Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 319 (2017) 194–216
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

An efficient and stable compact fourth-order finite difference scheme for the phase field crystal equation

Yibao Li[a], Junseok Kim[b,*]

[a] *School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China*
[b] *Department of Mathematics, Korea University, Seoul 02841, Republic of Korea*

Received 10 December 2016; received in revised form 13 January 2017; accepted 16 February 2017
Available online 4 March 2017

Highlights

- We propose an efficient and stable compact fourth-order method for the phase field crystal equation.
- The proposed numerical method with second order accuracy of time exhibits excellent stability.
- We implement the compact scheme in the adaptive mesh refinement framework.
- An adaptive time step method for the phase field crystal model is also presented.

Abstract

In this paper, we present a high-order accurate compact scheme for the phase field crystal model in two- and three-dimensional spaces. The proposed scheme is derived by combining a fourth-order compact finite difference formula in space and a backward differentiation for the time derivative term, which is second-order accurate in time. Furthermore, a nonlinearly stabilized splitting scheme is used and thus a larger time step can be allowed. Since the equations at the implicit time level are nonlinear, we introduce a Newton-type iterative method and employ a fast and efficient nonlinear multigrid solver to solve the resulting discrete system. In particular, we implement the compact scheme in the adaptive mesh refinement framework. An adaptive time step method for the phase field crystal model is also proposed. Various numerical experiments are presented and confirm the accuracy, stability, and efficiency of our proposed method.

Keywords: Phase-field crystal equation; Fourth-order compact scheme; Adaptive mesh refinement; Adaptive time-stepping

* Corresponding author. Fax: +82 2 929 8562.
E-mail addresses: yibaoli@mail.xjtu.edu.cn (Y. Li), cfdkim@korea.ac.kr (J. Kim).
URLs: http://gr.xjtu.edu.cn/web/yibaoli (Y. Li), http://math.korea.ac.kr/~cfdkim (J. Kim).

http://dx.doi.org/10.1016/j.cma.2017.02.022
0045-7825/

1. Introduction

The aim of the research presented in this paper is to develop a high-order accurate compact scheme, i.e., second-order accurate in time and fourth-order accurate in space, for the phase field crystal model [1,2]:

$$\frac{\partial \phi}{\partial t}(\mathbf{x}, t) = M \Delta \mu(\mathbf{x}, t), \tag{1}$$

$$\mu(\mathbf{x}, t) = \phi^3(\mathbf{x}, t) + (1-\epsilon)\phi(\mathbf{x}, t) + 2\Delta\phi(\mathbf{x}, t) + \Delta^2\phi(\mathbf{x}, t), \quad \mathbf{x} \in \Omega, \ 0 < t \le T, \tag{2}$$

$$\phi(\mathbf{x}, t) \text{ is } \Omega - \text{periodic}, \tag{3}$$

$$\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x}), \tag{4}$$

where the phase field $\phi(\mathbf{x}, t)$ approximates the number density of atoms in a binary mixture in the domain $\Omega = \Pi_{i=1}^{d}(0, L_i) \subset \mathbb{R}^d$ ($d = 2$ or 3). μ is the chemical potential and Δ is the Laplacian operator. The temperature related parameter ϵ and the mobility M are constants. For simplicity of exposition, here we consider the periodic boundary condition for both the order parameter and the chemical potential. This model, which is related to the dynamic density functional theory of freezing [2,3], has an important advantage over many atomistic models in that the characteristic time is determined by the diffusion time scale and not by that of the atomic vibrations. This enables relatively long time processes to be captured, in contrast to atomistic models. The review by Provatas et al. [4] outlines a wide range of applications of the phase field crystal modeling framework. A particularly challenging aspect of the numerical simulation of the phase field crystal model is the severe stability restriction on the time step, because this model is a sixth-order nonlinear partial differential equation. Generally, an explicit time scheme leads to severe and impracticable time step restrictions for stability: more precisely $\Delta t \sim O(h^6)$, where Δt is the time step and h is the mesh size [5–7]. Although a fully implicit approach can use a large time step, it only has first-order accuracy in time and requires small time steps to guarantee accuracy. Many research have been done to overcome the stability restriction [6–18]. In [11], a backward Euler scheme was presented with linearized discretization for the nonlinear term ϕ^3. Without having carried out a stability analysis, the authors numerically claimed that relatively larger time steps can be used. Depending upon a Fourier stability analysis, Cheng and Warren [12] proposed a linear spectral scheme, where the nonlinear term ϕ^3 is treated explicitly. Wise et al. [13] presented a first-order accurate (in time) finite difference scheme for the phase field crystal equation model, based on a convex splitting framework. Both the unconditionally unique solvability and discrete mass conservation were proved in their paper. Later, Hu et al. [14] extended their previous study [13] and proposed a stable and efficient second-order accurate (in time) finite-difference nonlinear scheme. Gomez et al. [15] proposed a nonlinear, second-order time accurate, and unconditionally gradient stable method with the modified Crank–Nicolson method, which is not based on convex splitting. Zhang et al. [6] developed two unconditionally stable schemes coupled with an adaptive time stepping strategy for solving the phase field crystal model. In [16], an operator splitting method for solving the phase field crystal equation was proposed. The authors split the original equation into linear and nonlinear subequations, in which the linear subequation has a closed-form solution in the Fourier space and the nonlinear subequation is solved by using a Newton-type iterative method. Vignal et al. [17] used finite element discretization to propose a novel algorithm, which guarantees mass conservation, unconditional energy stability, and second-order accuracy in time. Glasner et al. [18] proposed a linear, second-order time accurate, and unconditionally gradient stable method based on linear convex splitting. In [7], a local discontinuous Galerkin method and two unconditionally energy stable schemes were presented. The semi-discrete energy stability of this Galerkin was also proved.

The numerical simulation of the phase field crystal model is time consuming in that it needs considerable time to reach the steady state. On the other hand, a low-order accurate approach generally requires small time steps to guarantee accuracy. Therefore, a high-order accurate (in time and space) scheme is essential. Most of these numerical solutions reported to date have second-order accuracy in spatial and first- or second-order accuracy in temporal discretizations. We are not aware of any results on second-order time and fourth-order space discretization for the two- and three-dimensional phase field crystal equations. Compact difference methods achieve high-order accuracy without significantly increasing the grid points in each coordinate direction and have been developed for the Cahn–Hilliard equation [19,20]. Similar with the phase field crystal model, the Cahn–Hilliard equation was introduced to model phase separation phenomena in binary alloys and was widely studied [21–24]. Note that the

finite difference framework enables us to derive higher-order schemes by increasing the grid points in each coordinate direction straightforwardly.

However, the above-mentioned case would be a problematic if those points were close to the domain boundary. Possible ways to address this problem would be to either switch to a different solver near the boundary or to introduce a second boundary layer by interpolation [25]. This, however, generally lowers the overall accuracy of the solution. Furthermore, when either a multigrid method or an adaptive mesh refinement method is used, this problem becomes more obvious, since the same spatial operators need to be used on coarse and fine grid levels, which is much more difficult. Therefore, compact high-order schemes, which need less information from neighboring grid points, are more appropriate choices.

The objective of this study is to develop a compact difference scheme for the phase field crystal equation, which is second-order accurate in time and fourth-order accurate in space. A nonlinearly stabilized splitting scheme is used in the proposed scheme and thus a larger time step can be allowed. The equations at the implicit time level are linear and we employ a fast and efficient linear multigrid solver to solve the resulting system. In particular, we implement the compact scheme in the adaptive mesh refinement framework. An adaptive time step method for the phase field crystal model is also proposed. Several numerical experiments are performed to demonstrate the performance of our proposed algorithm.

This paper is organized as follows. Section 2 reviews the properties of the phase field crystal equation. In Section 3, we derive the fourth-order compact finite difference scheme and prove its properties. Section 4 presents the results of several numerical experiments. Section 5 contains our concluding remarks.

2. Phase field crystal model

The phase field crystal model can be derived from the mass-constrained gradient flow in a H^{-1} Hilbert space (see [13] for a mathematical definition of the space) of a free energy functional of the Swift–Hohenberg type as follows [1,5,26]:

$$\mathcal{E}(\phi) = \int_\Omega \left(\frac{1}{4}\phi^4 + \frac{1-\epsilon}{2}\phi^2 - |\nabla\phi|^2 + \frac{1}{2}(\Delta\phi)^2 \right) d\mathbf{x}.$$

The solution $\phi(\mathbf{x}, t)$ of Eqs. (1) and (2) possesses the properties that the total energy $\mathcal{E}(\phi)$ decreases with time

$$\begin{aligned} \frac{d}{dt}\mathcal{E}(\phi) &= \int_\Omega (\phi^3\phi_t + (1-\epsilon)\phi\phi_t - 2\nabla\phi\cdot\nabla\phi_t + \Delta\phi\Delta\phi_t) d\mathbf{x} \\ &= \int_\Omega \mu\phi_t d\mathbf{x} = \int_\Omega \mu M\Delta\mu d\mathbf{x} = -\int_\Omega M\nabla\mu\cdot\nabla\mu d\mathbf{x} = -\int_\Omega M|\nabla\mu|^2 d\mathbf{x} \le 0, \end{aligned}$$

and the total mass $\int_\Omega \phi\, d\mathbf{x}$ is conserved

$$\frac{d}{dt}\int_\Omega \phi\, d\mathbf{x} = \int_\Omega \phi_t\, d\mathbf{x} = M\int_\Omega \Delta\mu\, d\mathbf{x} = M\int_{\partial\Omega} \frac{\partial\mu}{\partial\mathbf{n}}\, ds = 0.$$

Here, we used the periodic boundary conditions for the chemical potential μ. In this paper, we focus on the phase field crystal model and propose a high-order accurate compact scheme with good numerical stability. For the sake of convenience, we let $M = 1$ in our work in this paper. For a mathematical point of view, the well-posedness and the regularity of solutions were studied in [27,28] and the existence and uniqueness of variational solutions for the phase field crystal equation with logarithmic nonlinear terms was studied in [29].

3. Numerical solution

In this section we present the compact finite difference schemes for the phase field crystal equation in two- and three-dimensional spaces. These schemes are similar to the scheme we presented earlier for the Cahn–Hilliard equation [19,20]. We refer the interested reader to [19,20] for a description of the compact finite difference framework that we employ to define our schemes.

