to use an adaptive mesh to compute large-scale and long-time simulations. There have been many research works on mesh adaptivity for dendritic growth simulations. Adaptive refinement on a finite element mesh was presented in [19]. The authors in [20] used an adaptive finite-difference algorithm for the computation of dendritic growth using a phase-field model. An adaptive finite volume method was presented for two-dimensional dendritic growth in a forced flow in [9] and for three-dimensional growth [21]. The adaptive mesh refinement (AMR) technique was used in [22]. In [23], directional solidification was simulated using an adaptive finite element method, in which the isoparametric quadrilateral element was used. The morphological instability and facet formation during the directional solidification of SiGe alloys were simulated using an adaptive phase-field model in [24]. Moreover, adaptive phase-field modeling was applied to simulate the directional solidification of silicon in three-dimensional space in [25]. In [26], the authors used AMR, implicit time stepping, a nonlinear multigrid solver, and parallel computations to simulate three-dimensional binary alloy solidification. Recently, a numerical method combining AMR and parallel computing was developed to simulate dendrite growth using a phase-field-lattice Boltzmann method [27]. However, most existing adaptive mesh techniques for dendritic growth simulations are complex and not easy to implement unless one is familiar with each specific methodology. For example, Fig. 3 shows snapshots of the three-dimensional evolution of crystal growth including the complex bounding boxes using AMR [22]. Therefore, the aim of this study is to develop a simpler adaptive numerical scheme, which is also fast and accurate, for solving phase-field models for dendritic growth.

This paper is organized as follows. In Section 2, we present a phase-field model for dendritic growth. In Section 3, we describe the proposed numerical algorithm. To demonstrate the performance of the proposed numerical scheme, we present several computational results of dendritic growth in Section 4. In Section 5, conclusions are drawn.

2. Phase-field model

We consider the dendritic growth of a pure substance from its supercooled melt in two- and three-dimensional spaces. In our phase-field model, we use an order parameter $\phi(\mathbf{x},t)$ that is -1 in the liquid phase and 1 in the solid phase. Here, $\mathbf{x}$ and t denote the spatial position in the Ω domain and time, respectively. The interface of the crystal is defined by the zero-level set, i.e., $\Gamma = \{\mathbf{x}\in\Omega|\phi(\mathbf{x},t)=0\}$. The governing equations in two-dimensional space are given as

$$\epsilon^2(\phi)\frac{\partial\phi}{\partial t} = \nabla\cdot(\epsilon^2(\phi)\nabla\phi) + [\phi - \lambda U(1-\phi^2)](1-\phi^2) + \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_x}\right)_x + \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_y, \quad (1)$$

$$\frac{\partial U}{\partial t} = D\Delta U + \frac{1}{2}\frac{\partial\phi}{\partial t}, \quad \text{for } \mathbf{x}\in\Omega,\ t>0, \quad (2)$$

where W_0 is a measure of the interface width [3] and $\epsilon(\phi)$ is the anisotropic function. The four-fold symmetry is defined as

$$\epsilon(\phi) = W_0\left[(1-3\delta_4) + 4\delta_4(\phi_x^4+\phi_y^4)/|\nabla\phi|^4\right], \quad (3)$$

where δ_4 is the anisotropic strength, $\phi_x = \partial\phi/\partial x$, and $\phi_y = \partial\phi/\partial y$. Additionally, λ is the dimensionless coupling parameter and $U(\mathbf{x},t)$ is the dimensionless temperature field. The three-dimensional model will be described in Section 4.6. More details about phase-field modeling for dendritic growth and its parameters can be found in [28].

3. Numerical solution

In this section, we present an adaptive explicit hybrid numerical scheme for Eqs. (1) and (2) in detail. Let $\Omega = (a,b)\times(c,d)$ be the spatial domain for computation and $h = (b-a)/N_x = (d-c)/N_y$ be

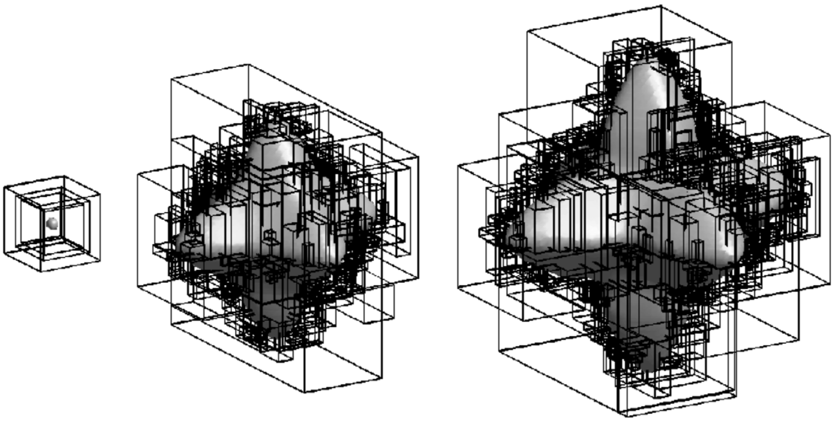


Fig. 3. Snapshots of the three-dimensional evolution of crystal growth including the bounding boxes using adaptive mesh refinement.
Source: Reprinted from [22] with permission from Elsevier Science.

the uniform spacing, where N_x and N_y are positive integers. Then, we define the discrete domain as $\Omega_h = \{(x_i, y_j)|x_i = a + ih, y_j = c + jh \text{ for } 0 \le i \le N_x,\ 0 \le j \le N_y\}$. Let ϕ_{ij}^n and U_{ij}^n be the numerical approximations of $\phi(x_i, y_j, n\Delta t)$ and $U(x_i, y_j, n\Delta t)$, respectively. Here, the time step is given as $\Delta t = T/N_t$, where T is the final time and N_t is the number of time steps. In the proposed numerical algorithm, we only treat the numerical solution of Eq. (1) in a discrete narrow-band domain Ω_{nb}, not in the whole discrete domain Ω_h. To illustrate how to define this narrow-band domain, let us consider a circular crystal shape with a radius of 30, i.e.,

$$\phi_{ij} = \tanh \frac{30 - \sqrt{x_i^2 + y_j^2}}{\sqrt{2}}, \tag{4}$$

as shown in Fig. 4(a). Then, we define a temporary narrow domain as $\Omega_{\text{tmp}} = \{(x_i, y_j)|\ |\phi_{ij}| \le \gamma,\ 0 \le i \le N_x,\ 0 \le j \le N_y\}$. This temporary narrow domain Ω_{tmp} is illustrated in Fig. 4(b) with $\gamma = 0.99$. In this figure, the contour line is the zero-level set of ϕ. Alternatively, the temporary narrow domain can be defined by the gradient of the phase-field variable; that is, $|\nabla\phi| > \xi$, where ξ is a threshold. Then, we can extend the temporary narrow domain with buffer points m to

$$\Omega_{\text{nb}} \cup \partial\Omega_{\text{nb}} = \bigcup_{p=-m}^{p=m} \bigcup_{q=-m}^{q=m} \left\{(x_{i+p}, y_{j+q})|(x_i, y_j) \in \Omega_{\text{tmp}}\right\}. \tag{5}$$

In Fig. 4(c), we can see the extended narrow-band domain, Ω_{nb} (open circles) and its boundary points, $\partial\Omega_{\text{nb}}$ (filled circles) with $m = 2$.

