Commun Nonlinear Sci Numer Simulat 131 (2024) 107822

Contents lists available at ScienceDirect

Communications in Nonlinear Science and Numerical Simulation

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

Research paper

An efficient and fast adaptive numerical method for a novel phase-field model of crystal growth

Seokjun Ham [a], Yibao Li [b], Soobin Kwak [a], Darae Jeong [c], Junseok Kim [a],*

[a] Department of Mathematics, Korea University, Seoul 02841, Republic of Korea
[b] School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China
[c] Department of Mathematics, Kangwon National University, Gangwon-do 24341, Republic of Korea

ARTICLE INFO

Keywords:
Novel phase-field model
Adaptive finite difference scheme
Dendritic growth

ABSTRACT

In this study, we present an efficient, fast, and fully explicit adaptive numerical scheme for solving a recently developed novel phase-field model of crystal growth in both two-dimensional (2D) and three-dimensional (3D) spaces. The novel phase-field model employed in this research incorporates a term specifically designed to eliminate the artificial curvature effect, thereby facilitating accelerated evolution when compared to traditional models. This enhancement significantly enhances the efficiency and speed of the simulation, leading to more expedited results. The crystal growth model employed in this study incorporates a phase-field equation to accurately represent the crystal interface, in addition to a heat equation that effectively models the distribution of temperature. To effectively solve the phase-field equation, we employ an adaptive numerical algorithm that optimizes the computational process. Our numerical scheme, specifically tailored for simulating dendritic growth, incorporates an adaptive narrow-band domain approach to accurately resolve the interfacial transition layer of the phase field. Furthermore, we enhance computational efficiency by implementing a double-sized grid for the temperature distribution, further improving the overall efficiency of the model. By combining these strategies, we achieve accurate and efficient solutions for the dendritic growth model. To validate the accuracy and efficiency of our proposed adaptive numerical method for solving the phase-field equation of dendritic growth, we conduct a series of numerical experiments in both 2D and 3D spaces. In these experiments, we assess the performance of our algorithm, analyzing its ability to accurately capture the intricate dynamics of dendritic growth while maintaining computational efficiency. By thoroughly evaluating the results obtained from these experiments, we provide strong evidence supporting the reliability and effectiveness of our adaptive numerical algorithm.

1. Introduction

The dendritic growth of crystals is a phenomenon commonly observed in nature, such as in snowflakes, frost patterns, metallic alloys, and crystallization in supersaturated solutions, among others. Therefore, computational and theoretical studies on dendritic growth are valuable for understanding and utilizing natural phenomena. Numerous theoretical studies have been conducted to understand the kinetic of crystal growth [1]. The phase-field method is a highly effective and widely used computational method for modeling and numerically simulating the dendritic growth of crystals [2,3]. Previous models of dendritic growth typically consist

* Corresponding author.
E-mail address: cfdkim@korea.ac.kr (J. Kim).
URL: https://mathematicians.korea.ac.kr/cfdkim/ (J. Kim).

https://doi.org/10.1016/j.cnsns.2024.107822
Received 27 June 2023; Received in revised form 3 November 2023; Accepted 4 January 2024
Available online 6 January 2024

of a system of two equations [4]. One equation is the phase-field equation which represents the interface dynamics of dendrites, while the other equation governs the temperature distribution. The phase-field equation in dendritic growth model is Allen–Cahn (AC) type equation. In terms of efficiency and accuracy, many studies have been conducted to solve the isotropic AC equation [5–7]. However, in the conventional phase-field equation, there is an undesired curvature term. In this study, we adopt a mathematical model that eliminates the curvature effect in the phase-field equation [8].

There have been numerous studies focusing on the efficient and rapid simulation of dendritic growth. In [9], two efficient linear, second-order time marching schemes using linear stabilization were presented. These schemes are provably unconditionally energy stable, enabling the use of large time steps in computations. In [10], an anisotropic lattice Boltzmann-phase field method to study dendritic growth with melt convection was proposed. The authors in [11] proposed an efficient time adaptive algorithm for numerical simulation of nonlinear systems for dendritic crystal growth. To optimize computing resources and reduce computation time while preserving accuracy, various numerical methods have been employed, including adaptive octree mesh [12], multi-GPUs parallel computation for phase-field-lattice Boltzmann model [13,14], adaptive tree-based grids for parallel level-set methods [15], multi-grid method [16], adaptive meshless solution procedure [17], and parallel-adaptive mesh refinement algorithm [18]. In [19], a parallel method using a sequential operator splitting approach was proposed to solve the phase-field crystal growth. Additionally, in [20], an efficient adaptive numerical scheme was proposed, which solves both the phase-field ϕ and temperature distribution T in an adaptive domain.

The adaptive domain for solving the phase-field model was introduced in [20,21]. In this study, we propose a new and efficient adaptive numerical algorithm for the recently proposed phase-field model. The fundamental mechanism of the novel phase-field model effectively eliminates the undesired curvature effect by removing the artificial curvature term. The model consists of the phase of crystal ϕ and the temperature T. Specifically, our proposed method involves numerically solving the phase-field ϕ in an adaptive domain, while solving the temperature distribution T in the entire domain using a double-sized grid, enhancing computational efficiency.

The outline of this paper is as follows. In Sections 2 and 3, we present the governing equations, the proposed fully explicit adaptive finite difference schemes, and computational experiments for dendritic growth in two- and three-dimensional spaces, respectively. Section 4 presents the concluding remarks and discussions drawn from the study.

2. Two-dimensional crystal growth

In this section, we present the two-dimensional (2D) phase-field equation, a numerical solution algorithm, and computational simulations for crystal growth.

2.1. Two-dimensional phase-field equation

A recently proposed novel phase-field model [8], which eliminates the undesired curvature effect for crystal growth in 2D space, is described as follows:

$$\begin{aligned}\epsilon^2(\phi)\frac{\partial\phi}{\partial t} &= \nabla\cdot(\epsilon^2(\phi)\nabla\phi)+[\phi-\lambda U(1-\phi^2)](1-\phi^2)-\epsilon^2(\phi)|\nabla\phi|\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\\ &\quad+\left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_x}\right)_x+\left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_y, \qquad (1)\\ \frac{\partial U}{\partial t} &= D\Delta U+\frac{1}{2}\frac{\partial\phi}{\partial t}, \quad \text{for} \quad \mathbf{x}\in\Omega, \quad t>0, \qquad (2)\end{aligned}$$

where subscripts denote partial derivatives with respect to the indicated variables and $\epsilon(\phi)$ is given by

$$\epsilon(\phi)=W_0(1-3\delta_4)\left(1+\frac{4\delta_4}{1-3\delta_4}\frac{\phi_x^4+\phi_y^4}{|\nabla\phi|^4}\right). \qquad (3)$$

In this model, the variable $\phi(\mathbf{x},t)$ represents the liquid phase when its value is close to -1, and the solid phase when its value is close to 1 at position $\mathbf{x}$ and time t. The variable $U(\mathbf{x},t)$ represents the temperature distribution. Let $\Gamma=\{\mathbf{x}\in\Omega|\phi(\mathbf{x},t)=0\}$ be the interface of the crystal. For more detailed information regarding the definitions of the parameters, please refer to [8,20,22]. The term $\epsilon^2(\phi)|\nabla\phi|\nabla\cdot(\nabla\phi/|\nabla\phi|)$ in Eq. (1) eliminates the undesired curvature effect for crystal growth [8], and if that term is absent, then Eq. (1) becomes the classical phase-field equation for crystal growth [22].

2.2. Two-dimensional numerical scheme

We describe a detailed explanation of an adaptive computational method for the phase-field and temperature equations. Let $\Omega=(L_x,R_x)\times(L_y,R_y)$ and $h=(R_x-L_x)/N_x=(R_y-L_y)/N_y$, where N_x and N_y are even natural numbers. Let $\Omega_h=\{(x_i,y_j)|x_i=L_x+ih,y_j=L_y+jh \text{ for } 0\le i\le N_x,\ 0\le j\le N_y\}$ be an entire discrete domain for the phase-field function. Let $\Omega_H=\{(X_I,Y_J)|X_I=L_x+IH,Y_J=L_y+JH \text{ for } 0\le I\le N_x/2, 0\le J\le N_y/2\}$ be an entire discrete domain for the temperature distribution, where $H=2h$, $X_I=x_{2I}$, and $Y_J=y_{2J}$. For the sake of notation simplicity, $\phi_{ij}^n=\phi(x_i,y_j,n\Delta t)$ and $U_{IJ}^n=U(X_I,Y_J,n\Delta t)$, where Δt is the time step. In the proposed computational procedure, we solve Eqs. (1) and (2) only in a discrete narrow-band domain Ω_{nb}^n (which will be subsequently defined) and Ω_H, respectively, rather than the entire discrete domain Ω_h. Fig. 1(a) shows

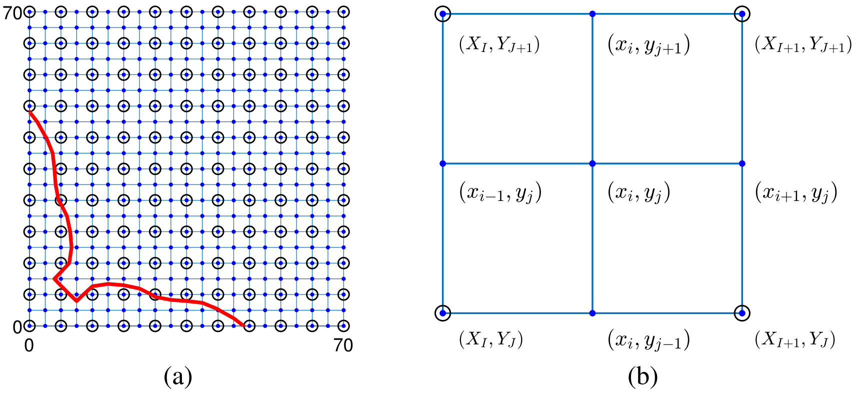


Fig. 1. Schematic illustration of (a) two computational domain for Ω_h (closed circle) and Ω_H (open circle), (b) indices of Ω_h and Ω_H.