3.1. Two-dimensional difference operators and respective inner products

Let $\Omega = (0, L_x) \times (0, L_y)$, where $L_x = N_x h$ and $L_y = N_y h$ where h is positive, and N_x and N_y are positive even integers. We denote a discrete computational domain by $\Omega_h = \{(x_i, y_j) : x_i = (i-0.5)h,\ y_j = (j-0.5)h,\ 1 \le i \le N_x,\ 1 \le j \le N_y\}$, which is the set of cell-centers. Let ϕ_{ij} and ψ_{ij} be the approximation of $\phi(x_i, y_j)$ and $\psi(x_i, y_j)$, respectively. The discrete differentiation operators are defined as

$$D_x\phi_{i+\frac{1}{2},j} = \frac{1}{12}\frac{\phi_{i+1,j+1}-\phi_{i,j+1}}{h} + \frac{5}{6}\frac{\phi_{i+1,j}-\phi_{i,j}}{h} + \frac{1}{12}\frac{\phi_{i+1,j-1}-\phi_{i,j-1}}{h}, \tag{5}$$

$$D_y\phi_{i,j+\frac{1}{2}} = \frac{1}{12}\frac{\phi_{i+1,j+1}-\phi_{i+1,j}}{h} + \frac{5}{6}\frac{\phi_{i,j+1}-\phi_{i,j}}{h} + \frac{1}{12}\frac{\phi_{i-1,j+1}-\phi_{i-1,j}}{h}. \tag{6}$$

Then we define the discrete l_2-inner product $(\phi, \psi)_h$ and $(\nabla_c\phi, \nabla_c\psi)_e$ as

$$\begin{aligned}(\phi, \psi)_h &= h^2 \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} \phi_{ij}\psi_{ij} \text{ and}\\ (\nabla_c\phi, \nabla_c\psi)_e &= h^2 \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} \left(D_x\phi_{i+\frac{1}{2},j} D_x\psi_{i+\frac{1}{2},j} + D_y\phi_{i,j+\frac{1}{2}} D_y\psi_{i,j+\frac{1}{2}}\right).\end{aligned} \tag{7}$$

The discrete total energy functional in the two-dimensional space is defined by

$$\mathcal{E}^h(\phi^n) = \frac{1}{4}((\phi^n)^4, \mathbf{1})_h + \frac{1-\epsilon}{2}((\phi^n)^2, \mathbf{1})_h - (\nabla_c\phi^n, \nabla_c\phi^n)_e + \frac{1}{2}(\Delta_c\phi^n, \mathbf{1})_h. \tag{8}$$

Here, $\mathbf{1}$ is a vector with all entries equal to 1. The 9-point discrete Laplacian operator, Δ_c, is defined as [19]

$$\Delta_c\phi_{ij} = \frac{4(\phi_{i-1,j}+\phi_{i+1,j}+\phi_{i,j-1}+\phi_{i,j+1}) + \phi_{i-1,j-1}+\phi_{i-1,j+1}+\phi_{i+1,j-1}+\phi_{i+1,j+1} - 20\phi_{ij}}{6h^2}. \tag{9}$$

Then using these above definitions, the discrete Laplacian $\Delta_c\phi_{ij}$ can be rewritten as

$$\Delta_c\phi_{ij} = \frac{D_x\phi_{i+\frac{1}{2},j} - D_x\phi_{i-\frac{1}{2},j}}{h} + \frac{D_y\phi_{i,j+\frac{1}{2}} - D_y\phi_{i,j-\frac{1}{2}}}{h}.$$

Furthermore, the Taylor series in two variables enables us to obtain

$$\Delta\phi_{ij} = \left(\frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2}\right)_{ij} = \Delta_c\phi_{ij} - \frac{h^2}{12}\Delta^2\phi_{ij} + O(h^4), \tag{10}$$

where $\Delta^2\phi = \Delta(\Delta\phi)$ is the biharmonic operator.

3.2. Three-dimensional difference operators and respective inner products

Suppose the three-dimensional domain $\Omega = (0, L_x) \times (0, L_y) \times (0, L_z)$. Let N_x, N_y, and N_z be positive even integers, and let $h = L_x/N_x = L_y/N_y = L_z/N_z$ be the uniform mesh size. As in the two-dimensional case, ϕ_{ijk} and ψ_{ijk} are defined in the cell-centers and are the approximations of $\phi(x_i, y_j, z_k)$ and $\psi(x_i, y_j, z_k)$, respectively. We define the following discrete inner products:

$$(\phi, \psi)_h = h^3 \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} \phi_{ijk}\psi_{ijk}, \tag{11}$$

$$\begin{aligned}(\nabla_c\phi, \nabla_c\psi)_e = h^3 \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} &\Big(D_x\phi_{i+\frac{1}{2},jk} D_x\psi_{i+\frac{1}{2},jk} + D_y\phi_{i,j+\frac{1}{2},k} D_y\psi_{i,j+\frac{1}{2},k}\\ &+ D_z\phi_{ij,k+\frac{1}{2}} D_z\psi_{ij,k+\frac{1}{2}}\Big),\end{aligned} \tag{12}$$

where the discrete differentiation operator $D_x\phi_{i+\frac{1}{2},jk}$, $D_y\phi_{i,j+\frac{1}{2},k}$, and $D_z\phi_{i,j,k+\frac{1}{2}}$ are defined as

$$\begin{aligned}
D_x\phi_{i+\frac{1}{2},jk} &= [128(\phi_{i+1,jk}-\phi_{ijk})+11(\phi_{i+1,j+1,k}-\phi_{i,j+1,k}+\phi_{i+1,j-1,k}-\phi_{i,j-1,k}\\
&\quad+\phi_{i+1,j,k+1}-\phi_{i,j,k+1}+\phi_{i+1,j,k-1}-\phi_{ij,k-1})\\
&\quad+2(\phi_{i+1,j+1,k+1}-\phi_{i,j+1,k+1}+\phi_{i+1,j-1,k+1}-\phi_{i,j-1,k+1}\\
&\quad+\phi_{i+1,j+1,k-1}-\phi_{i,j+1,k-1}+\phi_{i+1,j-1,k-1}-\phi_{i,j-1,k-1})]/(180h),\\
D_y\phi_{i,j+\frac{1}{2},k} &= [128(\phi_{i,j+1,k}-\phi_{ijk})+11(\phi_{i+1,j+1,k}-\phi_{i+1,j,k}+\phi_{i-1,j+1,k}-\phi_{i-1,j,k}\\
&\quad+\phi_{i,j+1,k+1}-\phi_{i,j,k+1}+\phi_{i,j+1,k-1}-\phi_{ij,k-1})\\
&\quad+2(\phi_{i+1,j+1,k+1}-\phi_{i+1,j,k+1}+\phi_{i-1,j+1,k+1}-\phi_{i-1,j,k+1}\\
&\quad+\phi_{i+1,j+1,k-1}-\phi_{i+1,j,k-1}+\phi_{i-1,j+1,k-1}-\phi_{i-1,j,k-1})]/(180h),
\end{aligned}$$

and

$$\begin{aligned}
D_z\phi_{i,j,k+\frac{1}{2}} &= [128(\phi_{i,j,k+1}-\phi_{ijk})+11(\phi_{i,j+1,k+1}-\phi_{i,j+1,k}+\phi_{i,j-1,k+1}-\phi_{i,j-1,k}\\
&\quad+\phi_{i+1,j,k+1}-\phi_{i+1,j,k}+\phi_{i-1,j,k+1}-\phi_{i-1,j,k})\\
&\quad+2(\phi_{i+1,j+1,k+1}-\phi_{i+1,j+1,k}+\phi_{i+1,j-1,k+1}-\phi_{i+1,j-1,k}\\
&\quad+\phi_{i-1,j+1,k+1}-\phi_{i-1,j+1,k}+\phi_{i-1,j-1,k+1}-\phi_{i-1,j-1,k})]/(180h).
\end{aligned}$$

The 27-point discrete Laplacian operator in three-dimensional space [20] is defined as

$$\begin{aligned}
\Delta_c\phi_{ijk} &= [14(\phi_{i+1,jk}+\phi_{i-1,jk}+\phi_{i,j+1,k}+\phi_{i,j-1,k}+\phi_{ij,k+1}+\phi_{ij,k-1})\\
&\quad+3(\phi_{i+1,j+1,k}+\phi_{i+1,j-1,k}+\phi_{i+1,j,k+1}+\phi_{i+1,j,k-1}+\phi_{i-1,j+1,k}+\phi_{i-1,j-1,k}\\
&\quad+\phi_{i-1,j,k+1}+\phi_{i-1,j,k-1}+\phi_{i,j+1,k+1}+\phi_{i,j-1,k-1}+\phi_{i,j+1,k-1}+\phi_{i,j-1,k+1})\\
&\quad+\phi_{i-1,j+1,k+1}+\phi_{i-1,j-1,k-1}+\phi_{i-1,j+1,k-1}+\phi_{i-1,j-1,k+1}+\phi_{i+1,j+1,k+1}\\
&\quad+\phi_{i+1,j-1,k-1}+\phi_{i+1,j+1,k-1}+\phi_{i+1,j-1,k+1}-128\phi_{ijk}]/(30h^2).
\end{aligned}\tag{13}$$

Then using the above definitions, we can rewrite the discrete Laplacian $\Delta_c\phi_{ijk}$ as

$$\Delta_c\phi_{ijk} = \frac{D_x\phi_{i+\frac{1}{2},j,k}-D_x\phi_{i-\frac{1}{2},j,k}}{h}+\frac{D_y\phi_{i,j+\frac{1}{2},k}-D_y\phi_{i,j-\frac{1}{2},k}}{h}+\frac{D_z\phi_{i,j,k+\frac{1}{2}}-D_z\phi_{i,j,k-\frac{1}{2}}}{h}.$$

As in the two-dimensional case, we can obtain the high-order Laplace approximation at node ijk:

$$\Delta\phi_{ijk} = \left(\frac{\partial^2\phi}{\partial x^2}+\frac{\partial^2\phi}{\partial y^2}+\frac{\partial^2\phi}{\partial z^2}\right)_{ijk} = \Delta_c\phi_{ijk}-\frac{h^2}{12}\Delta^2\phi_{ijk}+O(h^4).\tag{14}$$

Note that the definition of the discrete total energy functional in three-dimensional space proceeds similarly to that in the two-dimensional case.