In this paper, unless otherwise specified, we will use $\gamma = 0.99$ and $m = 2$, which are large enough for obtaining accurate numerical results. In Section 4.3, we will investigate the effects of changing the m and γ values. Now, we use the following operator-splitting scheme [29] to solve Eq. (1):

$$\begin{aligned}\epsilon^2(\phi)\frac{\partial\phi}{\partial t} &= \nabla\cdot(\epsilon^2(\phi)\nabla\phi) - \lambda U(1-\phi^2)^2 \\ &\quad + \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_x}\right)_x + \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_y,\end{aligned} \tag{6}$$

$$\epsilon^2(\phi)\frac{\partial\phi}{\partial t} = \phi(1-\phi^2). \tag{7}$$

First, we solve Eq. (6) in the narrow-band domain, Ω_{nb}, by using the explicit Euler method:

$$\begin{aligned}\epsilon^2(\phi_{ij}^n)\frac{\phi_{ij}^* - \phi_{ij}^n}{\Delta t} &= [\nabla\cdot(\epsilon^2(\phi)\nabla\phi)]_{ij}^n - \lambda U_{ij}^n(1-(\phi_{ij}^n)^2)^2 \\ &\quad + \left[\left(\frac{16\delta_4\epsilon(\phi)\phi_x(\phi_x^2\phi_y^2 - \phi_y^4)}{|\nabla\phi|^4}\right)_x\right]_{ij}^n \\ &\quad + \left[\left(\frac{16\delta_4\epsilon(\phi)\phi_y(\phi_x^2\phi_y^2 - \phi_x^4)}{|\nabla\phi|^4}\right)_y\right]_{ij}^n,\end{aligned} \tag{8}$$

where

$$\begin{aligned}&[\nabla\cdot(\epsilon^2(\phi)\nabla\phi)]_{ij} \\ &= \frac{\epsilon^2(\phi_{i+1,j}) + \epsilon^2(\phi_{ij})}{2h^2}(\phi_{i+1,j} - \phi_{ij}) \\ &- \frac{\epsilon^2(\phi_{ij}) + \epsilon^2(\phi_{i-1,j})}{2h^2}(\phi_{ij} - \phi_{i-1,j}) \\ &+ \frac{\epsilon^2(\phi_{i,j+1}) + \epsilon^2(\phi_{ij})}{2h^2}(\phi_{i,j+1} - \phi_{ij}) \\ &- \frac{\epsilon^2(\phi_{ij}) + \epsilon^2(\phi_{i,j-1})}{2h^2}(\phi_{ij} - \phi_{i,j-1}).\end{aligned}$$

For the last two terms in Eq. (8), we apply a centered finite difference scheme as stated above. When necessary, we use the values on the boundary points, $\partial\Omega_{\text{nb}}$. Because Eq. (7) is an ordinary differential equation, we can solve it analytically; the closed-form solution is given as

$$\phi_{ij}^{n+1} = \phi_{ij}^* \Bigg/ \sqrt{e^{-\frac{2\Delta t}{\epsilon^2(\phi_{ij}^n)}} + (\phi_{ij}^*)^2\left(1 - e^{-\frac{2\Delta t}{\epsilon^2(\phi_{ij}^n)}}\right)}. \tag{9}$$

Next, we use the explicit Euler method on the whole domain, Ω_h for Eq. (2):

$$\frac{U_{ij}^{n+1} - U_{ij}^n}{\Delta t} = D\Delta_h U_{ij}^n + \frac{\phi_{ij}^{n+1} - \phi_{ij}^n}{2\Delta t}, \tag{10}$$

where we used the standard five point stencil for the discrete Laplacian operator, i.e., $\Delta_h U_{ij} = (U_{i-1,j} + U_{i+1,j} - 4U_{ij} + U_{i,j-1} + U_{i,j+1})/h^2$.

4. Numerical experiments

Unless otherwise stated, we use the following initial conditions:

$$\phi(x, y, 0) = \tanh\left(\frac{R_0 - \sqrt{x^2 + y^2}}{\sqrt{2}}\right), \tag{11}$$

$$U(x, y, 0) = \begin{cases} 0 & \text{if } \phi > 0, \\ \Delta & \text{otherwise} \end{cases} \tag{12}$$

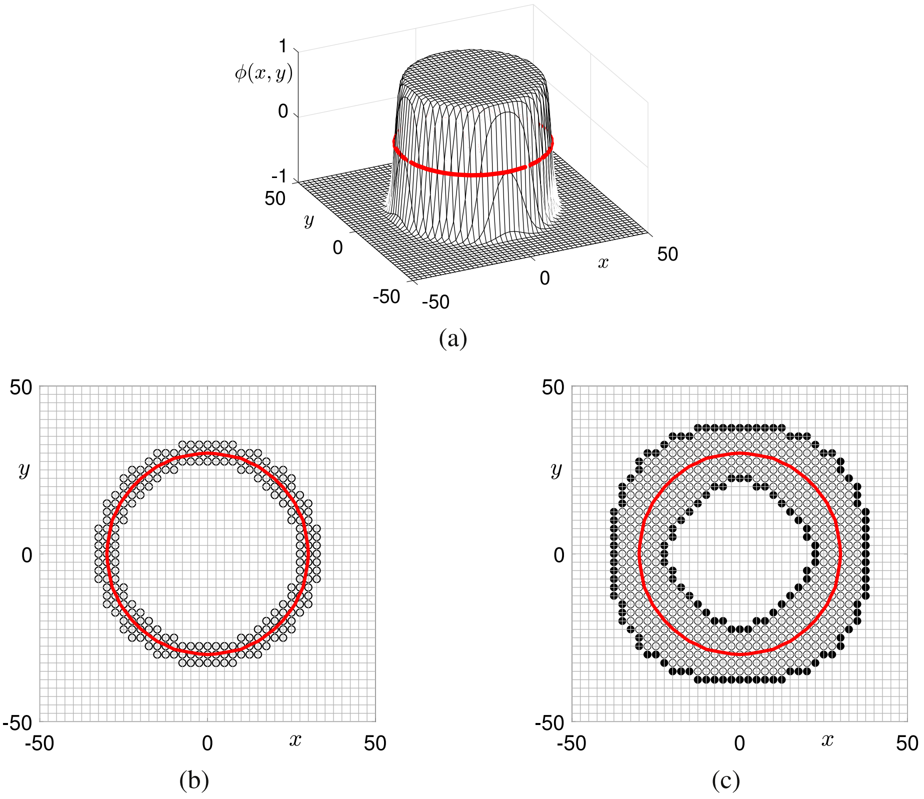


Fig. 4. (a) Phase-field function, which is defined as $\phi_{ij} = \tanh((30 - \sqrt{x_i^2 + y_j^2})/\sqrt{2})$. (b) Temporary domain, Ω_{tmp} (open circles). (c) Narrow-band domain, Ω_{nb} (open circles), and its boundary points, $\partial\Omega_{\text{nb}}$ (filled circles). The contour line is the zero-level set of the phase-field function, $\phi(x, y)$.