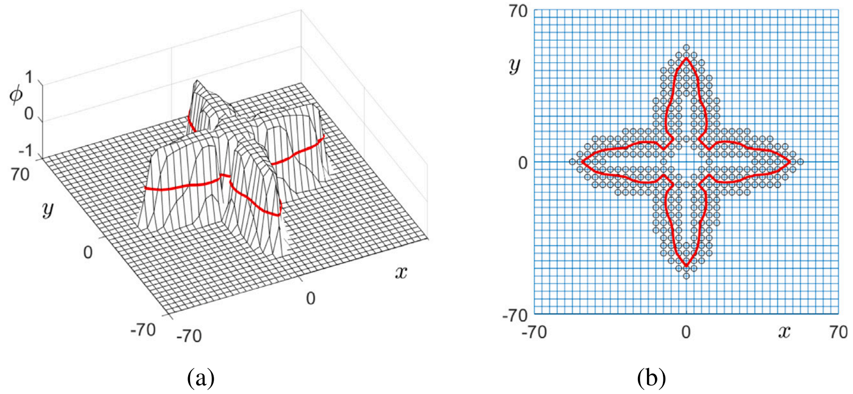


Fig. 2. (a) Phase-field function, which is 4-fold symmetric crystal shape. (b) Space–time adaptive discrete narrow-band domain $\Omega^n_{\rm nb}$ (open circles).

two different computational domains, while Fig. 1(b) shows the indices of the computational domain and illustrates the relationship between the two indices.

Let us now proceed to describe the construction procedure of $\Omega^n_{\rm nb}$ [20]. First, we consider a 4-fold symmetric crystal shape as shown in Fig. 2(a), where the red line is the zero-level contour line of ϕ. A temporal discrete narrow domain is defined as

$$\Omega^n_{\rm tmp} = \left\{(x_i, y_j) \ : \ |\nabla_d \phi^n_{ij}| > \xi, \quad 1 \le i \le N_x, \ 1 \le j \le N_y\right\}, \tag{4}$$

where ξ is a gradient criterion parameter and the discrete gradient is defined as follows:

$$\nabla_d \phi^n_{ij} = \left(\frac{\phi^n_{i+1,j} - \phi^n_{i-1,j}}{2h}, \frac{\phi^n_{i,j+1} - \phi^n_{i,j-1}}{2h}\right). \tag{5}$$

We define a discrete narrow-band domain $\Omega^n_{\rm nb}$ using temporal discrete narrow domain $\Omega^n_{\rm tmp}$:

$$\Omega^n_{\rm nb} = \bigcup_{p=-m}^{p=m} \bigcup_{q=-m}^{q=m} \left\{(x_{i+p}, y_{j+q}) \ : \ (x_i, y_j) \in \Omega^n_{\rm tmp}\right\} \tag{6}$$

for some positive integer m. In this study, $\xi = 0.5$ is used unless otherwise specified. A discrete narrow-band domain $\Omega^n_{\rm nb}$ is illustrated in Fig. 2(b).

Next, we apply the explicit Euler method to solve Eqs. (1) and (2):

$$\begin{aligned}\epsilon^2(\phi^n_{ij})\frac{\phi^{n+1}_{ij} - \phi^n_{ij}}{\Delta t} &= [\nabla_h \cdot (\epsilon^2(\phi)\nabla_h \phi)]^n_{ij} + [\phi^n_{ij} - \lambda \widetilde{U}^n_{ij}(1 - (\phi^n_{ij})^2)](1 - (\phi^n_{ij})^2) \\ &\quad - \left[\epsilon^2(\phi)|\nabla_h \phi| \nabla_h \cdot \left(\frac{\nabla_h \phi}{|\nabla_h \phi|}\right)\right]^n_{ij} + \left[\left(|\nabla_h \phi|^2 \epsilon(\phi)\frac{\partial \epsilon(\phi)}{\partial \phi_x}\right)_x\right]^n_{ij} \\ &\quad + \left[\left(|\nabla_h \phi|^2 \epsilon(\phi)\frac{\partial \epsilon(\phi)}{\partial \phi_y}\right)_y\right]^n_{ij},\end{aligned} \tag{7}$$

$$\frac{U_{IJ}^{n+1} - U_{IJ}^{n}}{\Delta t} = D\Delta_H U_{IJ}^{n} + \frac{\widetilde{\phi}_{IJ}^{n+1} - \widetilde{\phi}_{IJ}^{n}}{2\Delta t}, \tag{8}$$

where $\Delta_H U_{IJ}^n = (U_{I-1,J}^n + U_{I+1,J}^n + U_{I,J-1}^n + U_{I,J+1}^n - 4U_{IJ}^n)/H^2$. Additionally, $\widetilde{U}_{ij}^n$ and $\widetilde{\phi}_{IJ}^n$ are defined as

$$\widetilde{U}_{ij}^n = \left(U^n_{\left[\frac{i}{2}\right],\left[\frac{j}{2}\right]} + U^n_{\left[\frac{i+1}{2}\right],\left[\frac{j}{2}\right]} + U^n_{\left[\frac{i}{2}\right],\left[\frac{j+1}{2}\right]} + U^n_{\left[\frac{i+1}{2}\right],\left[\frac{j+1}{2}\right]} \right) \Big/ 4, \tag{9}$$

$$\widetilde{\phi}_{IJ}^n = \phi_{2I,2J}^n, \tag{10}$$

where $[x]$ is the greatest integer not greater than x. The term $[\nabla_h \cdot (\epsilon^2(\phi)\nabla_h\phi)]_{ij}^n$ in Eq. (7) is defined as follows:

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

The curvature term $\left[\epsilon^2(\phi)|\nabla_h\phi|\nabla_h \cdot (\nabla_h\phi/|\nabla_h\phi|)\right]_{ij}^n$ in Eq. (21) is defined by using the following discretization.

$$\begin{aligned}
\nabla_h \cdot \left(\frac{\nabla_h\phi}{|\nabla_h\phi|}\right)_{ij} = \frac{1}{2h}\Bigg(&\frac{\phi^x_{i+\frac{1}{2},j+\frac{1}{2}} + \phi^y_{i+\frac{1}{2},j+\frac{1}{2}}}{|\nabla_h\phi_{i+\frac{1}{2},j+\frac{1}{2}}|} + \frac{\phi^x_{i+\frac{1}{2},j-\frac{1}{2}} - \phi^y_{i+\frac{1}{2},j-\frac{1}{2}}}{|\nabla_h\phi_{i+\frac{1}{2},j-\frac{1}{2}}|} \\
&- \frac{\phi^x_{i-\frac{1}{2},j+\frac{1}{2}} - \phi^y_{i-\frac{1}{2},j+\frac{1}{2}}}{|\nabla_h\phi_{i-\frac{1}{2},j+\frac{1}{2}}|} - \frac{\phi^x_{i-\frac{1}{2},j-\frac{1}{2}} + \phi^y_{i-\frac{1}{2},j-\frac{1}{2}}}{|\nabla_h\phi_{i-\frac{1}{2},j-\frac{1}{2}}|}\Bigg),
\end{aligned}$$

where $|\nabla_h\phi_{ij}^n| = |\nabla_h\phi^n_{i+\frac{1}{2},j+\frac{1}{2}} + \nabla_h\phi^n_{i+\frac{1}{2},j-\frac{1}{2}} + \nabla_h\phi^n_{i-\frac{1}{2},j+\frac{1}{2}} + \nabla_h\phi^n_{i-\frac{1}{2},j-\frac{1}{2}}|/4$ and $\nabla_h\phi_{i+\frac{1}{2},j+\frac{1}{2}} = (\phi^x_{i+\frac{1}{2},j+\frac{1}{2}}, \phi^y_{i+\frac{1}{2},j+\frac{1}{2}}) = \big((\phi_{i+1,j} + \phi_{i+1,j+1} - \phi_{ij} - \phi_{i,j+1})/(2h), (\phi_{i,j+1} + \phi_{i+1,j+1} - \phi_{ij} - \phi_{i+1,j})/(2h) \big)$. The other terms are discretized as follows:

$$\left[\left(|\nabla_h\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_x}\right)_x\right]_{ij}^n = \left[\left(\frac{16\delta_4\epsilon(\phi)\phi_x\phi_y^2(\phi_x^2 - \phi_y^2)}{|\nabla\phi|^4}\right)_x\right]_{ij}^n,$$

$$\left[\left(|\nabla_h\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_y\right]_{ij}^n = \left[\left(\frac{16\delta_4\epsilon(\phi)\phi_y\phi_x^2(\phi_y^2 - \phi_x^2)}{|\nabla\phi|^4}\right)_y\right]_{ij}^n.$$

2.3. Numerical experiments in two-dimensional space

Unless stated otherwise, the following initial conditions in 2D space are employed:

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

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

with $\phi(x, y, t) = -1$ and $U(x, y, t) = \Delta$ on $\partial\Omega$.