3.3. Compact finite difference scheme for phase field crystal model

Now, we derive the fourth-order spatially accurate and practically stable compact finite difference scheme for the phase field crystal model. As a preparatory step, we rewrite Eq. (2) as

$$\mu(\mathbf{x},t) = \phi^3(\mathbf{x},t)+(1-\epsilon)\phi(\mathbf{x},t)+\Delta\kappa(\mathbf{x},t),\tag{15}$$

$$\kappa(\mathbf{x},t) = 2\phi(\mathbf{x},t)+\Delta\phi(\mathbf{x},t).\tag{16}$$

Here κ holds the periodic boundary condition because we assumed this condition for ϕ and μ. For simplicity, we describe the schemes in two-dimensional space. Three-dimensional discretization is defined analogously. Let ϕ_{ij}^n be the approximation of $\phi(x_i, y_j, n\Delta t)$, where $\Delta t = T/N_t$ is the time step, T is the final time, and N_t is the total number

of time steps. μ_{ij}^n and κ_{ij}^n are defined in a similar fashion. We can discretize Eq. (1) as

$$\phi_t = \Delta\mu = \Delta_c\mu - \frac{h^2}{12}\Delta^2\mu + O(h^4). \tag{17}$$

Here, we have used Eq. (10) and Eq. (14) in two- and three-dimensional space, respectively. Then in Eq. (17), we can replace $\Delta\mu$ as ϕ_t because of Eq. (1):

$$\phi_t = \Delta_c\mu - \frac{h^2}{12}(\Delta\phi_t) + O(h^4) = \Delta_c\mu - \frac{h^2}{12}(\Delta_c\phi_t) + O(h^4). \tag{18}$$

Since Δ_c is the second-order approximation of the Laplacian operator, Eq. (18) is sufficient to guarantee fourth-order accuracy of the solution. Furthermore, the backward differentiation formula offers a convenient choice to obtain second-order accuracy in time. Thus, the discretization of Eq. (18) is written as:

$$\frac{1}{\Delta t}\left(\frac{3}{2}\phi^{n+1} - 2\phi^n + \frac{1}{2}\phi^{n-1}\right) = \Delta_c\mu^{n+1} - \frac{h^2}{12\Delta t}\left(\frac{3}{2}\Delta_c\phi^{n+1} - 2\Delta_c\phi^n + \frac{1}{2}\Delta_c\phi^{n-1}\right) + O(\Delta t^2 + h^4), \tag{19}$$

where $\phi^{-1} = \phi^0$.

It should be noted that Eq. (19) is a three level scheme which requires a special setting of initial step ϕ^{-1} to guarantee the second-order accuracy for every time step. In our paper $\phi^{-1} = \phi^0$ is simple and will lower the accuracy in the first time step. But for later calculations, Eq. (19) is indeed with second-order accurate. Therefore for a long simulation, second-order accuracy with respect to time can be observed.

To obtain a high-order stable scheme, we use an implicit scheme for Eq. (2):

$$\mu^{n+1} = (\phi^{n+1})^3 + (1-\epsilon)\phi^{n+1} + \Delta\kappa^{n+1}. \tag{20}$$

Substituting Eq. (14) into Eq. (20), we obtain the fourth-order form for Eq. (20):

$$\begin{aligned}\mu^{n+1} &= (\phi^{n+1})^3 + (1-\epsilon)\phi^{n+1} + \Delta_c\kappa^{n+1} - \frac{h^2}{12}\Delta^2\kappa^{n+1} + O(h^4)\\ &= (\phi^{n+1})^3 + (1-\epsilon)\phi^{n+1} + \Delta_c\kappa^{n+1} - \frac{h^2}{12}\Delta_c\left(\mu^{n+1} - (\phi^{n+1})^3 - (1-\epsilon)\phi^{n+1}\right) + O(h^4).\end{aligned} \tag{21}$$

Since $\Delta_c(\phi^{n+1})^3$ is a Laplacian for the nonlinear term $(\phi^{n+1})^3$, which is difficult to solve numerically, we replace $\Delta_c(\phi^{n+1})^3$ by $\Delta_c(\tilde{\phi}^{n+1})^3$, thereby retaining the second-order accuracy in time and fourth-order accuracy in space. Here $\tilde{\phi}^{n+1} = 2\phi^n - \phi^{n-1}$. Then Eq. (23) is rewritten as :

$$\mu^{n+1} = (\phi^{n+1})^3 + (1-\epsilon)\phi^{n+1} + \Delta_c\kappa^{n+1} - \frac{h^2}{12}\Delta_c\left(\mu^{n+1} - (\tilde{\phi}^{n+1})^3 - (1-\epsilon)\phi^{n+1}\right) + O(h^4). \tag{22}$$

For Eq. (16), we obtain the following equation:

$$\begin{aligned}\kappa^{n+1} &= 2\tilde{\phi}^{n+1} + \Delta_c\phi^{n+1} - \frac{h^2}{12}\Delta^2\phi^{n+1} + O(h^4)\\ &= 2\tilde{\phi}^{n+1} + \Delta_c\phi^{n+1} - \frac{h^2}{12}\Delta_c\left(\kappa^{n+1} - 2\tilde{\phi}^{n+1}\right) + O(h^4).\end{aligned} \tag{23}$$

Finally, our proposed schemes are written as :

$$\frac{1}{\Delta t}\left(\frac{3}{2}\phi^{n+1} - 2\phi^n + \frac{1}{2}\phi^{n-1}\right) = \Delta_c\mu^{n+1} - \frac{h^2}{12\Delta t}\left(\frac{3}{2}\Delta_c\phi^{n+1} - 2\Delta_c\phi^n + \frac{1}{2}\Delta_c\phi^{n-1}\right), \tag{24}$$

$$\mu^{n+1} = (\phi^{n+1})^3 + (1-\epsilon)\phi^{n+1} + \Delta_c\kappa^{n+1} - \frac{h^2}{12}\left(\Delta_c\mu^{n+1} - \Delta_c(\tilde{\phi}^{n+1})^3 - (1-\epsilon)\Delta_c\phi^{n+1}\right), \tag{25}$$

$$\kappa^{n+1} = 2\tilde{\phi}^{n+1} + \Delta_c\phi^{n+1} - \frac{h^2}{12}\left(\Delta_c\kappa^{n+1} - 2\Delta_c\tilde{\phi}^{n+1}\right). \tag{26}$$

We use a nonlinear multigrid method [14,30] to solve the nonlinear discrete system (24)–(26) at the implicit time level. This involves representing the discrete system (24)–(26) as

$$\mathcal{L}(\phi^{n+1}, \mu^{n+1}, \kappa^{n+1}) = \mathcal{S}(\zeta^n, \varphi^n, \xi^n),$$

where $\mathcal{L}(\phi^{n+1}, \mu^{n+1}, \kappa^{n+1})$ and $\mathcal{S}(\zeta^n, \varphi^n, \xi^n)$ are the nonlinear operator and source terms at every node, respectively:

$$\begin{aligned}\mathcal{L}(\phi^{n+1}, \mu^{n+1}, \kappa^{n+1}) &= \begin{pmatrix} 3/2 + h^2/8\Delta_c & -\Delta t\Delta_c & 0 \\ -(1-\epsilon)(1 + h^2/12\Delta_c) & 1 + h^2/12\Delta_c & -\Delta_c \\ -\Delta_c & 0 & 1 + h^2/12\Delta_c \end{pmatrix} \begin{pmatrix} \phi^{n+1} \\ \mu^{n+1} \\ \kappa^{n+1} \end{pmatrix} \\ &\quad - \begin{pmatrix} 0 \\ (\phi^{n+1})^3 \\ 0 \end{pmatrix}, \\ \mathcal{S}(\zeta^n, \varphi^n, \xi^n) &= \begin{pmatrix} (1 + h^2/12\Delta_c)(2\phi^n - \phi^{n-1}/2) \\ h^2/12\Delta_c(\tilde{\phi}^{n+1})^3 \\ 2\tilde{\phi}^{n+1} + h^2/6\Delta_c\tilde{\phi}^{n+1} \end{pmatrix}.\end{aligned}$$

Since $(\phi^{n+1})^3$ is nonlinear with respect to ϕ^{n+1}, we use the Newton-type iterative method to linearize $(\phi^{n+1})^3$ as:

$$(\phi^{n+1,m+1})^3 = (\phi^{n+1,m})^3 + 3(\phi^{n+1,m})^2(\phi^{n+1,m+1} - \phi^{n+1,m}).$$

Starting with the initial guess $\phi^{n+1,0} = \tilde{\phi}^{n+1}$, a Newton-type iterative method is applied until the relative l_2 normal of the difference of $\phi^{n+1,m}$ and $\phi^{n+1,m+1}$ is smaller than a given tolerance tol. Then we let $\phi^{n+1} = \phi^{n+1,m+1}$. For additional details and background material of the multigrid method, please refer to the reference text [31]. Despite it being difficult to exactly prove that our proposed method has a good stability property, the results of the numerical tests presented in Section 4 indicate that our proposed scheme allows much larger time steps.

Our proposed method also satisfies mass conservation, i.e., $(\phi^{n+1}, \mathbf{1})_h = (\phi^n, \mathbf{1})_h$. A similar conclusion was drawn in our previous paper [20], in which we presented the proof. Therefore, the description thereof in this paper is kept brief. Using the combination of Eq. (24) and the discrete version of the integration by parts enables us to obtain

$$\begin{aligned}&\left(\frac{3}{2}\phi^{n+1} - 2\phi^n + \frac{1}{2}\phi^{n-1}, \mathbf{1}\right)_h = \Delta t(\Delta_c\mu^{n+1}, \mathbf{1})_h - \frac{h^2}{12}\left(\frac{3}{2}\Delta_c\phi^{n+1} - 2\Delta_c\phi^n + \frac{1}{2}\Delta_c\phi^{n-1}, \mathbf{1}\right)_h \\ &= \Delta t(\Delta_c\mu^{n+1}, \mathbf{1})_h - \frac{h^2}{8}(\Delta_c\phi^{n+1}, \mathbf{1})_h + \frac{h^2}{6}(\Delta_c\phi^n, \mathbf{1})_h - \frac{h^2}{24}(\Delta_c\phi^{n-1}, \mathbf{1})_h = 0.\end{aligned} \tag{27}$$

Here

$$\begin{aligned}(\Delta_c\mu^{n+1}, \mathbf{1})_h &= h\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}(D_x\mu_{i+\frac{1}{2},j} - D_x\mu_{i-\frac{1}{2},j}) + h\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}(D_y\mu_{i,j+\frac{1}{2}} - D_y\mu_{i,j-\frac{1}{2}}) \\ &= h\sum_{j=1}^{N_y}(D_x\mu_{N_x+\frac{1}{2},j} - D_x\mu_{\frac{1}{2},j}) + h\sum_{i=1}^{N_x}(D_y\mu_{i,N_y+\frac{1}{2}} - D_y\mu_{i,\frac{1}{2}}) = 0,\end{aligned}$$

for the periodic boundary conditions of the chemical potential and telescoping cancellation. $(\Delta_c\phi^{n+1}, \mathbf{1})_h = (\Delta_c\phi^n, \mathbf{1})_h = (\Delta_c\phi^{n-1}, \mathbf{1})_h = 0$ can be proved in a similar manner. Summing by parts for Eq. (27), we obtain $(3\phi^{n+1}, \mathbf{1})_h - (3\phi^n, \mathbf{1})_h = (\phi^n, \mathbf{1})_h - (\phi^{n-1}, \mathbf{1})_h$ which implies that $(\phi^{n+1}, \mathbf{1})_h = (\phi^n, \mathbf{1})_h$, if $(\phi^n, \mathbf{1})_h = (\phi^{n-1}, \mathbf{1})_h$. Since $(\phi^0, \mathbf{1})_h = (\phi^{-1}, \mathbf{1})_h$, we have the chain of equalities $(\phi^{n+1}, \mathbf{1})_h = (\phi^n, \mathbf{1})_h = \cdots = (\phi^0, \mathbf{1})_h$. Therefore, our proposed method satisfies the total conservation of mass.