with Dirichlet boundary conditions, i.e., $\phi(x, y, t) = -1$ and $U(x, y, t) = \Delta$ on $\partial\Omega$ for all t. Here, R_0 is the initial radius. and Δ is the undercooling parameter. The coupling parameter is given as $\lambda = D/0.6267$ [3]. All the programs are executed on a standard desktop PC with an Intel 3.2 GHz CPU.

4.1. Temporal evolution on narrow-band domain

As our first numerical experiment, we perform a standard dendritic growth simulation to see the temporal evolution of the interface and their corresponding narrow-band domains. Fig. 5(a) shows the temporal evolution of the dendritic interface until time $T = 10000\Delta t$. Here, we take an initial radius $R_0 = 5$ in Eq. (11), $\Delta = -0.55$ in Eq. (12), $\delta_4 = 0.05$, $D = 2$, $N_x = N_y = 300$, $h = 0.8$, and $\Delta t = 0.1h^2$ on the computational domain $\Omega = (-120, 120) \times (-120, 120)$. Fig. 5(b)–(d) represent their corresponding narrow domain at times $t = 0$, $5000\Delta t$, and $10000\Delta t$, respectively. In these figures, we can see the temporally moving narrow-band domain according to the interface location of the phase-field function.

4.2. Dimensionless steady-state tip velocity

In this section, we demonstrate the accuracy of our proposed method by calculating the dimensionless steady-state tip velocities obtained from it. We then compare our results with those from [22,28,29]. Here, we denote by $V_{\text{tip}}^{\text{KR}}$, $V_{\text{tip}}^{\text{LLK}}$, $V_{\text{tip}}^{\text{LK}}$, and $V_{\text{tip}}^{\text{GF}}$ the results from Karma and Rappel [28], Li et al. [29], Li and Kim [22], and Green's function calculations, respectively.

For comparison with [22,28,29], we carry out the numerical simulations with $h = 0.4$, $\Delta t = 0.15$, $R_0 = 3.462$, and $W_0 = 1$ on the computational domain $\Omega = (-200, 200)^2$. Furthermore, we use a quadratic polynomial approximation for the steady-state velocity. Because of the symmetric shape of crystals, we only describe the procedure along y-axis. Let y_k be the farthest position on the interface of the crystal away from the origin. Then, we can calculate the quadratic polynomial approximation $y = \alpha x^2 + \beta x + \gamma$ passing through points (x_{k-1}, y_{k-1}), (x_k, y_k), and (x_{k+1}, y_{k+1}). Using this polynomial, we obtain the tip position y_*, which satisfies $y'(x_*) = 0$ and $y_* = \alpha x_*^2 + \beta x_* + \gamma$. Using these tip positions, we can compute the crystal tip velocity. From the set of numerical results presented in Table 1, we can see that values obtained using our proposed numerical scheme are in good agreement with the results of the previous methods over the range of Δ, δ_4, and d_0/W_0 values, where d_0 is the capillary length [6].

Table 1
Comparison of the dimensionless steady-state tip velocities calculated using the proposed scheme ($V_{\text{tip}} = V \cdot d_0/D$), $V_{\text{tip}}^{\text{LLK}}$ [29], $V_{\text{tip}}^{\text{KR}}$ [28], $V_{\text{tip}}^{\text{LK}}$ [22], and Green's function calculations $V_{\text{tip}}^{\text{GF}}$ [28]. Here, $\delta_4 = 0.05$ is used.

Δ	D	d_0/W_0	$V_{\text{tip}}^{\text{LLK}}$	$V_{\text{tip}}^{\text{KR}}$	$V_{\text{tip}}^{\text{LK}}$	$V_{\text{tip}}^{\text{GF}}$	V_{tip}
−0.55	2	0.277	0.0171	0.0168	0.0170	0.0170	0.0173
−0.55	3	0.185	0.0174	0.0175	0.0172	0.0170	0.0170
−0.50	3	0.185	0.0103	0.0101	0.0100	0.0099	0.0103

4.3. Verification of the narrow-band domains Ω_{nb}

Now, we investigate the effects of γ and m on the computational results. To do this, we calculate the dimensionless steady-state tip velocity of the crystal on various narrow-band domains. Here, we use the following parameters: $\delta_4 = 0.05$, $\Delta = -0.55$, $D = 3$, $d_0 = 0.185$, $R_0 = 3.462$, $W_0 = 1$, $h = 0.4$, and $\Delta t = 0.2h^2/D$ on the computational domain $\Omega = (-200, 200)^2$.

Fig. 6 shows the obtained dimensionless steady-state tip velocities with respect to Ω_{nb}. Here, solid lines with ○, △, and □ markers denote the results calculated on the adaptive mesh Ω_{nb} with $\gamma =$