Let us consider a convergence test for the parameter m. Fig. 3 represents the zero-level contours of the solutions (red lines) and their corresponding narrow-band domains (dotted points) at time $t = 5000\Delta t$. We use $R_0 = 3$, $\Delta = -0.55$, $\delta_4 = 0.05$, $W_0 = 1$, $\lambda = 3.1913$, $D = 0.6267\lambda$, $N_x = N_y = 400$, $h = 0.35$, and $\Delta t = 0.1h^2/D$ on $\Omega = (-70, 70) \times (-70, 70)$ [22]. The results presented in Fig. 3 demonstrate that the solution converges as the parameter m increases. Furthermore, it has been observed that when setting $m = 6$, the narrow-band domain Ω_{nb} becomes adequately large to accurately represent the solution. Therefore, we adopt $m = 6$ for all other tests.

The parameter m that controls the buffer of the narrow-band domain should depend on the interfacial thickness of the model. Therefore, we estimate the appropriate m, depending on the interfacial thickness by the following numerical procedure. Fig. 4(a) shows the zero-level contour of the 2D numerical solution. In addition, the cross-section profile of the 2D solution $\phi(x, y, t)$ at $y = 0$ near the right tip is represented in Fig. 4(a). Fig. 4(b) is a magnified view of the image represented in (a) which shows the interfacial thickness of 2ζ and the narrow-band buffer with parameter m. Here, the interfacial thickness is defined as $2\zeta = |x_2 - x_1|$, where $\phi(x_1, 0, t) = 0.9$ and $\phi(x_2, 0, t) = -0.9$. In the presented crystal growth model, the interfacial thickness depends on W_0, which is fixed at 1 in our study. As a result, the interfacial thickness is approximately constant. Because the buffers consist of interior and exterior regions to the interfacial transition layer, we can adjust the narrow-band buffer to a multiple of ζ, as $mh = \alpha\zeta$ for some positive values of α. To determine the parameter m, we use the rounding function defined as $m = \lfloor(\alpha\zeta)/h + 0.5\rfloor$, where $\lfloor x\rfloor = \max\{k \in \mathbb{Z} | k \le x\}$ is

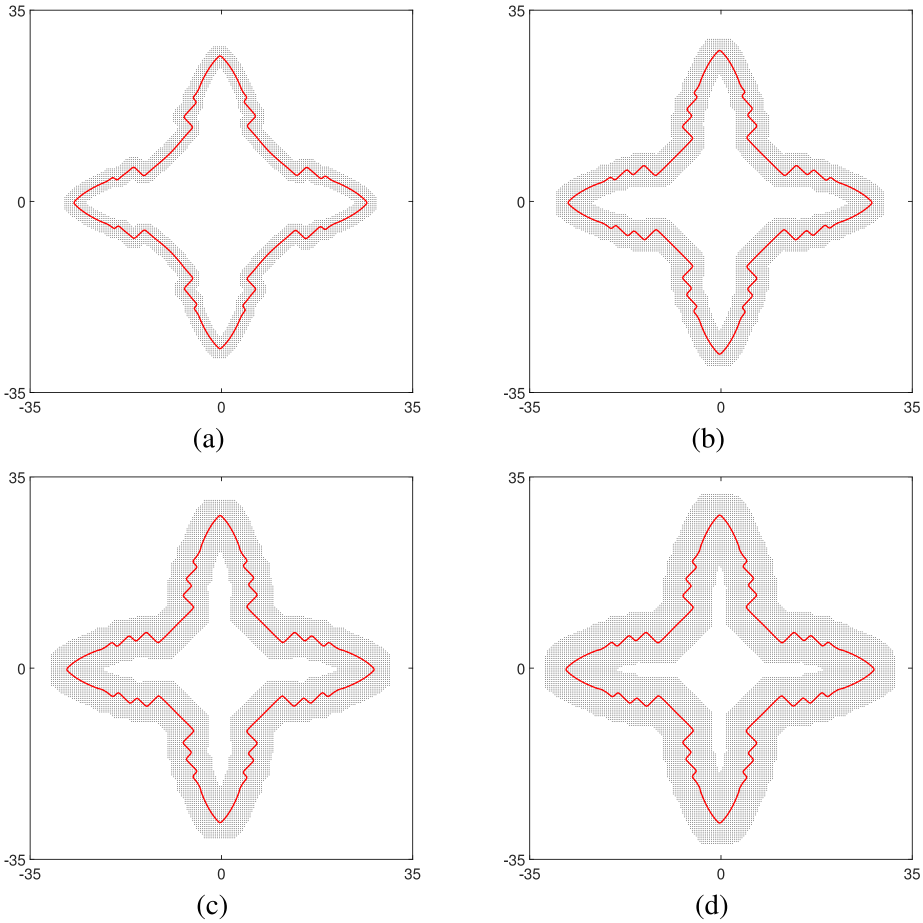


Fig. 3. Zero level contours of solution ϕ at $t = 5000\Delta t$ and four different narrow-band domains Ω_{nb}; $\xi = 0.5$ (a) $m = 2$, (b) $m = 4$, (c) $m = 6$, and (d) $m = 8$.

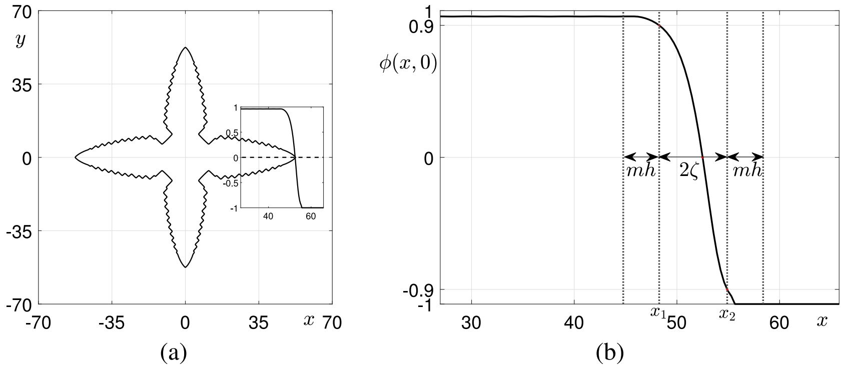


Fig. 4. (a) Zero-level contour of the 2D numerical solution and the cross-section profile of the $\phi(x, y, t)$ at $y = 0$. (b) Magnified cross-section profile in (a) with interfacial thickness 2ζ and buffer length mh.

the floor function. Hence, we can take parameter m, depending on the interfacial thickness of the model and numerical parameter h.

Next, we proceed to compare the solutions obtained for the entire and adaptive domains. Fig. 5(a) shows the zero-level contours of the solutions for the entire domain (solid line) and for the adaptive domain (asterisk markers). Figs. 5(b) and (c) show temporal snapshots of the progression of the zero-level contours of ϕ up to time $t = 10000\Delta t$ for the entire domain and the adaptive domain, respectively. Generally, an adaptive mesh technique is a method used in computational simulations, particularly in numerical methods for solving partial differential equations. It involves adjusting the mesh or grid used in the simulation to focus computational

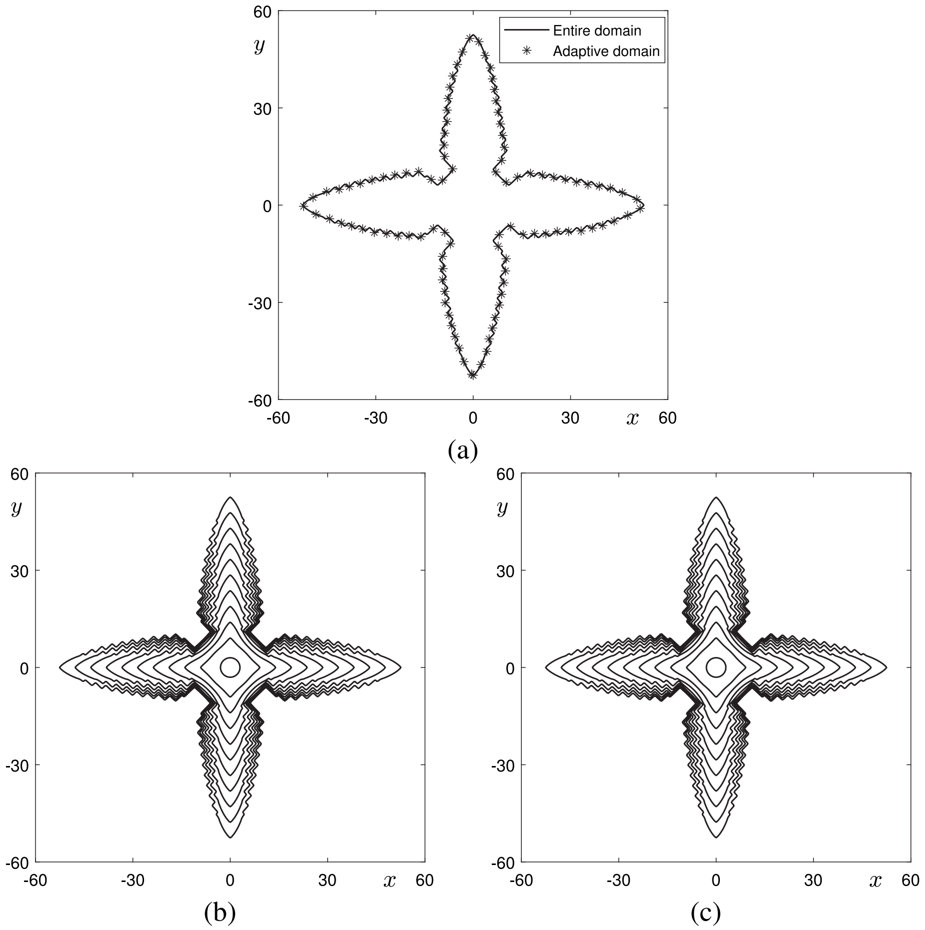


Fig. 5. (a) comparison of the zero-level contours of two solutions for the entire domain and the adaptive domain at $t = 10000\Delta t$. The temporal snapshots of the progression of the contours of ϕ at zero level up to time $t = 10000\Delta t$: (b) solution for the entire domain and (c) solution for the adaptive domain.