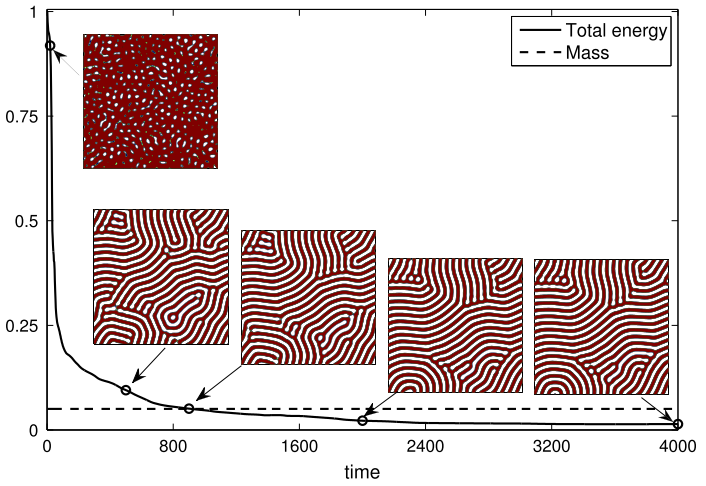


Fig. 1. Non-increasing discrete energy and mass conservation of our proposed method in two-dimensional space. The inscribed small figures are the density field ϕ at the indicated times. Note that we have normalized the total energy by the total energy at the initial time.

4. Numerical results

This section contains the results of the following numerical experiments we performed: evolution of phase field crystal in both two- and three-dimensional space, a convergence test, a test of the stability of the proposed scheme, a study of effective time step, a study of the effect of the parameter ϵ, a study of the effect of the mass of the density field, phase field crystal growth with three boundary conditions, a comparison with previous method and physical crystal, and extensions of both the adaptive mesh refinement and the adaptive time step methods. Unless otherwise specified, we take $tol = 1e{-}8$.

4.1. Evolution of phase field crystal

We start with a numerical simulation to show the evolution of the phase field crystal in two- and three-dimensional space. In the two-dimensional space, the initial condition is set as $\phi(x, y, 0) = 0.05 + 0.1\text{rand}(x, y)$ on the domain $(0, 128) \times (0, 128)$, where $\text{rand}(x, y)$ is a random number between -1 and 1. Here $\epsilon = 0.5$ and $h = 1$ are used. The calculation is run until $t = 4000$ with the time step $\Delta t = 1$. Similar to the two-dimensional case, we choose $\phi(x, y, z, 0) = 0.05 + 0.1\text{rand}(x, y, z)$ on the domain $(0, 128) \times (0, 128) \times (0, 128)$ for the three-dimensional space. The parameters are chosen to be the same as those in the two-dimensional case. The evolution of two- and three-dimensional growth of a crystal is shown in Figs. 1 and 2, respectively. Here we also combine the discrete energy and total mass evolutions of the numerical solutions. The inscribed small figures are the density field ϕ at the indicated times. Note that we normalized the total energy by the total energy at the initial time. The results in the two cases indicate that our proposed method clearly prevents the discrete energy from increasing and maintains the total conservation of mass.

4.2. Convergence test

We verify the convergence predicted by our proposed method by evolving the same initial condition as a function of time with increasingly finer values for both grid spacing and time spacing. The initial condition is chosen as:

$$\phi(x, y, 0) = 0.1 \cos\left(\frac{\pi(x - 64)}{64}\right) \sin\left(\frac{\pi(y - 64)}{64}\right) \tag{28}$$

on the domain $(0, 128) \times (0, 128)$. The simulation is evolved to time $t = 32$ with $\epsilon = 0.25$. Since there is no closed-form analytical solution for this problem, we define the Cauchy error between the numerical solution ϕ_h and the next finer grid cells covering it to be: $e_{ij}^{h/\frac{h}{2}} = \phi_{ij}^{h} - (\phi_{2i,2j}^{h/2} + \phi_{2i-1,2j}^{h/2} + \phi_{2i,2j-1}^{h/2} + \phi_{2i,2j}^{h/2})/4$.

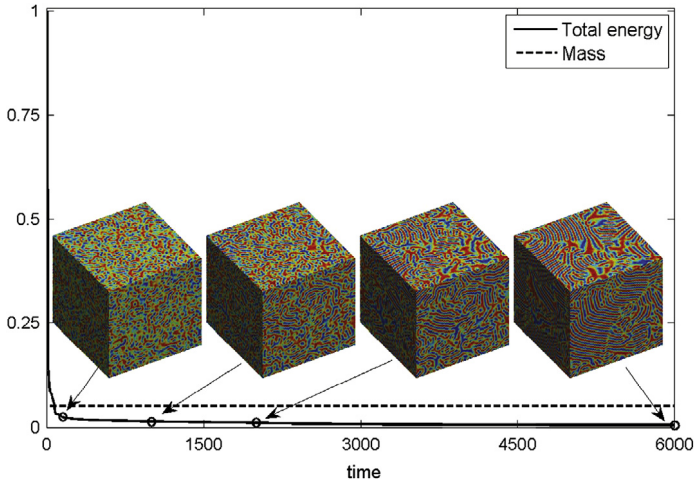


Fig. 2. Non-increasing discrete energy and mass conservation of our proposed method in three-dimensional space. The inscribed small figures are density field ϕ at the indicated times. Note that we have normalized the total energy by the total energy at the initial time.

Table 1
Error and convergence results with various grid spaces. Here $\Delta t = 2h^2$.

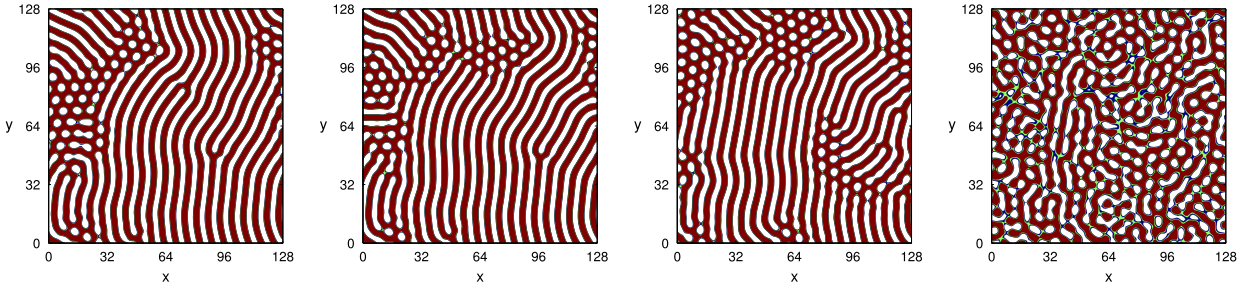
Mesh size	(32×32)–(64×64)	(64×64)–(128×128)	(128×128)–(256×256)
l_2-error	2.485E−4	1.637E−5	1.002E−6
Rate	3.92	4.03	

Note that the Cauchy Error, which is defined as the difference between any two terms of a sequence and not with the exact solution, is widely used to measure the error without knowing the exact solution. The rate of convergence is defined as the ratio of successive errors: $\log_2(\|e^h\|_2/\|e^{\frac{h}{2}}\|_2)$. The grid sizes are set as 32×32, 64×64, 128×128, and 256×256 and the time step is chosen as $\Delta t = 2h^2$. It should be noted that the residual error converges to a tolerance $1e{-}12$ in this test. The errors and rates of convergence obtained using these definitions are given in Table 1. The convergence of the results under spatial and temporal refinements is evident. Since we refined the spatial and temporal grids by a factor of 4 and 2, respectively, the ratio of successive errors can be expected to increase by a factor of 4. The results in Table 1 indicate that fourth-order accuracy with respect to space is observed, as expected from the discretization.

To obtain the convergence rate for temporal discretization, we consider the same test with the above by fixing the space step size as $h = 1$ and choosing a set of decreasing time steps $\Delta t = 2$, 1, 0.5, and 0.25. We also define the error of a grid as the discrete l_2-norm of the difference between the numerical solutions obtained by using the time step Δt and the next finer time space $\Delta t/2$. The rate of convergence is defined as the ratio of successive errors: $\log_2(\|e_{\Delta t}\|_2/\|e_{\Delta t/2}\|_2)$. The errors and rates of convergence obtained using these definitions are given in Table 2. These results suggest our proposed numerical scheme is second-order accuracy with respect to time.

4.3. Stability of the proposed scheme

In this section, we examine the stability of our proposed scheme with the same random initial conditions $\phi(x, y, 0) = 0.1 + 0.1\text{rand}(x, y)$ on a square domain $(0, 128) \times (0, 128)$ with the mesh grid 128×128. Here $\epsilon = 0.25$. A set of different time steps $\Delta t = 1$, 10, 100, and 1000 are considered. The calculations are run up to time $t = 3000$. The numerical solutions at this time are shown in Fig. 3(a–d). As can be seen, the numerical solutions do not blow up. Therefore, our proposed scheme functions effectively with large time steps. However, although a larger time step can be used, the evolution is less accurate, i.e., the evolving patterns are different from those obtained by using smaller time steps. Thus, a highly accurate numerical solution would require a smaller time step to be used.



(a) $\Delta t = 1$. (b) $\Delta t = 10$. (c) $\Delta t = 100$. (d) $\Delta t = 1000$.

Fig. 3. (a–d) Numerical solutions with different time steps at time $t = 3000$.

Table 2
Error and convergence results with various time steps. Here $h = 1$ is fixed. Here (2)–(1) means $\|e_{\Delta t}\|_2/\|e_{\Delta t/2}\|_2$ with $\Delta t = 2$.