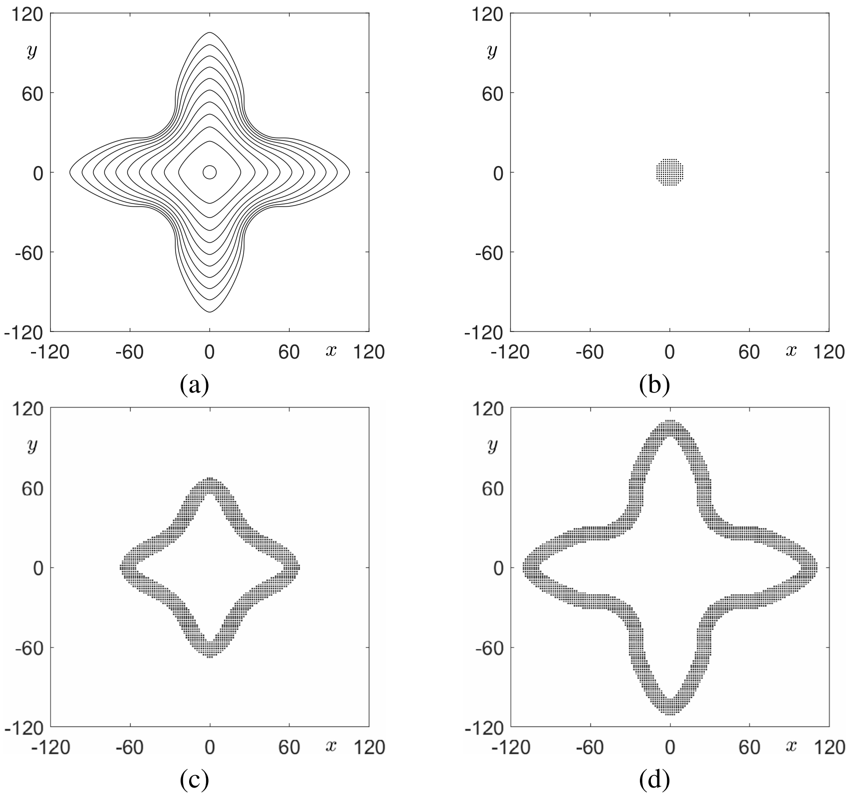


Fig. 5. (a) Temporal evolution of the dendritic interface up to time $T = 10000\Delta t$. (b)–(d) Narrow domains Ω_{nb} used at time $t = 0, 5000\Delta t$, and $10000\Delta t$, respectively. For better visualization, only a quarter of all points are plotted.

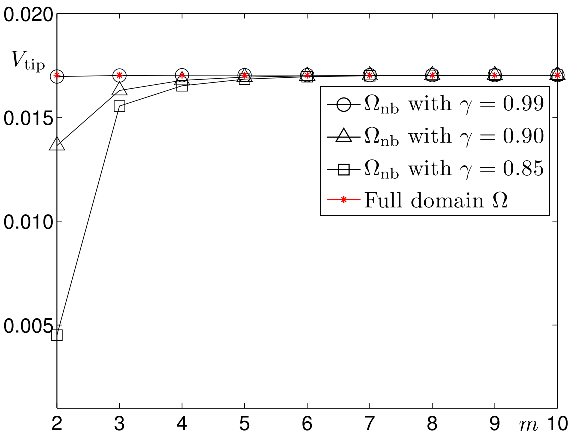


Fig. 6. Comparison of the dimensionless steady-state tip velocity V_{tip} of crystal growth with respect to m and γ.

0.99, 0.90, and 0.85, respectively. Additionally, for comparison, we add the result obtained on the full discrete domain Ω_h (see the solid line with $\star$ marker). As shown in Fig. 6, excessively small values of both γ and m do not generate good results because the band becomes too narrow. However, when using large enough γ or m values, the results in Ω_{nb} are in good agreement with the ones in the full domain Ω_h. Therefore, to obtain efficient and accurate numerical results, the choice of $\gamma = 0.99$ and $m = 2$ is appropriate.

4.4. Effects of parameters

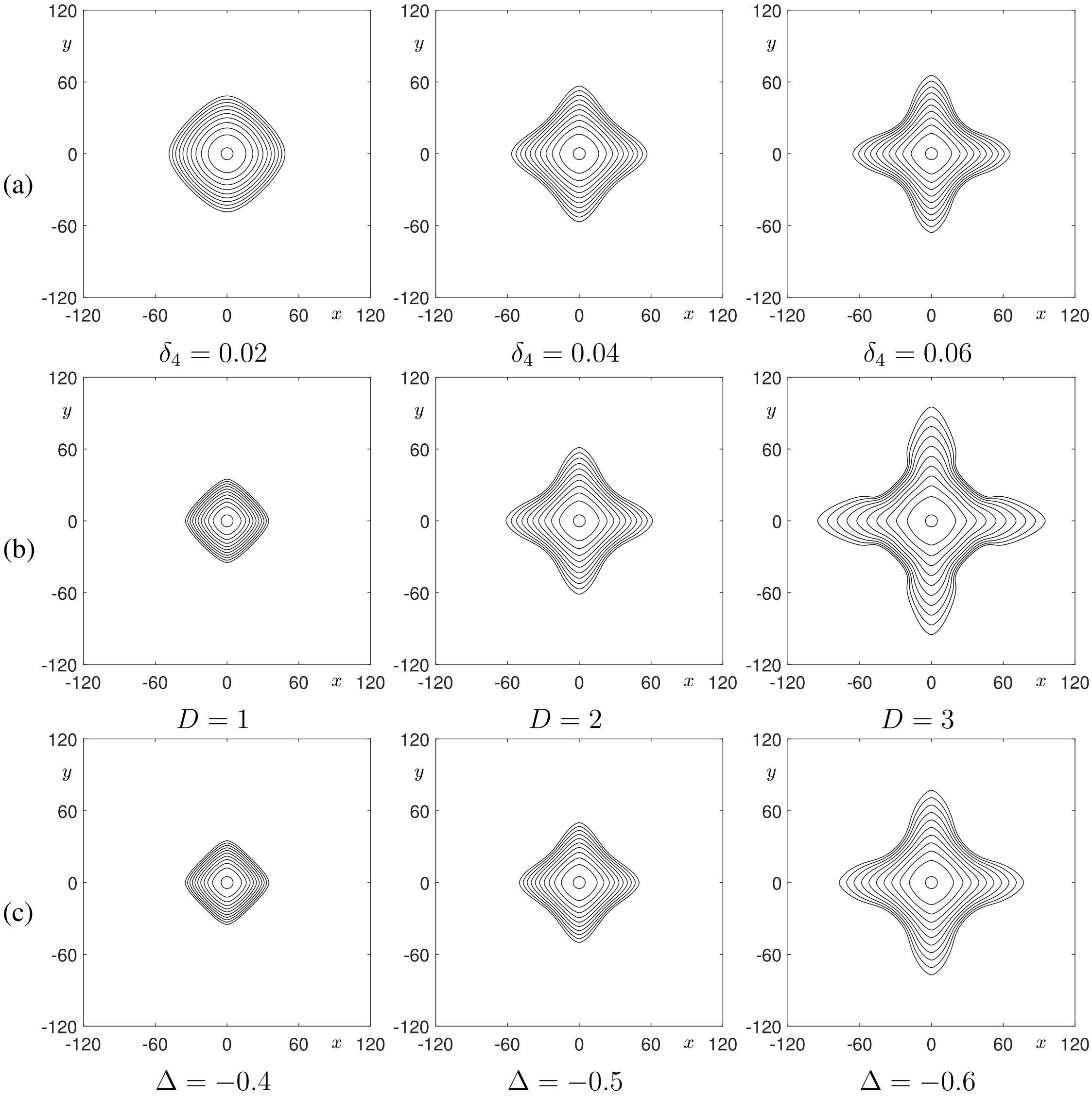
In this section, we investigate the effects of certain parameters, such as δ_4, D, and Δ, on dendritic growth dynamics. For these tests, we use the following parameters: $R_0 = 5, N_x = N_y = 300, h = 0.8$, $\Delta t = 0.05h^2$, and $T = 10000\Delta t$ on the computational domain $\Omega = (-120, 120)^2$.