Table 1
CPU time (s) for the entire and adaptive domains with $K = 1, 100$, and 500.

	Entire domain	$K = 1$	$K = 100$	$K = 500$
CPU time (s)	86.110	64.625	39.765	33.796

resources more efficiently in regions where the solution varies rapidly or where high accuracy is required, while using coarser mesh spacing in less critical areas. However, one of the most crucial considerations when applying the efficient adaptive mesh technique is maintaining accuracy. In other words, we must ensure that the adaptive mesh technique does not compromise accuracy. As demonstrated in Fig. 5, we can confirm the proposed adaptive mesh method for the novel phase-field model of crystal growth maintains accuracy.

To compare the CPU times between the entire and adaptive domains, we establish a temporally periodic updating scheme for the adaptive domain. Specifically, we update the adaptive computational domain Ω^n_{nb} at intervals of $n = Kp$, where $p = 0,\ 1,\ 2,\ \ldots$, and K is a positive integer. Table 1 lists the CPU times for the entire and adaptive domains with $K = 1,\ 100$, and 500. As expected, CPU times for the adaptive domains with all K values are smaller than for the entire domain. Furthermore, as the value of K increases, the CPU time decreases.

Fig. 6 illustrates the numerical results obtained for the entire and adaptive domains with $K = 1,\ 100$, and 500. Fig. 6(a) shows contours of the numerical solutions of ϕ at zero level for the entire domain (red solid line) and adaptive domains with $K = 1,\ 100$, and 500 at time $t = 10000\Delta t$. We can observe a significant deviation from the entire domain result when $K = 500$ is used. Fig. 6(b) displays the temporal evolution of the discrete area $A(t)$ for the entire and adaptive domains with $K = 1,\ 100$, and 500, where the discrete area $A(t)$ is defined as follows:

$$A(t) = \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \frac{\left(\phi^n_{ij} + 1\right)}{2} h^2. \tag{13}$$

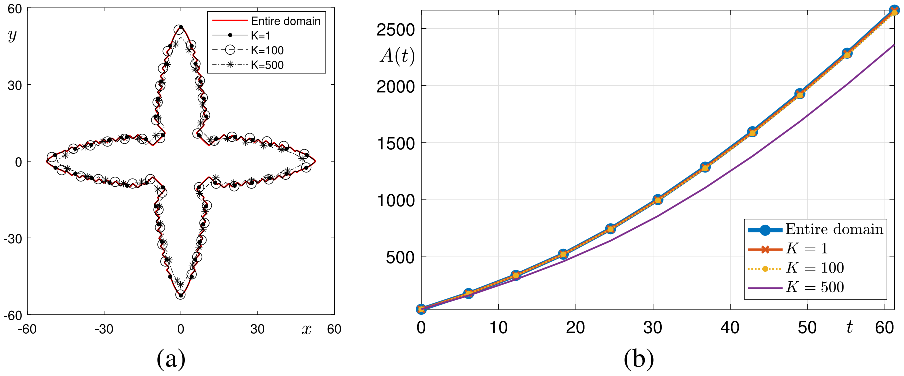


Fig. 6. (a) Contours of the numerical solutions of ϕ at zero level are shown for the entire domain (red solid line) and adaptive domains with $K = 1,\ 100$, and 500 at time $t = 10000\Delta t$. (b) Areas of ϕ on entire domain and adaptive domains with $K = 1, 100$ and 500. This test is conducted up to time $t = 10000\Delta t$.

Table 2
Quantitative comparison of the dimensionless steady-sate tip velocities obtained by our proposed scheme($V_{OP} = Vd_0/D$), the results in [23] (V_{LLK}), results in [24] (V_{KR}), and Green's function formula(V_{GF}), respectively.

Δ	δ_4	D	d_0/W_0	V_{LLK}	V_{KR}	V_{GF}	V_{OP}
−0.55	0.05	2	0.277	0.0171	0.0168	0.0170	0.0170
−0.55	0.05	3	0.185	0.0174	0.0175	0.0170	0.0172
−0.55	0.05	4	0.139	0.0172	0.0174	0.0170	0.0171
−0.50	0.05	3	0.185	0.0103	0.0101	0.0099	0.0102
−0.45	0.05	3	0.185	0.0060	0.0056	0.0055	0.0055

Here, the test is conducted up to time $t = 10000\Delta t$. We can also note a substantial difference in the area compared to the entire domain result when using $K = 500$. Based on the computational results shown in Fig. 6 and the data presented in Table 1, choosing $K = 100$ appears to be a reasonable option for achieving both efficient and accurate computational results.

Then, we compare the dimensionless steady-state tip velocities of two solutions for entire domain and adaptive domain. The quantitative comparison of the dimensionless steady-state tip velocities are obtained by our proposed schemes ($V_{OP} = Vd_0/D$), the results in [23] (V_{LLK}), results in [24] (V_{KR}), and Green's function formula(V_{GF}), respectively. This numerical test is investigated on the domain $\Omega = (-200, 200)^2$ with a 1024×1024 mesh grid under the following parameters setting: $R_0 = 3$, $\Delta = -0.55$, $\delta_4 = 0.05$, $W_0 = 1$, $\lambda = D/a_2$, $a_2 = 0.6267$, $h = 0.35$, and $\Delta t = 0.1h^2/D$. The computational formula for the steady-state velocity is based on a quadratic polynomial approximation, $y = \alpha x^2 + \beta x + \gamma$. We assume that the three points (x_{k-1}, y_{k-1}), (x_k, y_k), (x_{k+1}, y_{k-1}), are on the interface where y_k is a maximum value. Thus, we can find the tip position y_* by taking $y\prime(x_*) = 0$ and $y_* = \alpha x_*^2 + \beta x_* + \gamma$. It can be observed from the results in Table 2 that the results are almost identical.

Next, we consider the six-fold crystal growth model applied in the proposed adaptive method and presents numerical experiments to demonstrate the consistency of the proposed adaptive method through comparison. The six-fold crystal model is presented in [8] as follows:

$$\begin{aligned} \epsilon^2(\theta)\frac{\partial \phi}{\partial t} &= \nabla \cdot (\epsilon^2(\theta)\nabla\phi) + [\phi - \lambda U(1-\phi^2)](1-\phi^2) - \epsilon^2(\theta)|\nabla\phi|\nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right) \\ &\quad - (\epsilon'(\theta)\epsilon(\theta)\phi_y)_x + (\epsilon'(\theta)\epsilon(\theta)\phi_x)_y \end{aligned} \tag{14}$$

$$\frac{\partial U}{\partial t} = D\Delta U + \frac{1}{2}\frac{\partial \phi}{\partial t}, \quad \text{for } \mathbf{x} \in \Omega, \quad t > 0, \tag{15}$$

where θ satisfies $\tan\theta = (-\phi_y)/(-\phi_x)$ and $\epsilon(\phi) = \epsilon_0(1 + \epsilon_6\cos(6\phi))$ is the anisotropic function. Then, we discretize Eqs. (14) and (15) using the explicit Euler scheme as follows:

$$\begin{aligned} \epsilon^2(\theta_{ij}^n)\frac{\phi_{ij}^{n+1} - \phi_{ij}^n}{\Delta t} &= \left[\nabla_h \cdot (\epsilon^2(\theta)\nabla_h\phi)\right]_{ij}^n + [\phi_{ij}^n - \lambda\widetilde{U}_{ij}^n(1-(\phi_{ij}^n)^2)][1-(\phi_{ij}^n)^2] \\ &\quad - \left[\epsilon^2(\theta)|\nabla_h\phi|\nabla_h \cdot \left(\frac{\nabla_h\phi}{|\nabla_h\phi|}\right)\right]_{ij}^n - \left[D_x\left(\epsilon'(\theta)\epsilon(\theta)D_y\phi\right)\right]_{ij}^n \\ &\quad + \left[D_y\left(\epsilon'(\theta)\epsilon(\theta)D_x\phi\right)\right]_{ij}^n, \end{aligned} \tag{16}$$

$$\frac{U_{IJ}^{n+1} - U_{IJ}^n}{\Delta t} = D\nabla_H^2 U_{IJ}^n + \frac{\widetilde{\phi}_{IJ}^{n+1} - \widetilde{\phi}_{IJ}^n}{2\Delta t}. \tag{17}$$