Time step	(2)–(1)	(1)–(0.5)	(0.5)–(0.25)
l_2-error	4.671E−4	1.149E−4	2.851E−5
Rate	2.02	2.01	

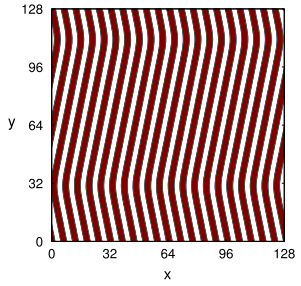
Table 3
Total number of iterative cycles and CPU time consumed until $\|\phi^n - \phi^{n+1}\|_2 < 1e{-}5$. Here different time steps are used.

Time step	$\Delta t = 1$	$\Delta t = 10$	$\Delta t = 100$	$\Delta t = 1000$
Iterative cycles	15 030	3726	2498	6787
CPU time (s)	2784.24	1615.16	1723.12	4689.48
CPU time/Iterative cycles	0.1852	0.2307	0.6898	0.6910

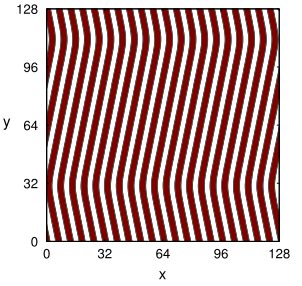
Next, we examine the steady solution of our proposed method by using different time steps $\Delta t = 1$, 10, 100, and 1000. Note that the steady solution is defined as ϕ^n, when $\|\phi^n - \phi^{n-1}\|_2 < 1e{-}5$. The same initial condition and parameters are used. Inspection of the steady solutions in Fig. 4(a–d) reveals that all these solutions are similar even when different time steps are used. Observing the evolutions of energy in Fig. 4(e), we can find that the energies for the mentioned four cases decrease rapidly initially and are followed by slow evolution at later time. Furthermore, in the early stage of the evolution, the difference is obvious when using large time steps comparing with the small time step. But in the late stage, the difference is no significant. It should be noted that much large time step creates the oscillation of energy. The total number of iterative cycles and the CPU time listed in Table 3 indicate that the CPU time is non-decreasing by a factor of 10 as the step is increased by a factor of 10. Furthermore, a much larger time step leads to increased CPU time because additional iterative cycles are required to solve Eqs. (24)–(26) in one time step. On the other hand, when the time step is small, although we need fewer iterative cycles to solve Eqs. (24)–(26) for one time step, the total CPU time is even higher because the simulation of the phase field crystal model requires considerable time to reach a steady state.

4.4. Study of effective time step

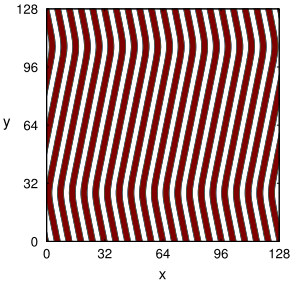
In this section, we quantify how the numerical time step affects the dynamics of phase field and choose a suitable time step to reduce the computational time. With the same initial condition in Section 4.3, we compare results at the same energy level by using different time steps, which means the energy with each row in Fig. 5 is the same. From top to bottom, the discrete energies are chosen as $\mathcal{E}^h = -10$, $\mathcal{E}^h = -20$, $\mathcal{E}^h = -30$, and $\mathcal{E}^h = -50$, respectively. From left to right, time steps are $\Delta t = 1$, $\Delta t = 2$, $\Delta t = 10$, $\Delta t = 20$, and $\Delta t = 100$, respectively. The associated



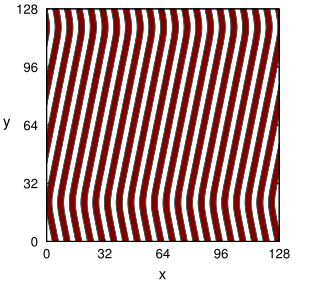
(a) $\Delta t = 1$.



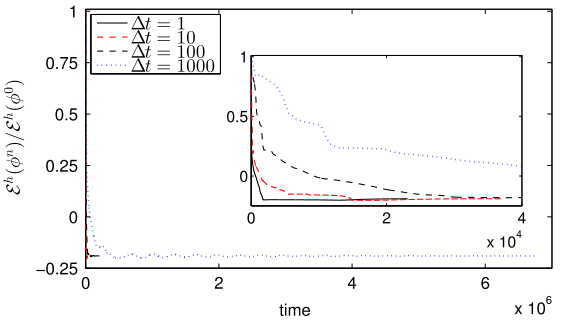
(b) $\Delta t = 10$.



(c) $\Delta t = 100$.



(d) $\Delta t = 1000$.



(e) Evolution of energy.

Fig. 4. (a–d) Steady solutions with different time steps. The steady solution is defined as ϕ^n, when $\|\phi^n - \phi^{n-1}\|_2 < 1e{-}5$. (e) Evolution of energy with four time steps.

times are listed below each figure. From Fig. 5, we can see that the density fields obtained at the same energy level by using different time step are qualitatively similar. However, the numerical times are dramatically different.

Fig. 6(a–c) show plots of the iterations, CPU time, and errors versus a range of time steps, respectively. Note that the error is defined as $(\|\phi(\cdot, \mathcal{E}^h = \mathcal{E}_0^h, \Delta t) - \phi^{ref}\|_2)/\|\phi^{ref}\|_2$. Here $\phi(\cdot, \mathcal{E}^h = \mathcal{E}_0^h, \Delta t)$ is the numerical solution which is obtained with the time step Δt at $\mathcal{E}^h = \mathcal{E}_0^h$. We chose $\mathcal{E}_0^h = -10$. ϕ^{ref} is the reference numerical solution which is obtained with the fine time step $\Delta t = 1$. As can be observed from Fig. 6, as time step is increased, the iterations and CPU time decrease rapidly and increase slowly, while the error increases as we decrease the time step. Furthermore the error is below 10% when $\Delta t \leq 20$. Note that the chosen time step is certainly related to ϵ, mesh size h, and the mass of density field. However, in practice, we cannot consider all related factors and cannot accurately give the value of Δt. Therefore, we will simply use $1 \leq \Delta t \leq 20$ in this work.

4.5. Effect of parameter ϵ

This section presents our study of the effect of ϵ in the phase field crystal equation on the pattern formation with different values $\epsilon = 0.04,\ 0.2$, and 1. The parameter ϵ is related to temperature [32]. The initial conditions are the same random initial conditions $\phi(x, y, 0) = 0.1 + 0.1\text{rand}(x, y)$ on a square domain $(0, 256) \times (0, 256)$ with the mesh grid 256×256. The time step is chosen as $\Delta t = 2$. The density field at time $t = 3000$ with three different values of ϵ are shown in Fig. 7. The results shown that, depending on the value of parameter ϵ, different patterns are produced, e.g., striped (subfigure on the left in Fig. 7) and hexagonal (subfigure on the right in Fig. 7).

4.6. Effect of the average density field

In this section, we verify how the average density field affects the numerical solution of the phase field crystal. Since the phase field crystal model satisfies mass conservation, we can consider this problem by performing a numerical experiment with different initial conditions. Here, the initial condition is chosen as $\phi(x, y, 0) = \phi_{ave} + \text{rand}(x, y)$ on the domain $(0, 256) \times (0, 256)$ with the mesh grid 256×256. Then, the initial condition is a random perturbation from

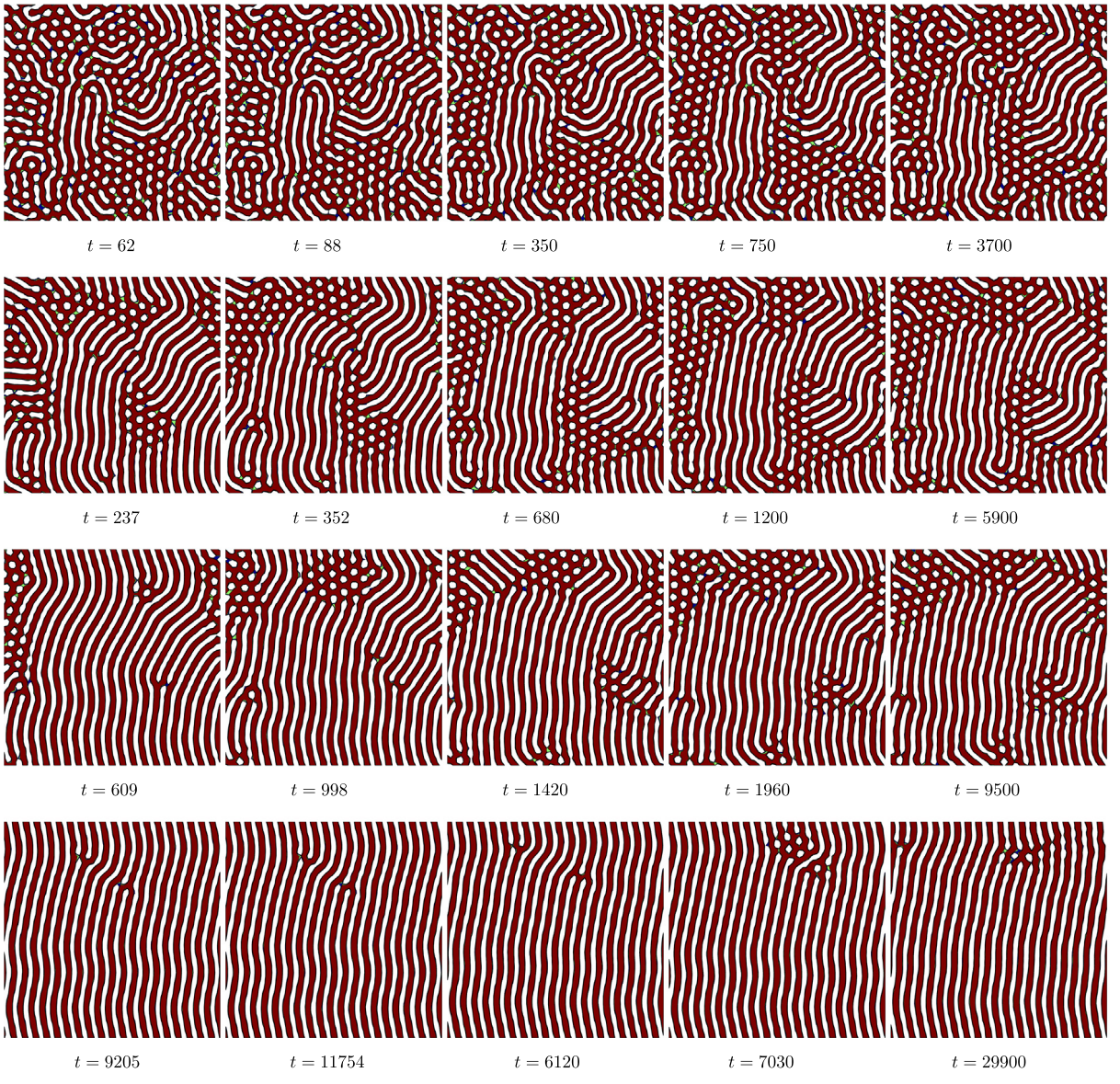


Fig. 5. Density field ϕ with different time steps. From top to bottom, the discrete energies are chosen as $\mathcal{E}^h = -10$, $\mathcal{E}^h = -20$, $\mathcal{E}^h = -30$, and $\mathcal{E}^h = -50$, respectively. From left to right, the time steps are $\Delta t = 1$, $\Delta t = 2$, $\Delta t = 10$, $\Delta t = 20$, and $\Delta t = 100$, respectively. The energy with each row is the same.

the initial average ϕ_{ave}. Simulations with $\phi_{ave} = 0.05$, 0.1, and 0.2 are run until time $t = 3000$. Here $\epsilon = 0.2$ is fixed. Fig. 8 shows the density field ϕ with different average ϕ_{ave} at time $t = 3000$. Observing these results, we can see that, depending on the value of the initial average ϕ_{ave}, the patterns are different. That is, a small value of ϕ_{ave} produces a striped pattern of the phase field crystal. As the initial average of ϕ_{ave} increases, the pattern becomes hexagonal.