Fig. 7 shows the temporal evolutions of the dendritic interface until time $T = 10000\Delta t$ for different δ_4, D, and Δ values. As shown in Fig. 7(a), increasing δ_4, which is the strength of anisotropy, promotes more facets on the crystal. In this case, we use $\Delta = -0.55$ and $D = 2$. Moreover, when we use $\Delta = -0.55$, $\delta_4 = 0.05$, and various values of D, we obtain the numerical results shown in Fig. 7(b). Increasing the diffusion coefficient increases the heat transfer from the interface and results in the fast growth of the crystal. Lastly, we can see the effects of Δ in Fig. 7(c). Decreasing the undercooling parameter Δ results in the faster growth of the crystal. In this case, we use the following parameters: $D = 2$ and $\delta_4 = 0.05$.

4.5. Computational time

Fig. 8(a), (b), and (c) show the temporal evolution of the dendritic interface until time $T = 76000\Delta t$, the narrow domain at time $76000\Delta t$, and a close-up view of a part of the narrow domain, respectively. The following problem parameters are used: $R_0 = 5$, $D = 2$, $\delta_4 = 0.05$, $N_x = N_y = 1500$, $h = 0.8$, and $\Delta t = 0.1h^2$. Calculating these results took only 1 h 40 m of CPU time. The implicit scheme took approximately 50 h for the same simulation.

Now, to show the efficiency of the proposed scheme, we compare the computational times obtained with our scheme and the implicit hybrid scheme [29]. In this test, we us the same parameters as those defined in Section 4.3. In the top row in Fig. 9, we can see that the proposed scheme is faster than the implicit scheme. The bottom row in Fig. 9 shows the temporal evolution of crystal growth obtained via the proposed scheme and the implicit hybrid scheme [29].



$\delta_4 = 0.02$ $\delta_4 = 0.04$ $\delta_4 = 0.06$

$D = 1$ $D = 2$ $D = 3$

$\Delta = -0.4$ $\Delta = -0.5$ $\Delta = -0.6$

Fig. 7. Temporal evolutions of the dendritic interface until time $T = 10000\Delta t$ for different values of (a) δ_4, (b) D, and (c) Δ.

4.6. Three-dimensional dendritic growth

The proposed adaptive algorithm can be straightforwardly extended to three-dimensional space. In this section, we consider dendritic growth on a three-dimensional domain using the proposed algorithm. Let $\Omega = (a, b) \times (c, d) \times (e, f)$ be the computational domain and $h = (b-a)/N_x = (d-c)/N_y = (f-e)/N_z$ be the uniform mesh size, where N_x, N_y, and N_z are positive integers. Then, $\Omega_h = \{(x_i, y_j, z_k)|x_i = a + ih, y_j = c + jh, z_k = e + kh \text{ for } 0 \le i \le N_x,\ 0 \le j \le N_y,\ 0 \le k \le N_z\}$ is the discrete domain. Let ϕ^n_{ijk} and U^n_{ijk} be approximations of $\phi(x_i, y_j, z_k, n\Delta t)$ and $U(x_i, y_j, z_k, n\Delta t)$, respectively. We define a temporary narrow domain as $\Omega_{\text{tmp}} = \{(x_i, y_j, z_k)|\ |\phi_{ijk}| \le 0.99,\ 0 \le i \le N_x,\ 0 \le j \le N_y,\ 0 \le k \le N_z\}$. Then, we extend this temporary narrow domain to

$$\Omega_{\text{nb}} \cup \partial\Omega_{\text{nb}} = \bigcup_{p=-m}^{p=m} \bigcup_{q=-m}^{q=m} \bigcup_{r=-m}^{r=m} \left\{(x_{i+p}, y_{j+q}, z_{k+r})|(x_i, y_j, z_k) \in \Omega_{\text{tmp}}\right\}. \tag{13}$$

The governing equations for dendritic growth in three-dimensional space are given as

$$\epsilon^2(\phi)\frac{\partial \phi}{\partial t} = \nabla \cdot (\epsilon^2(\phi)\nabla\phi) + [\phi - \lambda U(1-\phi^2)](1-\phi^2) \tag{14}$$

$$+ \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_x}\right)_x + \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_y$$

$$+ \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_z}\right)_z,$$

$$\frac{\partial U}{\partial t} = D\Delta U + \frac{1}{2}\frac{\partial\phi}{\partial t}, \tag{15}$$

where $\epsilon(\phi) = (1 - 3\delta_4) + 4\delta_4(\phi_x^4 + \phi_y^4 + \phi_z^4)/|\nabla\phi|^4$ (see [28] for more details about the three-dimensional phase-field model). The numerical solutions for the three-dimensional problem are as follows. First, we solve one part of the splitting scheme in the narrow-band domain, Ω_{nb}.

$$\epsilon^2(\phi^n_{ijk})\frac{\phi^*_{ijk} - \phi^n_{ijk}}{\Delta t} = [\nabla \cdot (\epsilon^2(\phi)\nabla\phi)]^n_{ijk} - \lambda U^n_{ijk}(1 - (\phi^n_{ijk})^2)^2 \tag{16}$$

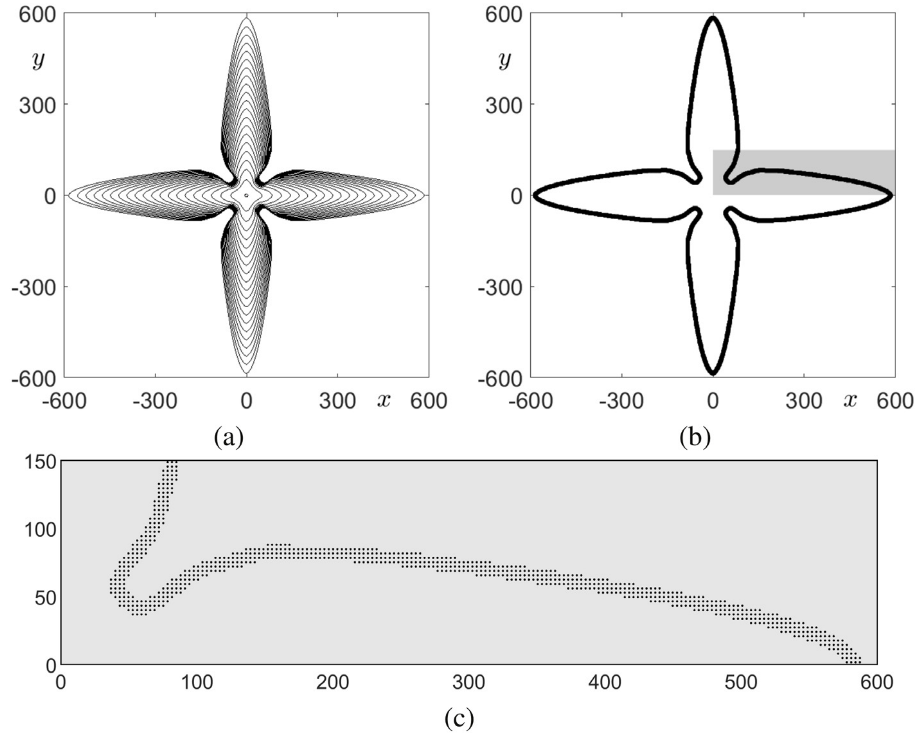


Fig. 8. (a) Temporal evolution of the dendritic interface until time $T = 76000\Delta t$. (b) Narrow domain at time $T = 76000\Delta t$. (c) Close-up view of a gray-colored region of (b). For better visualization, only half of the points are plotted.