The angle between the normal vector of interface is obtained as $\theta_{ij}^n = \tanh\left[(\phi_{i,j+1}^n - \phi_{i,j-1}^n)/(\phi_{i+1,j}^n - \phi_{i-1,j}^n)\right]$. The terms in the right hand side of Eq. (16) are discretized as follows:

$$\begin{aligned}
\left[\nabla_h \cdot (\epsilon^2(\theta)\nabla_h \phi)\right]_{ij}^n &= \frac{\epsilon^2(\theta_{i,j+\frac{1}{2}}^n)\left(\phi_{i,j+1}^n - \phi_{ij}^n\right) - \epsilon^2(\theta_{i,j-\frac{1}{2}}^n)\left(\phi_{ij}^n - \phi_{i,j-1}^n\right)}{h^2} \\
&+ \frac{\epsilon^2(\theta_{i+\frac{1}{2},j}^n)\left(\phi_{i+1,j}^n - \phi_{ij}^n\right) - \epsilon^2(\theta_{i-\frac{1}{2},j}^n)\left(\phi_{ij}^n - \phi_{i-1,j}^n\right)}{h^2}, \\
\nabla_h \cdot \left(\frac{\nabla_h \phi}{|\nabla_h \phi|}\right)_{ij} &= \frac{1}{2h}\left(\frac{\phi_{i+\frac{1}{2},j+\frac{1}{2}}^x + \phi_{i+\frac{1}{2},j+\frac{1}{2}}^y}{|\nabla_h \phi_{i+\frac{1}{2},j+\frac{1}{2}}|} + \frac{\phi_{i+\frac{1}{2},j-\frac{1}{2}}^x - \phi_{i+\frac{1}{2},j-\frac{1}{2}}^y}{|\nabla_h \phi_{i+\frac{1}{2},j-\frac{1}{2}}|}\right. \\
&\left. - \frac{\phi_{i-\frac{1}{2},j+\frac{1}{2}}^x - \phi_{i-\frac{1}{2},j+\frac{1}{2}}^y}{|\nabla_h \phi_{i-\frac{1}{2},j+\frac{1}{2}}|} - \frac{\phi_{i-\frac{1}{2},j-\frac{1}{2}}^x + \phi_{i-\frac{1}{2},j-\frac{1}{2}}^y}{|\nabla_h \phi_{i-\frac{1}{2},j-\frac{1}{2}}|}\right), \\
\left[D_x\left(\epsilon'(\theta)\epsilon(\theta)D_y\phi\right)\right]_{ij}^n &= \frac{\epsilon'(\theta_{i+\frac{1}{2},j})\epsilon(\theta_{i+\frac{1}{2},j})\left(\phi_{i+1,j+1}^n - \phi_{i+1,j-1}^n + \phi_{i,j+1}^n - \phi_{i,j-1}^n\right)}{4h^2} \\
&- \frac{\epsilon'(\theta_{i-\frac{1}{2},j})\epsilon(\theta_{i-\frac{1}{2},j})\left(\phi_{i,j+1}^n - \phi_{i,j-1}^n + \phi_{i-1,j+1}^n - \phi_{i-1,j-1}^n\right)}{4h^2}, \\
\left[D_y\left(\epsilon'(\theta)\epsilon(\theta)D_x\phi\right)\right]_{ij}^n &= \frac{\epsilon'(\theta_{i,j+\frac{1}{2}})\epsilon(\theta_{i,j+\frac{1}{2}})\left(\phi_{i+1,j+1}^n - \phi_{i-1,j+1}^n + \phi_{i+1,j}^n - \phi_{i-1,j}^n\right)}{4h^2} \\
&- \frac{\epsilon'(\theta_{i,j-\frac{1}{2}})\epsilon(\theta_{i,j-\frac{1}{2}})\left(\phi_{i+1,j}^n - \phi_{i-1,j}^n + \phi_{i+1,j-1}^n - \phi_{i-1,j-1}^n\right)}{4h^2},
\end{aligned}$$

where $\epsilon'(\theta) = -6\epsilon_0\epsilon_6 \sin(6\theta)$ and

$$\begin{aligned}
\theta_{i+\frac{1}{2},j}^n &= \tan^{-1}\left(\frac{\phi_{i+1,j-1}^n - \phi_{i+1,j+1}^n + \phi_{i,j-1}^n - \phi_{i,j+1}^n}{4(\phi_{ij}^n - \phi_{i+1,j}^n)}\right), \\
\theta_{i-\frac{1}{2},j}^n &= \tan^{-1}\left(\frac{\phi_{i,j-1}^n - \phi_{i,j+1}^n + \phi_{i-1,j-1}^n - \phi_{i-1,j+1}^n}{4(\phi_{i-1,j}^n - \phi_{ij}^n)}\right), \\
\theta_{i,j+\frac{1}{2}}^n &= \tan^{-1}\left(\frac{4(\phi_{ij}^n - \phi_{i,j+1}^n)}{\phi_{i-1,j+1}^n - \phi_{i+1,j+1}^n + \phi_{i-1,j}^n - \phi_{i+1,j}^n}\right), \\
\theta_{i,j-\frac{1}{2}}^n &= \tan^{-1}\left(\frac{4(\phi_{i,j-1}^n - \phi_{ij}^n)}{\phi_{i-1,j}^n - \phi_{i+1,j}^n + \phi_{i-1,j-1}^n - \phi_{i+1,j-1}^n}\right).
\end{aligned}$$

For the following numerical experiments of the six-fold crystal growth, we use $R_0 = 3$, $\Delta = -0.55$, $\epsilon_0 = 1$, $\epsilon_6 = 0.05$, $\lambda = 3.1913$, $D = 0.6267\lambda$, $N_x = N_y = 400$, $h = 0.35$, and $\Delta t = 0.1h^2/D$ on the computational domain $\Omega = (-70, 70) \times (-70, 70)$. In Fig. 7, we compare the solutions obtained for the entire and adaptive domains for the six-fold crystal growth model. Fig. 7(a) shows the zero-level contours of two solutions for the entire domain (−) and adaptive domain ($*$) at $t = 25000\Delta t$. Figs. 7(b) and (c) illustrate temporal snapshots of the zero-level contours up to time $t = 25000\Delta t$. From the results of the numerical test in Fig. 7, we can confirm that the proposed adaptive method works well in the six-fold crystal growth model.

3. Three-dimensional crystal growth

3.1. Three-dimensional phase-field equation

Next, we extend the 2D equation to incorporate three dimensions. The three-dimensional (3D) phase-field equation for the 3D crystal growth is expressed as follows:

$$\begin{aligned}
\epsilon^2(\phi)\frac{\partial \phi}{\partial t} &= \nabla \cdot (\epsilon^2(\phi)\nabla\phi) + [\phi - \lambda U(1-\phi^2)](1-\phi^2) - \epsilon^2(\phi)|\nabla\phi|\nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right) \qquad (18) \\
&+ \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_x}\right)_x + \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_y + \left(|\nabla\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_z}\right)_z, \\
\frac{\partial U}{\partial t} &= D\Delta U + \frac{1}{2}\frac{\partial\phi}{\partial t}, \quad \text{for } \mathbf{x} \in \Omega, \quad t > 0, \qquad (19)
\end{aligned}$$

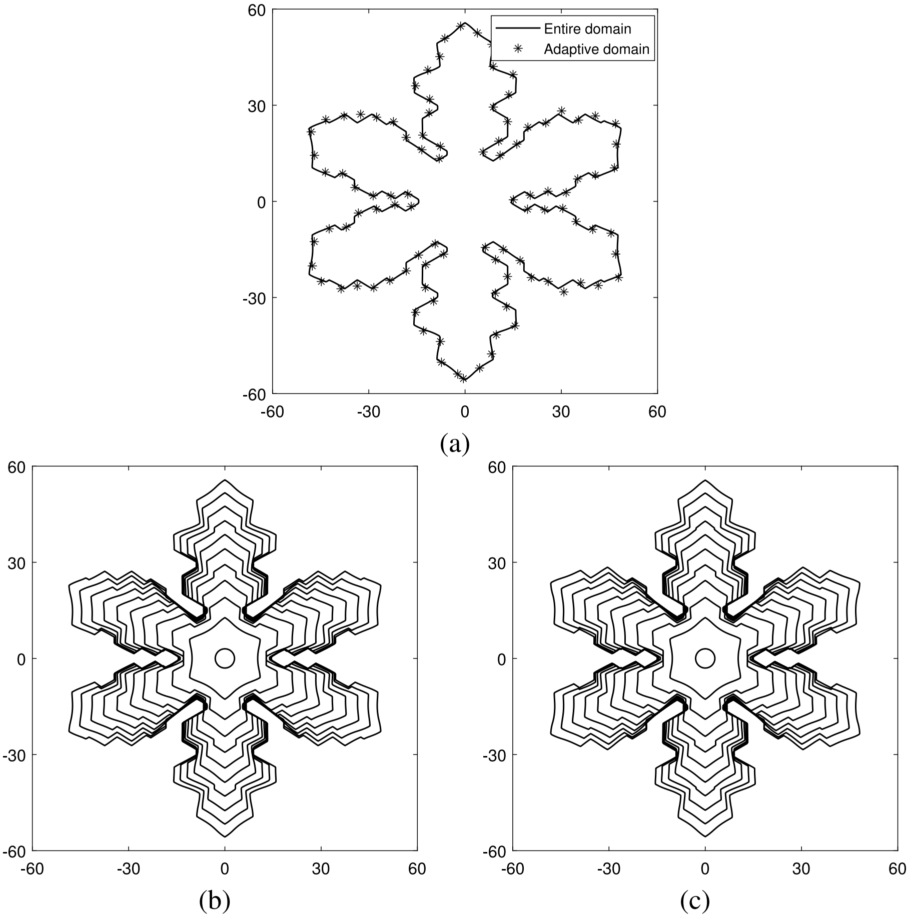


Fig. 7. (a) comparison of the zero-level contours of two solutions for entire domain and adaptive domain, respectively, at $t = 25000\Delta t$. (b) and (c) are temporal snapshots of the progression of the contours of ϕ at zero level up to time $t = 25000\Delta t$ on the entire domain and adaptive domain, respectively.