4.7. Phase field crystal growth with three boundary conditions

We consider the phase field crystal growth with periodic, Neumann, and Dirichlet boundary conditions for the density field ϕ. Note that when the density field ϕ is chosen as the Neumann and Dirichlet boundary conditions, we use Neumann boundary condition for μ and κ to keep the mass conservation. We choose $\phi(x, y, t) = 0.1$ for the case of Dirichlet boundary condition. We use the same initial condition $\phi(x, y, 0) = 0.1 + 0.1\text{rand}(x, y)$ on the domain

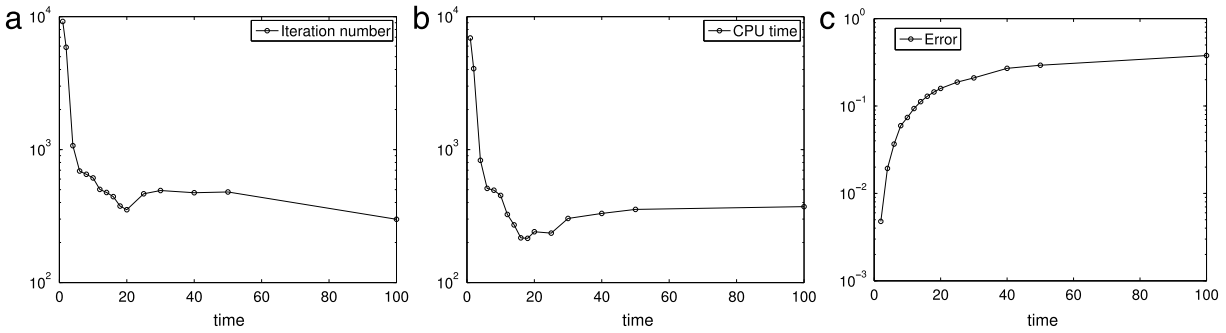
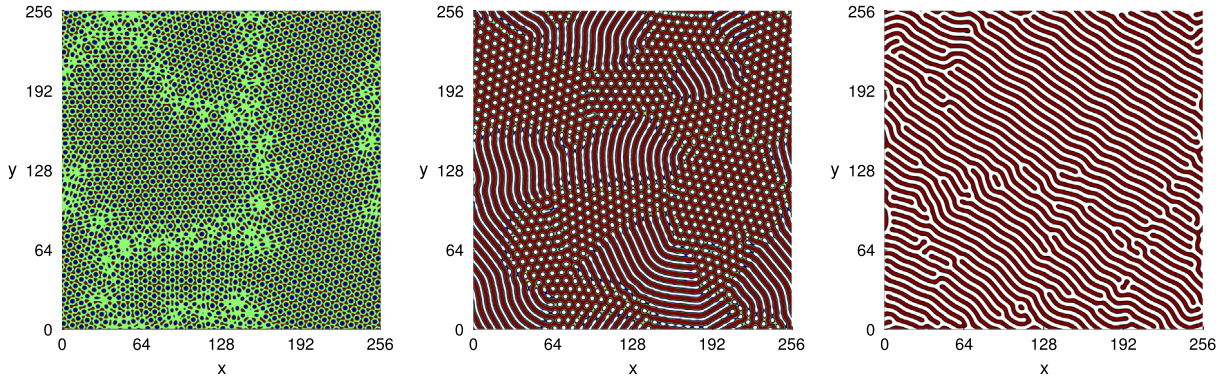
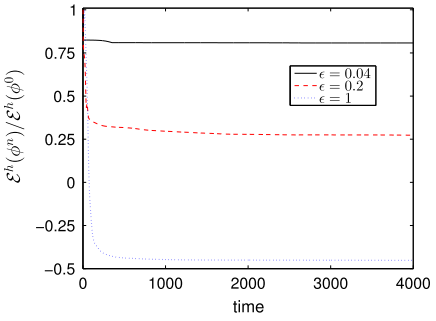


Fig. 6. Plots of (a) Iterations, (b) CPU time, and (c) errors versus the time step. Note that the error is defined as $(\|\phi(\cdot, \mathcal{E}^h = \mathcal{E}_0^h, \Delta t) - \phi^{ref}\|_2)/\|\phi^{ref}\|_2$. Here $\phi(\cdot, \mathcal{E}^h, \Delta t)$ is the numerical solution which is obtained with the time step Δt at $\mathcal{E}^h = \mathcal{E}_0^h$. We chose $\mathcal{E}_0^h = -10$. ϕ^{ref} is the reference numerical solution which is obtained with the fine time step $\Delta t = 1$.



(a) $\epsilon = 0.04$. (b) $\epsilon = 0.2$. (c) $\epsilon = 1$.

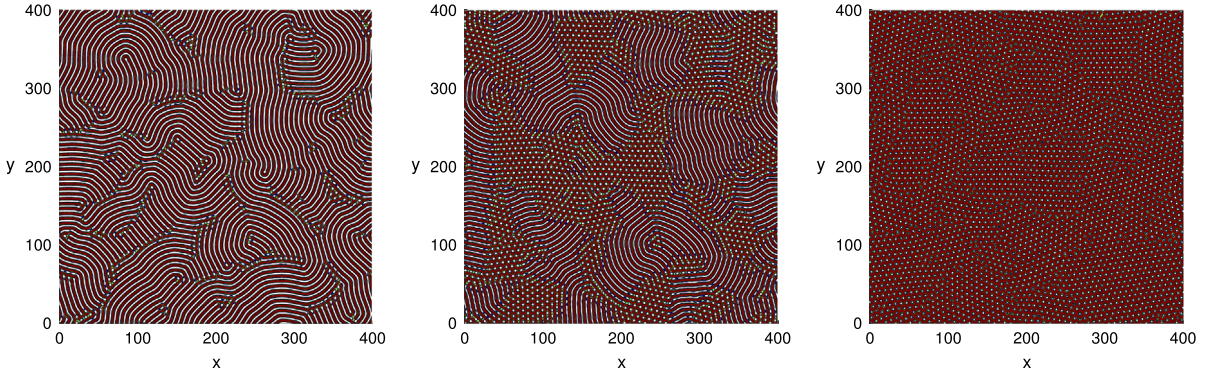


(d) Evolution of energy.

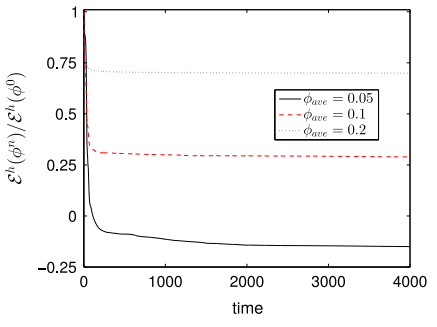
Fig. 7. Density field ϕ with different values of ϵ at time $t = 3000$. The subfigures from left to right show the results for $\epsilon = 0.04$, $\epsilon = 0.2$, and $\epsilon = 1$.

$(0, 256) \times (0, 256)$. A 256×256 mesh grid and $\epsilon = 0.25$ are chosen. Fig. 9 shows the evolutions of the phase field crystal growth using three boundary conditions with the time step $\Delta t = 1$ at times $t = 2000$ and $t = 10\,000$. We can see that our method can work well for simulating the phase field crystal growth under different boundary conditions. Furthermore, with the periodic and Neumann boundary conditions, the pattern of the phase field crystal growth is mixed. With the Dirichlet boundary condition, the mixed pattern becomes hexagonal.

Fig. 10 shows the evolution of the energy and total mass with three boundary conditions. Note that we have normalized the total energy by the total energy at the initial time. We observe that these energies are non-increasing and the total masses are conserved.



(a) $\phi_{ave} = 0.05$. (b) $\phi_{ave} = 0.1$. (c) $\phi_{ave} = 0.2$.



(d) Evolution of energy.

Fig. 8. Density field ϕ with different average values of ϕ_{ave} at time $t = 3000$. The subfigures from left to right show the results for $\phi_{ave} = 0.05$, $\phi_{ave} = 0.1$, and $\phi_{ave} = 0.2$.

4.8. Comparison with previous method and physical crystal

A well-known method for the phase field model was proposed by Hu et al. [14]. They extended their previous study [13] and presented a stable and efficient second-order accurate (in time) finite-difference nonlinear scheme. In this section, we compare the second-order scheme [14] with our proposed method. The initial condition is set as

$$\phi(x, y, 0) = 0.1 - 0.05\cos\left(\frac{\pi(x-20)}{32}\right)\sin\left(\frac{\pi(y-20)}{32}\right) + 0.05\cos^2\left(\frac{\pi(x+20)}{32}\right)\cos^2\left(\frac{\pi(y+20)}{32}\right) \tag{29}$$

on the domain $(0, 128) \times (0, 128)$. The calculation is run until $t = 100$ with the time step $\Delta t = 1$ and $\epsilon = 0.1$.

Fig. 11(a–c) show the solutions obtained by second-order scheme [14] with a 256×256 mesh grid, our proposed method with a 64×64 mesh grid, and our proposed method with a 256×256 mesh grid, respectively. Fig. 11(d) shows the plots of $\phi(x, 64)$ at time $t = 100$ and Fig. 11(e) is the closed view of Fig. 11(d). We can find that the agreement between the solutions computed by the second-order scheme with a 256×256 grid and our proposed method with 64×64 is good.