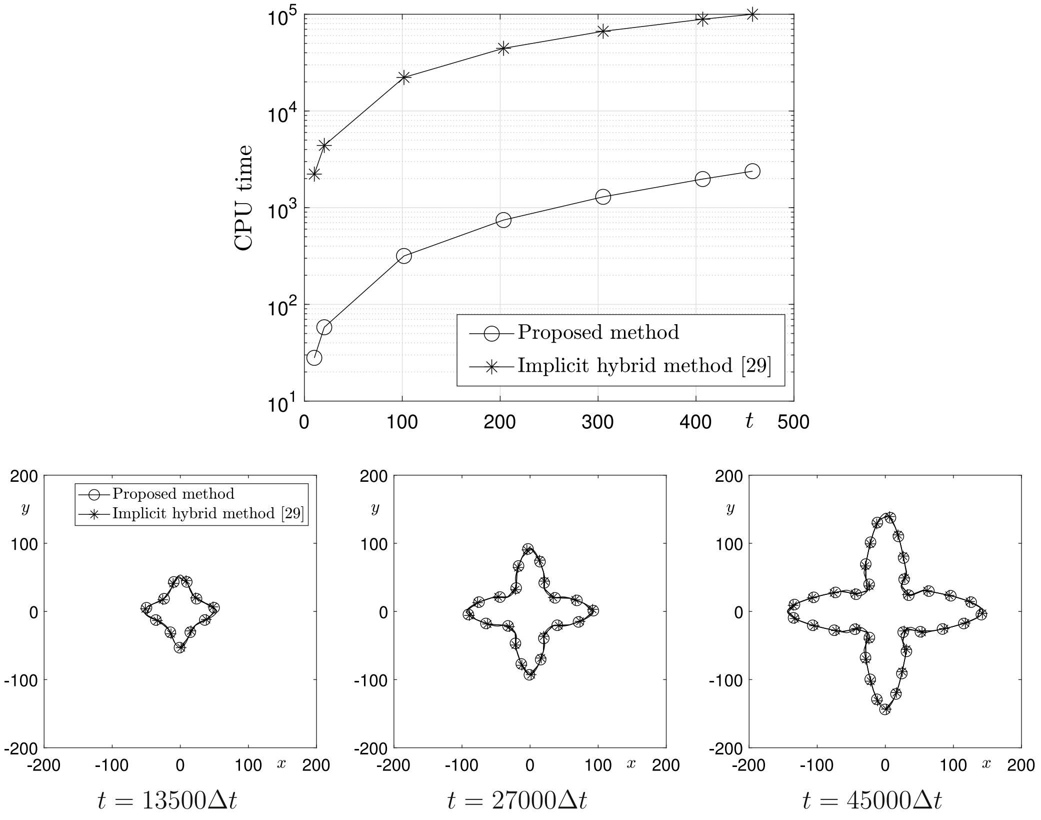


Fig. 9. First row shows a comparison of the computational times for crystal growth using the proposed scheme and the implicit hybrid scheme [29]. Second row shows the temporal evolution of crystal growth obtained via the proposed scheme and the implicit hybrid scheme [29].

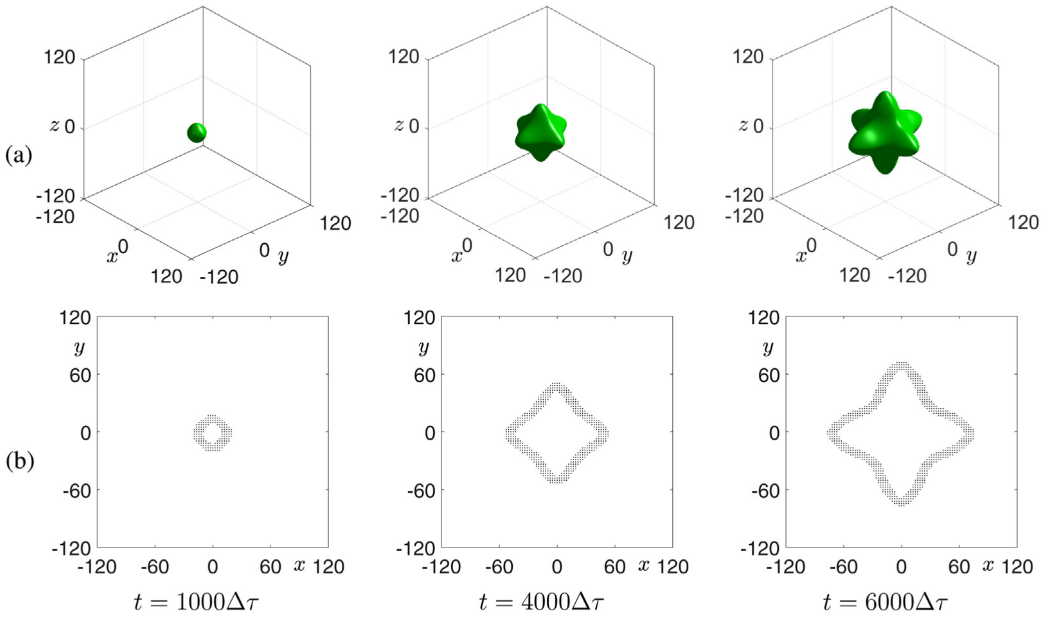


Fig. 10. (a) Isosurfaces of the numerical solution at level $\phi(x, y, z) = 0$ and (b) cross sections of the narrow-band domain.