where $\epsilon(\phi)$ is given as

$$\epsilon(\phi) = W_0(1-3\delta_4)\left(1+\frac{4\delta_4}{1-3\delta_4}\frac{\phi_x^4+\phi_y^4+\phi_z^4}{|\nabla\phi|^4}\right).$$

3.2. Three-dimensional numerical scheme

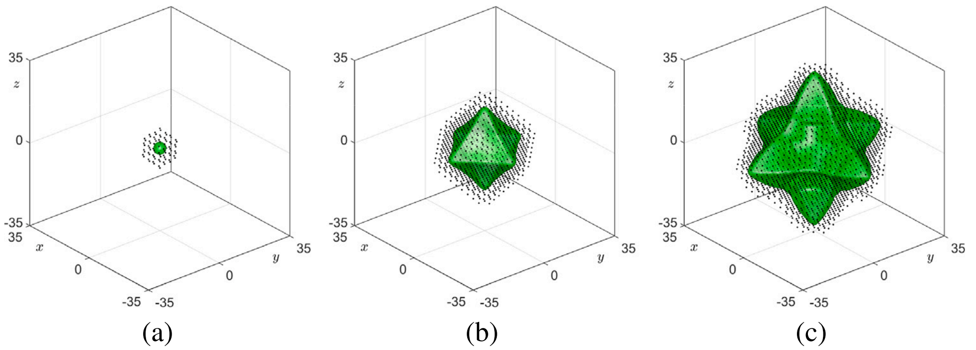
We use an adaptive computational method for Eqs. (18) and (19) on a 3D domain $\Omega = (L_x, R_x)\times(L_y, R_y)\times(L_z, R_z)$. For positive even integers N_x, N_y, and N_z, let $h = (R_x - L_x)/N_x = (R_y - L_y)/N_y = (R_z - L_z)/N_z$ be the space grid size. The discrete domain is then defined as follows: $\Omega_h = \{(x_i, y_j, z_k)|x_i = L_x + ih, y_j = L_y + jh, z_k = L_z + kh$ for $0 \le i \le N_x,\ 0 \le j \le N_y,\ 0 \le k \le N_z\}$ and $\Omega_H = \{(X_I, Y_J, Z_K)|X_I = L_x + IH, Y_J = L_y + JH, Z_K = L_z + KH$ for $0 \le I \le N_x/2, 0 \le J \le N_y/2, 0 \le K \le N_z/2\}$, where $H = 2h$, $X_I = x_{2I}$, $Y_J = y_{2J}$ and $Z_K = z_{2K}$. For the sake of notation simplicity, we set $\phi^n_{ijk} = \phi(x_i, y_j, z_k, n\Delta t)$ and $U^n_{IJK} = U(X_I, Y_J, Z_K, n\Delta t)$, where Δt is the time step. We define 3D temporal discrete narrow domain as $\Omega^n_{\text{tmp}} = \{(x_i, y_j, z_k) \mid |\nabla_d \phi^n_{ijk}| > \xi,\ 0 \le i \le N_x,\ 0 \le j \le N_y,\ 0 \le k \le N_z\}$, where $\nabla_d \phi_{ijk}$ is defined as follows:

$$\nabla_d \phi_{ijk} = \left(\frac{\phi_{i+1,j,k} - \phi_{i-1,j,k}}{2h}, \frac{\phi_{i,j+1,k} - \phi_{i,j-1,k}}{2h}, \frac{\phi_{i,j,k+1} - \phi_{i,j,k-1}}{2h}\right).$$

A 3D space–time adaptive discrete narrow-band domain Ω^n_{nb} is defined as

$$\Omega^n_{\text{nb}} = \bigcup_{p=-m}^{p=m}\bigcup_{q=-m}^{q=m}\bigcup_{r=-m}^{r=m}\left\{(x_{i+p}, y_{j+q}, z_{k+r})\ :\ (x_i, y_j, z_k) \in \Omega^n_{\text{tmp}}\right\} \tag{20}$$

for positive integer m. The schematic of the 3D results in the discrete narrow-band domain is illustrated in Fig. 8.



(a)　(b)　(c)

Fig. 8. Schematic illustration of the 3D dendritic growth in the narrow-band domain Ω_{nb} (dotted points) over time.

The computational solutions for the 3D crystal growth are obtained using the following explicit Euler method:

$$\begin{aligned}
\epsilon^2(\phi^n_{ijk})\frac{\phi^{n+1}_{ijk}-\phi^n_{ijk}}{\Delta t} &= [\nabla_h\cdot(\epsilon^2(\phi)\nabla_h\phi)]^n_{ijk}+[\phi^n_{ijk}-\lambda\widetilde{U}^n_{ijk}(1-(\phi^n_{ijk})^2)](1-(\phi^n_{ijk})^2)\\
&\quad-\left[\epsilon^2(\phi)|\nabla_h\phi|\nabla_h\cdot\left(\frac{\nabla_h\phi}{|\nabla_h\phi|}\right)\right]^n_{ijk}+\left[\left(|\nabla_h\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_x}\right)_x\right]^n_{ijk}\\
&\quad+\left[\left(|\nabla_h\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_y\right]^n_{ijk}+\left[\left(|\nabla_h\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_z\right]^n_{ijk}, \qquad (21)\\
\frac{U^{n+1}_{IJK}-U^n_{IJK}}{\Delta t} &= D\Delta_H U^n_{IJK}+\frac{\widetilde{\phi}^{n+1}_{IJ}-\widetilde{\phi}^n_{IJ}}{2\Delta t}, \qquad (22)
\end{aligned}$$

where $\Delta_H U^n_{IJK}=(U^n_{I-1,J,K}+U^n_{I+1,J,K}+U^n_{I,J-1,K}+U^n_{I,J+1,L}+U^n_{I,J,K-1}+U^n_{I,J,K+1}-6U^n_{IJK})/H^2$. Also $\widetilde{U}^n_{ijk}$ and $\widetilde{\phi}^n_{IJK}$ are defined as

$$\begin{aligned}
\widetilde{U}^n_{ijk} &= \Bigg(U^n_{\left[\frac{i}{2}\right],\left[\frac{j}{2}\right],\left[\frac{k}{2}\right]}+U^n_{\left[\frac{i+1}{2}\right],\left[\frac{j}{2}\right],\left[\frac{k}{2}\right]}+U^n_{\left[\frac{i}{2}\right],\left[\frac{j+1}{2}\right],\left[\frac{k}{2}\right]}+U^n_{\left[\frac{i}{2}\right],\left[\frac{j}{2}\right],\left[\frac{k+1}{2}\right]}+U^n_{\left[\frac{i+1}{2}\right],\left[\frac{j+1}{2}\right],\left[\frac{k}{2}\right]}\\
&\qquad+U^n_{\left[\frac{i}{2}\right],\left[\frac{j+1}{2}\right],\left[\frac{k+1}{2}\right]}+U^n_{\left[\frac{i+1}{2}\right],\left[\frac{j}{2}\right],\left[\frac{k+1}{2}\right]}+U^n_{\left[\frac{i+1}{2}\right],\left[\frac{j+1}{2}\right],\left[\frac{k+1}{2}\right]}\Bigg)\Big/8, \qquad (23)\\
\widetilde{\phi}^n_{IJK} &= \phi^n_{2I,2J,2K}. \qquad (24)
\end{aligned}$$

The term $[\nabla_h\cdot(\epsilon^2(\phi)\nabla_h\phi)]^n_{ij}$ in 3D space is defined as follows:

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

The curvature term $\left[\epsilon^2(\phi)|\nabla_h\phi|\nabla_h\cdot(\nabla_h\phi/|\nabla_h\phi|)\right]^n_{ijk}$ in Eq. (21) can be similarly discretized to 2D space, and further details can be found in [25]. The other terms are discretized as follows:

$$\begin{aligned}
\left[\left(|\nabla_h\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_x}\right)_x\right]^n_{ijk} &= \left[\left(\frac{16\delta_4\epsilon(\phi)\phi_x(\phi_x^2\phi_y^2+\phi_x^2\phi_z^2-\phi_y^4-\phi_z^4)}{|\nabla\phi|^4}\right)_x\right]^n_{ijk},\\
\left[\left(|\nabla_h\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_y\right]^n_{ijk} &= \left[\left(\frac{16\delta_4\epsilon(\phi)\phi_y(\phi_y^2\phi_x^2+\phi_y^2\phi_z^2-\phi_x^4-\phi_z^4)}{|\nabla\phi|^4}\right)_y\right]^n_{ijk},\\
\left[\left(|\nabla_h\phi|^2\epsilon(\phi)\frac{\partial\epsilon(\phi)}{\partial\phi_y}\right)_x\right]^n_{ijk} &= \left[\left(\frac{16\delta_4\epsilon(\phi)\phi_z(\phi_z^2\phi_x^2+\phi_z^2\phi_y^2-\phi_x^4-\phi_y^4)}{|\nabla\phi|^4}\right)_z\right]^n_{ijk}.
\end{aligned}$$