We also list the CPU time for the three cases in Table 4. From these results, we can find that in order to get more accurate numerical solutions, smaller time steps should be used in the second-order scheme. Therefore, it took much more computational time than that by the fourth-order scheme.

Next, we consider the growth of a polycrystal in a supercooled liquid. This study serves to show the applicability of our model to a physical problem. Similar numerical examples can be found in [2,14,15]. We start our simulation with a constant density field $\phi_{ave} = 0.285$ on a domain $\Omega = (0, 400) \times (0, 400)$ with a 512×512 mesh grid. As

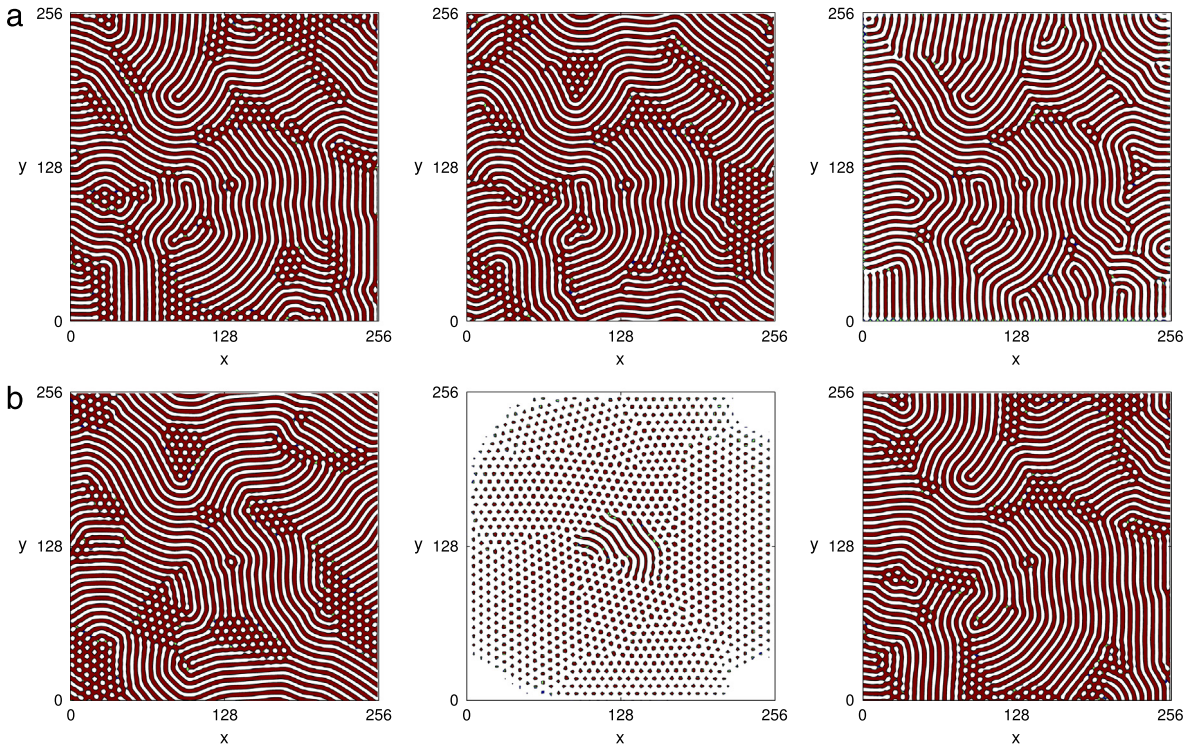


Fig. 9. The solutions with three boundary conditions (a) at time $t = 2000$ and (b) $t = 10\,000$. Form left to right, they are with periodic, Neumann, and Dirichlet boundary conditions, respectively.

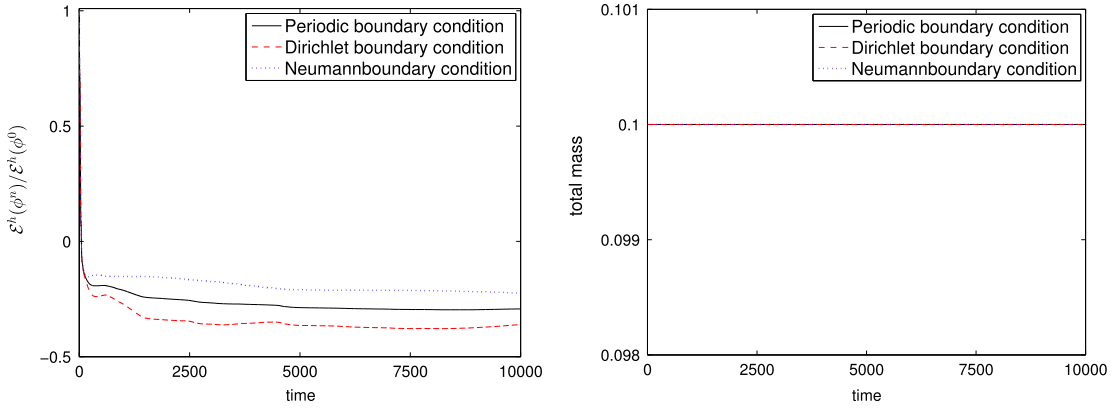


Fig. 10. Evolutions of energy and total mass with periodic, Neumann, and Dirichlet boundary conditions.

seeds for nucleation, we place four random perturbations on four small square patches with the following expression:

$$\phi(x, y) = \phi_{ave} + A\text{rand}(x, y).$$

Here A is amplitude. The centers of four small square patches are located at (100, 100), (100, 300), (300, 100), and (300, 300) and the length of each square is 10. The amplitudes are chosen as $A = 0.1$, 0.2, 0.3, and 0.4 in turn. The parameters are set to be $\Delta t = 2$ and $\epsilon = 0.25$. In this test, the boundary conditions are chosen as be homogeneous Neumann boundaries for ϕ, μ, and κ. Fig. 12 shows snapshots of the crystal microstructure at several times. Four different crystal grains grow and eventually become large enough to form grain boundaries. It also can be observed in the figures that the speed of the moving interfaces is strongly dependent on the amplitude. The larger amplitude leads to the faster polycrystal growth.