$$+\left[\left(\frac{16\delta_4\epsilon(\phi)\phi_x(\phi_x^2\phi_y^2+\phi_x^2\phi_z^2-\phi_y^4-\phi_z^4)}{|\nabla\phi|^4}\right)_x\right]^n_{ijk}$$
$$+\left[\left(\frac{16\delta_4\epsilon(\phi)\phi_y(\phi_y^2\phi_x^2+\phi_y^2\phi_z^2-\phi_x^4-\phi_z^4)}{|\nabla\phi|^4}\right)_y\right]^n_{ijk}$$
$$+\left[\left(\frac{16\delta_4\epsilon(\phi)\phi_z(\phi_z^2\phi_x^2+\phi_z^2\phi_y^2-\phi_x^4-\phi_y^4)}{|\nabla\phi|^4}\right)_z\right]^n_{ijk},$$

where

$$\begin{aligned}
&[\nabla\cdot(\epsilon^2(\phi)\nabla\phi)]_{ijk}\\
&=\frac{\epsilon^2(\phi_{i+1,jk})+\epsilon^2(\phi_{ijk})}{2h^2}(\phi_{i+1,jk}-\phi_{ijk})\\
&-\frac{\epsilon^2(\phi_{ijk})+\epsilon^2(\phi_{i-1,jk})}{2h^2}(\phi_{ijk}-\phi_{i-1,jk})\\
&+\frac{\epsilon^2(\phi_{i,j+1,k})+\epsilon^2(\phi_{ijk})}{2h^2}(\phi_{i,j+1,k}-\phi_{ijk})\\
&-\frac{\epsilon^2(\phi_{ijk})+\epsilon^2(\phi_{i,j-1,k})}{2h^2}(\phi_{ijk}-\phi_{i,j-1})\\
&+\frac{\epsilon^2(\phi_{ijk+1})+\epsilon^2(\phi_{ijk})}{2h^2}(\phi_{ijk+1}-\phi_{ijk})\\
&-\frac{\epsilon^2(\phi_{ijk})+\epsilon^2(\phi_{ij,k-1})}{2h^2}(\phi_{ijk}-\phi_{ij,k-1}).
\end{aligned}$$

Next, we update the phase-field using the closed-form solution:

$$\phi_{ijk}^{n+1}=\phi_{ijk}^*\Bigg/\sqrt{e^{-\frac{2\Delta t}{\epsilon^2\left(\phi_{ijk}^n\right)}}+(\phi_{ijk}^*)^2\left(1-e^{-\frac{2\Delta t}{\epsilon^2\left(\phi_{ijk}^n\right)}}\right)}. \qquad (17)$$

Finally, we use the explicit Euler method on Ω_h for Eq. (15):

$$\frac{U_{ijk}^{n+1}-U_{ijk}^n}{\Delta t}=D\Delta_h U_{ijk}^n+\frac{\phi_{ijk}^{n+1}-\phi_{ijk}^n}{2\Delta t}, \qquad (18)$$

where we used the standard seven point stencil for the discrete Laplacian operator, i.e., $\Delta_h U_{ijk} = (U_{i-1,jk} + U_{i+1,jk} + U_{i,j-1,k} + U_{i,j+1,k} + U_{ij,k-1} + U_{ij,k+1} - 6U_{ijk})/h^2$.

We use the following initial conditions:

$$\phi(x,y,z,0)=\tanh\left(\frac{1-\sqrt{x^2+y^2+z^2}}{\sqrt{2}}\right) \text{ and}$$

$$U(x,y,0)=\begin{cases}0 & \text{if } \phi>0\\ \Delta & \text{otherwise}\end{cases}$$

with Dirichlet boundary conditions, i.e., $\phi(x, y, z, t) = -1$ and $U(x, y, z, t) = \Delta$ on $\partial\Omega$. The parameters are used as $D = 2$, $\Delta = -0.55$, $\delta = 0.05$, and the coupling parameter $\lambda = D/0.6267$ [3]. Fig. 10(a) shows the temporal evolution of the crystal at times $t = 0, 1000\Delta t, 4000\Delta t$, and $6000\Delta t$. The numerical results shown in Fig. 10(a) are calculated using the narrow-band domains as shown in Fig. 10(b).

4.6.1. *Effects of parameters*

We investigate the effects of parameter δ_4 on crystal growth dynamics. In these tests, we use the following parameters: $N_x = N_y = N_z = 300$, $h = 0.8$, $\Delta t = 0.025h^2$, and $T = 7000\Delta t$.

Fig. 11(a)–(c) show the effect of various δ_4, D, and Δ values, respectively. We use the following parameters: (a) $\Delta = -0.55$ and $D = 2$, (b) $\Delta = -0.55$ and $\delta_4 = 0.05$, (c) $D = 2$ and $\delta_4 = 0.05$. From the test results shown in Fig. 11, we can see that as δ_4 and D increase and Δ decreases, the tip velocity increases.

5. Conclusions

In this paper, we developed a fast and accurate adaptive finite difference method for solving a phase-field model for dendritic growth from an undercooled melt. The proposed hybrid numerical algorithm is very simple and does not use conventional complex adaptive data structures. Our various numerical simulation results were consistent with previous results and were obtained using less CPU time. Finally, extending the present work toward graphic processing units (GPUs) [30,31], the effects of the applied temperature gradient [32], fluid flows [10,33], and parallel computing [34,35] are interesting near-future research directions.

Acknowledgments

The authors thank the reviewers for their constructive and helpful comments on the revision of this article. The first author