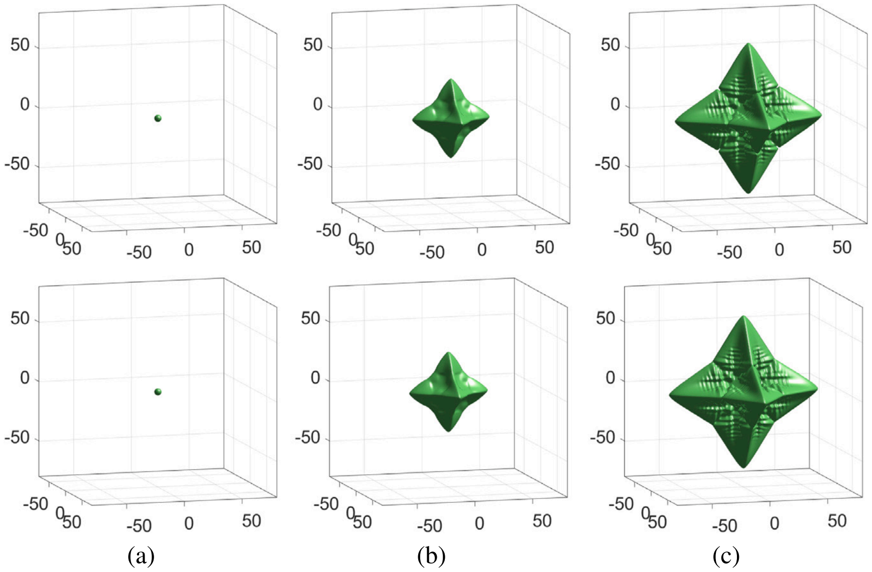


Fig. 9. Temporal progress of the zero-level isosurface of ϕ for the 3D novel dendritic growth model. From top to bottom rows, we consider the entire and adaptive domains, respectively. Here, (a), (b), and (c) are the computational results at times $t = 0$, $900\Delta t$, $1800\Delta t$, respectively. $K = 100$ is used.

Table 3
CPU time (s) for the entire and adaptive domains with $K = 1$, 100, and 150 in 3D space.

Case	Entire domain	$K = 1$	$K = 100$	$K = 150$
CPU time (s)	843.219	470.781	288.859	270.495

3.3. Numerical experiments in three-dimensional space

Unless stated otherwise, the following initial conditions in 3D space are employed:

$$\phi(x, y, z, 0) = \tanh\left(\frac{R_0 - \sqrt{x^2 + y^2 + z^2}}{\sqrt{2}}\right), \tag{25}$$

$$U(x, y, z, 0) = \begin{cases} 0, & \text{if } \phi > 0, \\ \Delta, & \text{otherwise,} \end{cases} \tag{26}$$

with $\phi(x, y, z, t) = -1$ and $U(x, y, z, t) = \Delta$ on $\partial\Omega$ for all t. For the following 3D tests, we use an initial radius $R_0 = 3$, $\Delta = -0.55$, $\delta_4 = 0.05$, $W_0 = 1$, $\lambda = 3.1913$, and $D = 0.6267\lambda$. The computational parameters used are $N_x = N_y = N_z = 200$, $h = 0.8$, $\Delta t = 0.1h^2/D$, $m = 6$, $K = 100$, and $\xi = 0.5$ on $\Omega = (-80, 80) \times (-80, 80) \times (-80, 80)$. Fig. 9 displays temporal progress of the isosurface of the computational solutions at level 0 for the 3D novel dendritic growth model up to time $t = 1800\Delta t$. In Fig. 9, the top and bottom rows illustrate the numerical solutions for the entire domain and adaptive domain, respectively. Here, (a), (b), and (c) are the computational results at times $t = 0$, $900\Delta t$, $1800\Delta t$, respectively. We can confirm that the computational results on the adaptive domain closely match those on the entire domain computation.

From top to bottom, Figs. 10(a)–(d) display the temporal evolutions of the numerical results for the entire and adaptive domains with $K = 1$, $K = 100$, and $K = 150$ in 3D space, respectively. When we examine the computational outcomes visually, we find that they exhibit a high degree of similarity in terms of their appearance and characteristics. However, as shown in Fig. 10(e), the temporal evolution of the discrete volume of ϕ with $K = 150$ deviates from the results of the entire domain. Here, the discrete volume $V(t)$ is defined as follows:

$$V(t) = \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \sum_{k=0}^{N_z} \frac{\left(\phi_{ijk}^n + 1\right)}{2} h^3. \tag{27}$$

Table 3 lists the CPU times for the entire and adaptive domains with $K = 1$, 100, and 150 in 3D space. As expected, CPU times for the adaptive domains with all K values are smaller than for the entire domain. Furthermore, as the value of K increases, the CPU time decreases. Based on the numerical results shown in Fig. 10 and Table 3, choosing $K = 100$ is both efficient and accurate for achieving computational results.

Finally, we conduct a comparison test of 2D and 3D numerical results for the crystal growth models. The simulations are performed in the domains $\Omega = (80, 80)^d$, where $d = 2$ or 3 is the space dimension. For this test, we take the following parameter

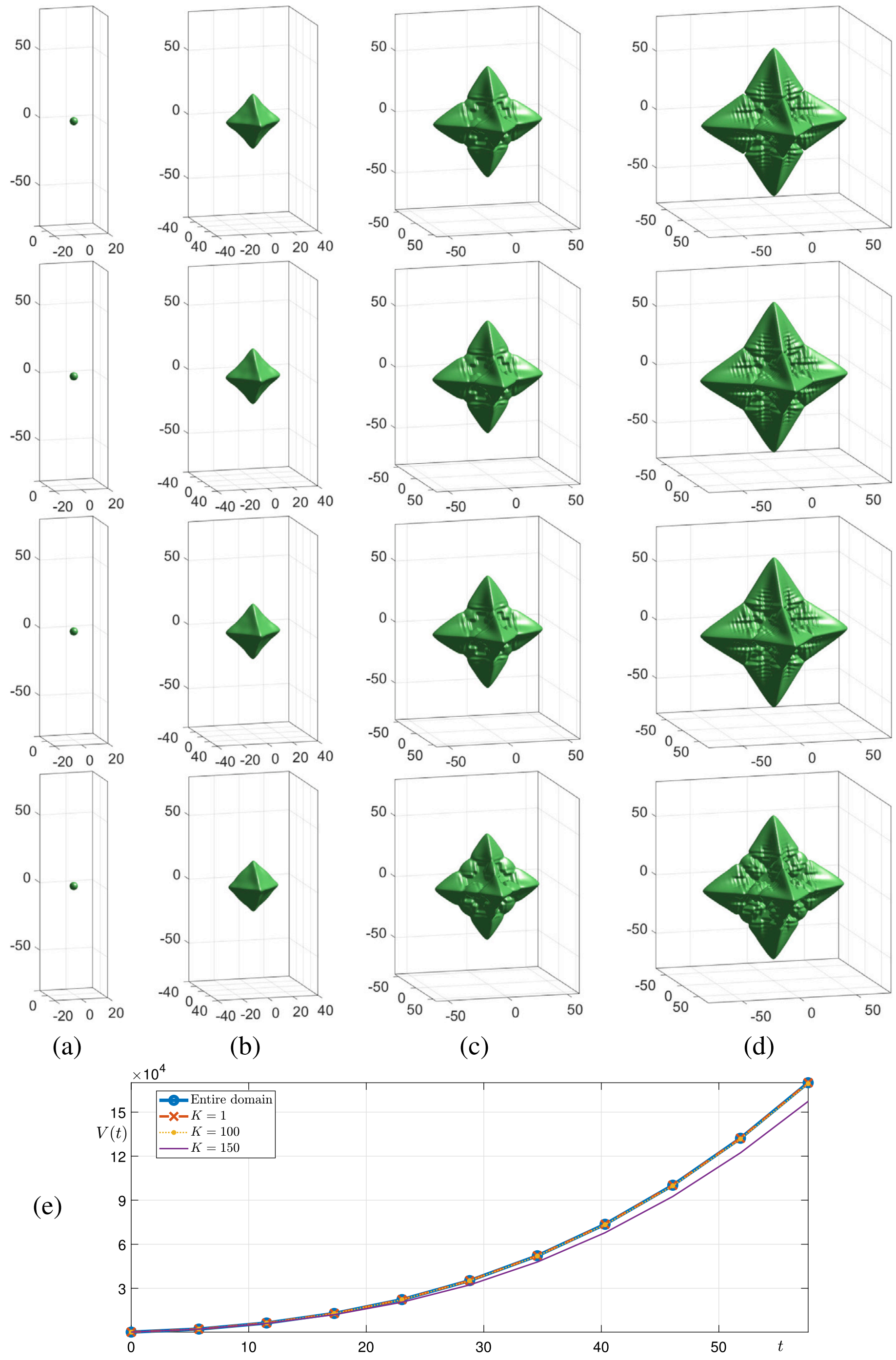


Fig. 10. From the top to bottom row, temporal evolution of the zero-level isosurface of the numerical solution ϕ for the entire and adaptive domain with $K = 1$, $K = 100$, and $K = 150$ at (a) $t = 0$, (b) $t = 540\Delta t$, (c) $t = 1080\Delta t$, (d) $t = 1800\Delta t$. (e) Volumes of ϕ on entire domain and adaptive domains with $K = 1$, 100 and 150.

values: $R_0 = 3$, $\Delta = -0.55$, $\delta_4 = 0.05$, $W_0 = 1$, $\lambda = 3.1913$, $D = 0.6267\lambda$, $N_x = N_y = N_z = 200$, $h = 0.8$, and $\Delta t = 0.1h^2/D$. Temporal evolutions of the zero-level contours and isosurfaces of the numerical solutions for 2D and 3D crystal growth are shown in Figs. 11(a) and (b), respectively. From a 3D space perspective, the 2D result is effectively a cylindrical expansion. The result of expansion into 3D cylinder form is shown in Fig. 11(c). Fig. 11(d) compares the 2D result and 3D slice result $\phi(x, y, 0, t)$ projected onto the x–y