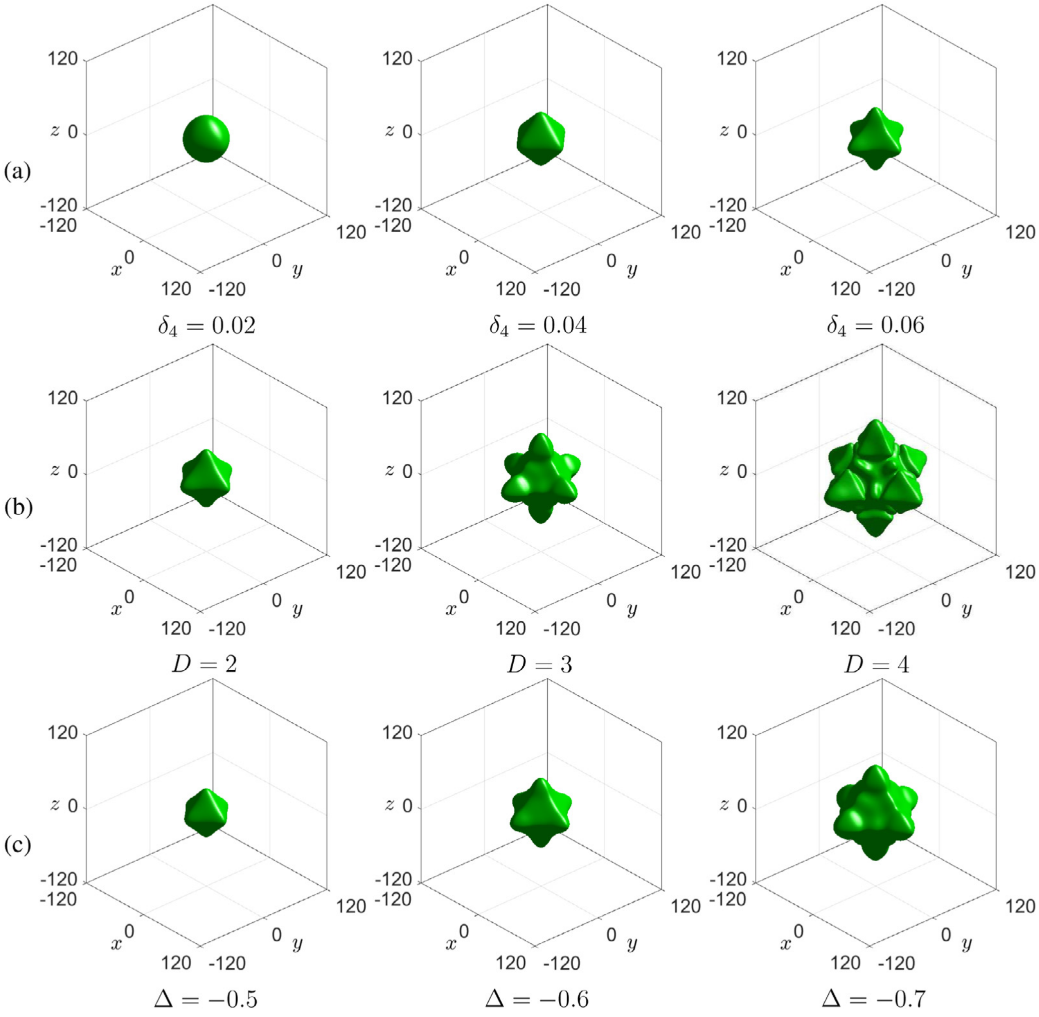


Fig. 11. Numerical solutions of the dendritic interface at $T = 7000\Delta t$ for different values of (a) δ_4, (b) D, and (c) Δ.

(D. Jeong) was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2017R1E1A1A03070953). The corresponding author (J.S. Kim) was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1A1B03933243).

Conflicts of interest

We declare that we have no conflict of interest.

References

[1] J. Gao, M. Han, A. Kao, K. Pericleous, D.V. Alexandrov, P.K. Galenko, Acta Mater. 103 (2016) 184–191.
[2] L. Tan, N. Zabaras, J. Comput. Phys. 221 (1) (2007) 9–40.
[3] J. Rosam, P.K. Jimack, A. Mullis, J. Comput. Phys. 225 (2) (2007) 1271–1287.
[4] D. Tourret, A. Karma, Acta Mater. 82 (2015) 64–83.
[5] Z. Guo, J. Mi, S. Xiong, P.S. Grant, J. Comput. Phys. 257 (2014) 278–297.
[6] X. Tong, C. Beckermann, A. Karma, Q. Li, Phys. Rev. E 63 (6) (2001) 061601.
[7] C.W. Lan, C.M. Hsu, C.C. Liu, Y.C. Chang, Phys. Rev. E 65 (6) (2002) 061601.
[8] C.-W. Lan, C.-J. Shih, J. Cryst. Growth 264 (1–3) (2004) 472–482.
[9] C.W. Lan, C.M. Hsu, C.C. Liu, J. Cryst. Growth 241 (3) (2002) 379–386.
[10] S. Lee, Y. Li, J. Shin, J.S. Kim, Comput. Phys. Comm. 216 (2017) 84–94.
[11] Y. Lu, C. Beckermann, J.C. Ramirez, J. Cryst. Growth 280 (1–2) (2005) 320–334.
[12] C.C. Chen, C.W. Lan, J. Cryst. Growth 312 (8) (2010) 1437–1442.
[13] J.H. Jeong, N. Goldenfeld, J.A. Dantzig, Phys. Rev. E 64 (4) (2001) 041602.
[14] A.M. Mullis, Comput. Mater. Sci. 38 (2006) 426–431.
[15] G. Demange, H. Zapolsky, R. Patte, M. Brunel, NPJ Comput. Mater. 3 (1) (2017) 15;1–7.
[16] M. Ode, J.S. Lee, T. Suzuki, S.G. Kim, W.T. Kim ISIJ Int. 40 (9) (2000) 870–876.
[17] H.G. Lee, J. Kim, Comput. Phys. Comm. 183 (10) (2012) 2107–2115.
[18] M.A. Jaafar, D.R. Rousse, S. Gibout, J.P. Bedecarrats, Renew. Sust. Energ. Rev. 74 (2017) 1064–1079.
[19] N. Provatas, G. Nigel, D. Jonathan, J. Comput. Phys. 148 (1) (1999) 265–290.
[20] R.J. Braun, B.T. Murray, J. Soto Jr., Model. Simul. Mater. Sci. Eng. 5 (4) (1997) 365–380.
[21] C.C. Chen, C.W. Lan, J. Cryst. Growth 311 (3) (2009) 702–706.
[22] Y. Li, J.S. Kim, Int. J. Heat Mass Transfer 55 (25–26) (2012) 7926–7932.
[23] T. Takaki, T. Fukuoka, Y. Tomita, J. Cryst. Growth 283 (1–2) (2005) 263–278.
[24] H.K. Lin, H.Y. Chen, C.W. Lan, J. Cryst. Growth 385 (2014) 44–48.
[25] H.K. Lin, C.W. Lan, J. Cryst. Growth 401 (2014) 740–747.
[26] P.C. Bollada, C.E. Goodyer, P.K. Jimack, A.M. Mullis, F.W. Yang, J. Comput. Phys. 287 (2015) 130–150.
[27] X. Zhang, J. Kang, Z. Guo, S. Xiong, Q. Han, Comput. Phys. Comm. 223 (2018) 18–27.
[28] A. Karma, W.-J. Rappel, Phys. Rev. E 57 (1998) 4323–4349.
[29] Y. Li, H.G. Lee, J. Kim, J. Cryst. Growth 321 (1) (2011) 176–182.
[30] C. Yang, Q. Xu, B. Liu, Comput. Mater. Sci. 136 (2017) 133–143.
[31] X. Luo, L. Wang, W. Ran, F. Qin, Comput. Phys. Comm. 207 (2016) 114–122.
[32] X.H. Wu, G. Wang, L.Z. Zhao, D.C. Zeng, Z.W. Liu, Comput. Mater. Sci. 117 (2016) 286–293.
[33] D. Salac, Comput. Phys. Comm. 204 (2016) 97–106.
[34] S. Qiu, K. Liu, V. Eliasson, Comput. Phys. Comm. 207 (2016) 186–192.
[35] M. Tegeler, O. Shchyglo, R.D. Kamachali, A. Monas, I. Steinbach, G. Sutmann, Comput. Phys. Comm. 215 (2017) 173–187.