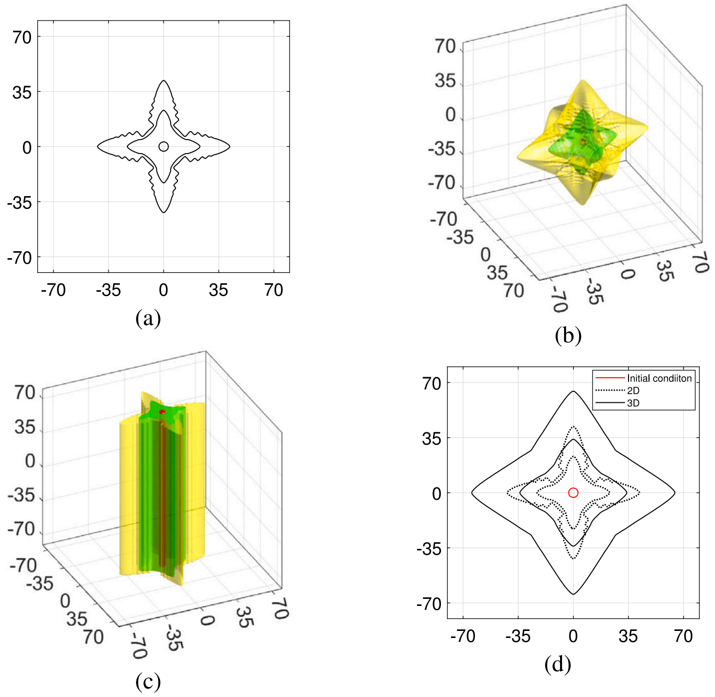


Fig. 11. (a) Temporal snapshots of the zero-level contours of the numerical solutions for the 2D crystal growth up to time $t = 1800\Delta t$. (b) Temporal snapshots of the zero-level isosurfaces of the numerical solutions for the 3D crystal growth up to time $t = 1800\Delta t$. (c) Temporal snapshots of the zero-level isosurfaces of the 2D numerical solution expanded into 3D cylinder form. (d) Temporal snapshots of the 2D numerical results (dotted line) and zero-level contours of the 3D numerical solution $\phi(x, y, 0, t)$ projected onto the x–y plane (solid line).

plane. In contrast to the 2D result, the tip experiences an additional undercooling effect in the upward and downward directions along the z-axis in the 3D result. Therefore, the 3D crystal grows more rapidly than it does in 2D space.

4. Conclusions

In this study, we have presented an efficient, fast, and fully explicit adaptive numerical scheme for solving a novel phase-field equation of crystal growth in both 2D and 3D spaces. The recently developed phase-field equation incorporates a term that eliminates artificial curvature effects, resulting in faster evolution compared to conventional models. The dendritic growth model consists of a phase-field equation for capturing the crystal interface and a heat equation for describing temperature distribution. To solve the phase-field equation accurately and efficiently, we have introduced an adaptive numerical algorithm. This algorithm uses an adaptive narrow-band domain to capture the phase-field interface and enhances accuracy by employing a double-sized grid for temperature distribution. Through a series of numerical experiments in 2D and 3D spaces, we have demonstrated the accuracy and efficiency of our proposed adaptive numerical algorithm for solving the phase-field equation of crystal growth. Overall, our work contributes to the advancement of phase-field simulation in dendritic growth and provides an effective approach for accurately capturing the complex interface dynamics. The developed adaptive numerical scheme offers improved computational efficiency and accuracy, opening avenues for further investigations and applications in the field of materials science and solidification phenomena.

CRediT authorship contribution statement

Seokjun Ham: Formal analysis, Writing – original draft, Validation, Visualization, Review & editing. **Yibao Li:** Formal analysis, Writing – original draft, Software, Review & editing. **Soobin Kwak:** Software, Data curation, Review & editing. **Darae Jeong:** Methodology, Software, Review & editing. **Junseok Kim:** Supervision, Methodology, Software, Review & editing, Project administration, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

Y.B. Li is supported by National Natural Science Foundation of China (No. 12271430). D. Jeong was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2023-00248564). The corresponding author (J.S. Kim) was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1A2C1003844). The authors express their gratitude to the reviewers for their helpful feedback on this revised version.

References

[1] Zhang J, Chen C, Yang X. A novel decoupled and stable scheme for an anisotropic phase-field dendritic crystal growth model. Appl Math Lett 2019;95:122–9.

[2] Gong TZ, Chen Y, Li DZ, Cao YF, Fu PX. Quantitative comparison of dendritic growth under forced flow between 2D and 3D phase-field simulation. Int J Heat Mass Transf 2019;135:262–73.

[3] Zhang Y, Wang X, Yang S, Chen W, Hou H. The morphology and solute segregation of dendrite growth in Ti-4.5% Al alloy: A phase-field study. Materials 2021;14(23):7257.

[4] Lee S, Li Y, Shin J, Kim J. Phase-field simulations of crystal growth in a two-dimensional cavity flow. Comput Phys Commun 2017;216:84–94.

[5] Zhu L, Ju L, Zhao W. Fast high-order compact exponential time differencing Runge–Kutta methods for second-order semilinear parabolic equations. J Sci Comput 2016;67:1043–65.

[6] Wang X, Kou J, Gao H. Linear energy stable and maximum principle preserving semi-implicit scheme for Allen–Cahn equation with double well potential. Commun Nonlinear Sci Numer Simul 2021;98:105766.

[7] Wu C, Feng X, He Y, Qian L. A second-order strang splitting scheme with exponential integrating factor for the Allen–Cahn equation with logarithmic Flory–Huggins potential. Commun Nonlinear Sci Numer Simul 2023;117:106983.

[8] Li Y, Yu Q, Ham S, Kwak S, Lee C, Kim J. A phase-field model without artificial curvature effect for the crystal growth simulation. Int J Heat Mass Transf 2023;203:123847.

[9] Yang X. Efficient linear, stabilized, second-order time marching schemes for an anisotropic phase field dendritic crystal growth model. Comput Meth Appl Mech Eng 2019;347:316–39.

[10] Sun D, Xing H, Dong X, Han Y. An anisotropic lattice Boltzmann-phase field scheme for numerical simulations of dendritic growth with melt convection. Int J Heat Mass Transf 2019;133:1240–50.

[11] Shah A, Sabir M, Bastian P. An efficient time-stepping scheme for numerical simulation of dendritic crystal growth. Eur J Comput Mech 2016;25(6):475–88.

[12] Yin Y, Li Y, Wu K, Zhou J. Numerical simulation of solidification microstructure based on adaptive octree grids. Arch Foundry Eng 2016;16(2):33–40.

[13] Sakane S, Takaki T, Rojas R, Ohno M, Shibuta Y, Shimokawabe T, et al. Multi-GPUs parallel computation of dendrite growth in forced convection using the phase-field-lattice Boltzmann model. J Cryst Growth 2017;474:154–9.

[14] Sakane S, Aoki T, Takaki T. Parallel GPU-accelerated adaptive mesh refinement on two-dimensional phase-field lattice Boltzmann simulation of dendrite growth. Comput Mater Sci 2022;211:111507.

[15] Mirzadeh M, Guittet A, Burstedde C, Gibou F. Parallel level-set methods on adaptive tree-based grids. J Comput Phys 2016;322:345–64.

[16] Natsume Y, Ohsasa K. Cellular automaton modeling of dendritic growth using a multi-grid method. In: IOP conference series: Materials science and engineering, vol. 84 no. 1. IOP Publishing; 2015, 012050.

[17] Dobravec T, Mavrič B, Šarler B. Phase field modelling of dendritic solidification by using an adaptive meshless solution procedure. In: IOP conference series: Materials science and engineering, vol. 861 no. 1. IOP Publishing; 2020, 012060.

[18] Guo Z, Xiong SM. Study of dendritic growth and coarsening using a 3-D phase field model: Implementation of the Para-AMR algorithm. In: IOP conference series: Materials science and engineering, vol. 84 no. 1. IOP Publishing; 2015, 012067.

[19] Wang Y, Xiao X, Feng X. An accurate and parallel method with post-processing boundedness control for solving the anisotropic phase-field dendritic crystal growth model. Commun Nonlinear Sci Numer Simul 2022;115:106717.

[20] Jeong D, Kim J. Fast and accurate adaptive finite difference method for dendritic growth. Comput Phys Commun 2019;236:95–103.

[21] Ham S, Li Y, Jeong D, Lee C, Kwak S, Hwang Y, et al. An explicit adaptive finite difference method for the Cahn–Hilliard equation. J Nonlinear Sci 2022;32(6):32–80.

[22] Rosam J, Jimack PK, Mullis AA. Fully adaptive time and space discretisation method for phase-field simulation of binary alloy solidification. J Comput Phys 2007;225(2):1271–87.

[23] Li Y, Lee HG, Kim J. And accurate operator splitting method for phase-field simulations of crystal growth. J Cryst Growth 2011;321(1):176–82.

[24] Karma A, Rappel WJ. Quantitative phase-field modeling of dendritic growth in two and three dimensions. Phys Rev E 1998;57(4):4323–49.

[25] Yang SD, Lee HG, Kim J. A phase-field approach for minimizing the area of triply periodic surfaces with volume constraint. Comput Phys Commun 2010;181(6):1037–46.