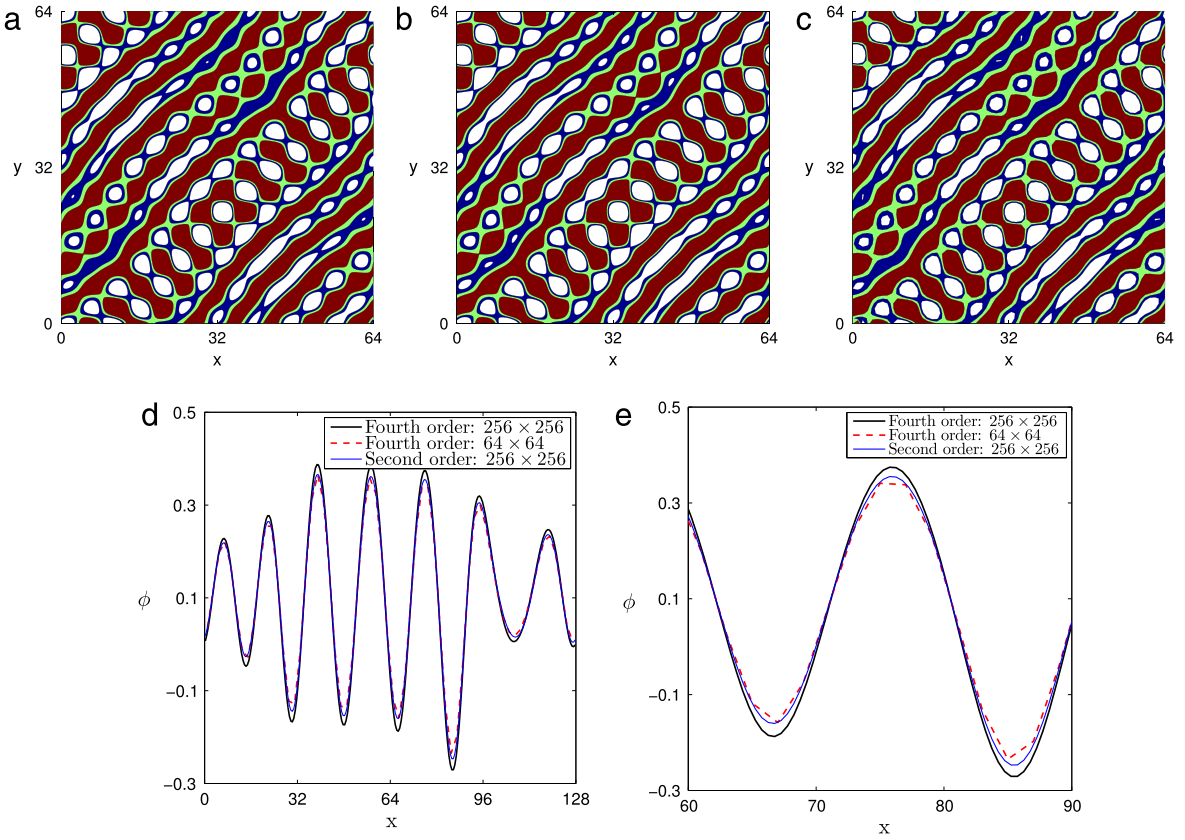


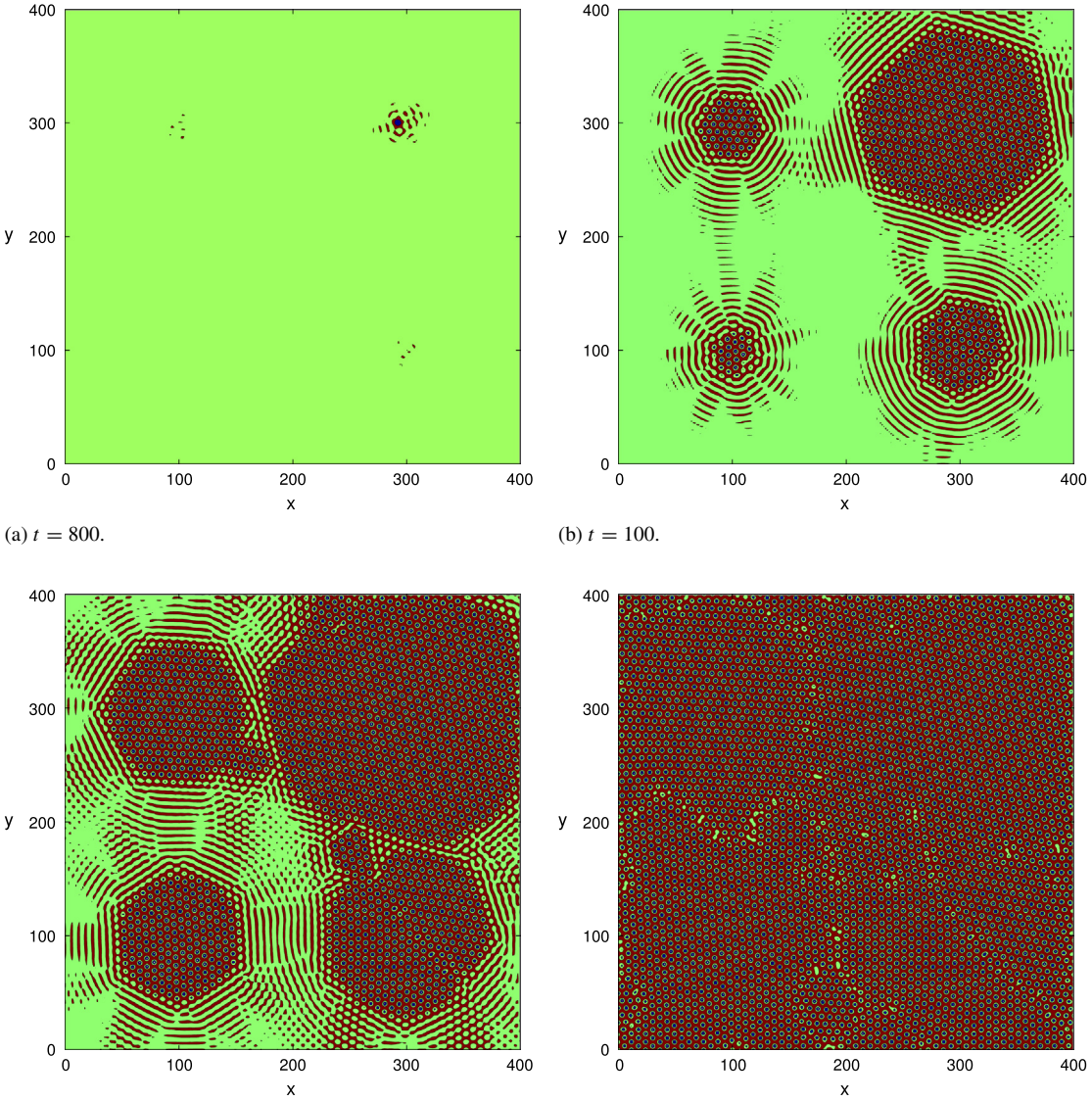
Fig. 11. (a–c) The solutions at time $t = 100$: (a) obtained by second-order scheme [14] with a 256×256 mesh grid, (b) obtained by our proposed method with a 64×64 mesh grid, and (c) obtained by our proposed method with a 256×256 mesh grid. (d) The plots of $\phi(x, 64)$ at time $t = 100$ and (e) a closed view of (d).

Table 4
Comparison of CPU time by using the second-order scheme [14] and our proposed fourth-order scheme.

Cases	Second-order scheme	Fourth-order scheme	Fourth-order scheme
Mesh size	256×256	64×64	256×256
CPU time	153.20 s	11.92 s	188.37 s

4.9. Adaptive mesh refinement for the phase field crystal equation

The adaptive mesh refinement method [33,34] is more efficient than the method based on the uniform mesh because it allows multi-resolution in interesting regions without requiring a fine grid resolution of the whole domain. This advantageous property has resulted in it being widely used [35,36]. Our method can be applied to the adaptive mesh refinement framework in a straightforward manner. The initial condition is chosen as $\phi(x, y, 0) = 0.2 - 0.05(1 + \tanh(0.1(\sqrt{(x-800)^2 + (y-800)^2} - 5)))$ in the domain $\Omega = (0, 1600) \times (0, 1600)$. We perform the simulation on the unit domain with the base mesh grids 64×64. There are five levels of refinement on the computational domain, with respective mesh spacing of 1600/64, 1600/128, 1600/256, 1600/512, 1600/1024, and 1600/2048. The other parameters are taken as $\Delta t = 2$ and $\epsilon = 0.25$. Fig. 13 shows the evolution using the adaptive mesh refinement method and the dynamical adjustment of the grid hierarchy structure around the interface transition region. The figures show that the phase field crystal develops and becomes larger as time evolves. At the same time, the grid hierarchy structure dynamically adjusts itself to capture the developing finger feature. The adaptive method uses 3600 nodes at $t = 0$ and 1 915 456 nodes at $t = 1000$, which is small compared to the 2048×2048 nodes in the uniform mesh. In the three-dimensional domain $\Omega = (0, 300) \times (0, 300) \times (0, 300)$, we simulate the phase field crystal growth



(a) $t = 800$. (b) $t = 100$.

(c) $t = 1000$. (d) $t = 1300$.

Fig. 12. Growth of a polycrystal in a supercooled liquid. The computational time is listed below each figure.

with the initial condition $\phi(x, y, z, 0) = 0.2 - 0.05(1 + \tanh(0.1(\sqrt{(x-150)^2 + (y-150)^2 + (z-150)^2} - 5)))$. Here, we use four levels of refinement to achieve the maximum grid and minimum spacings be $300/8$ and $300/256$. $\Delta t = 2$ and $\epsilon = 0.25$ are used. Fig. 14 shows the evolution of the phase field ϕ at times $t = 100$ and 300. At time $t = 0$ and $t = 300$, the nodes used in the adaptive method are 0.92% and 34.76% of the nodes in the uniform mesh ($256 \times 256 \times 256$), respectively. These simulations demonstrate that it is straightforward to apply our proposed high-order numerical scheme to the adaptive mesh refinement framework.

4.10. Adaptive time step method for the phase field crystal equation

As shown in Section 4.1, the temporal evolution of the phase field crystal equation has multiple time scales. For crystal growth simulation, an initial random perturbation evolves on a fast time scale, and whereas subsequent coarsening evolves on a very slow time scale. Therefore, if a small time step is used to capture the fast dynamics, the computation is quite costly. On the other hand, if a large time step is used, fast time evolutions may remain

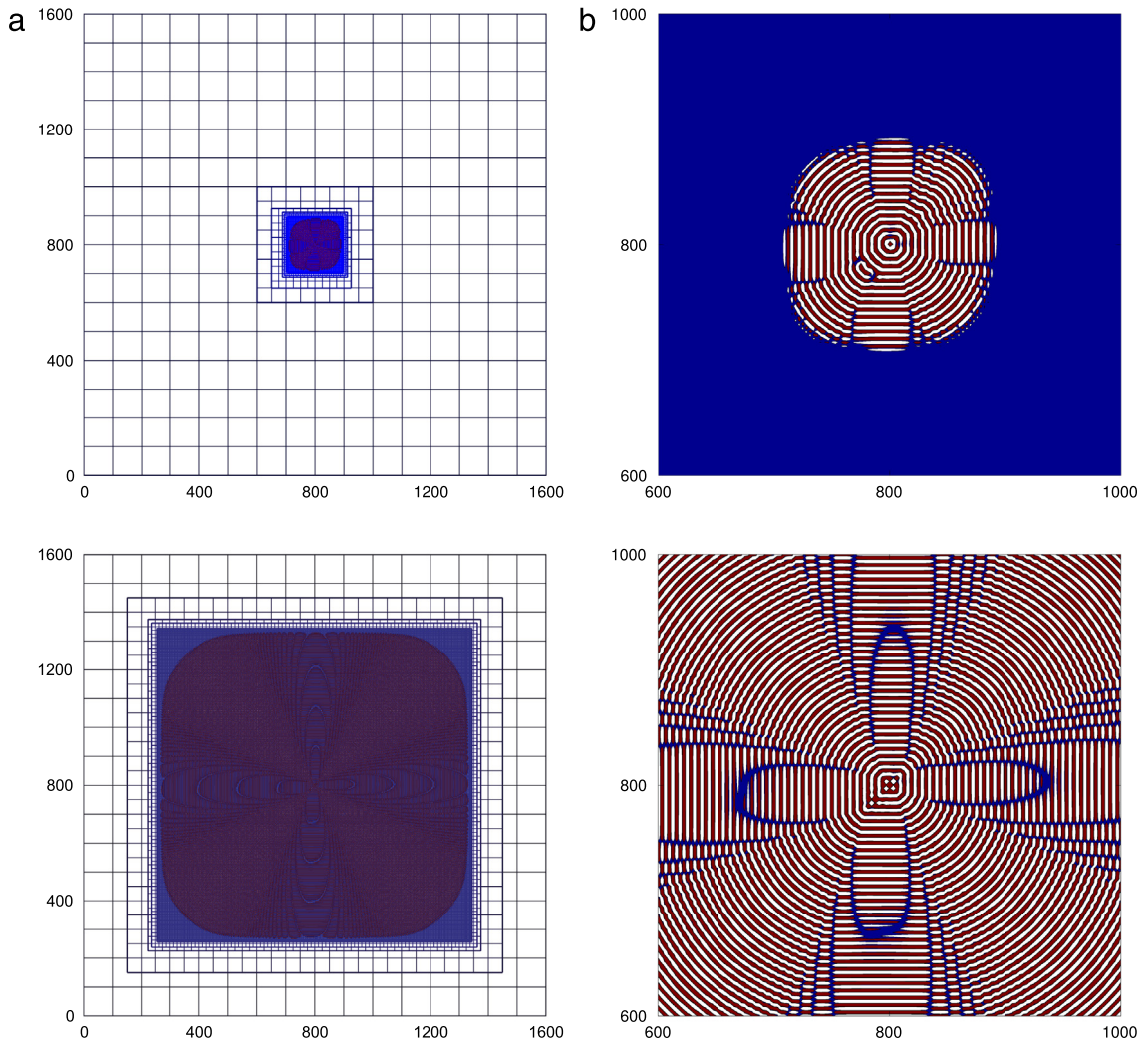


Fig. 13. Time evolution with the adaptive mesh method in two-dimensional space. (a) Density field ϕ with the adaptive mesh, (b) closeup view of density field ϕ. The upper and lower plots show the results at $t = 200$ and $t = 1000$, respectively.

undetected. Hence, it is essential to use an adaptive time step method to simulate phenomena with multiple time scales. The time adaptivity algorithm we used is based on the discrete maximum norm of the difference between the numerical solutions of two consecutive time steps, since this difference is much larger and much smaller in the fast and slow dynamics, respectively. We previously proposed a similar method [37], which has a computationally efficient adaptive time step for the Cahn–Hilliard equation. The adaptive time-step strategy is realized by using the following procedure. Set the minimum and maximum time step sizes, $\Delta t_{\min}$ and $\Delta t_{\max}$, respectively. Let Δt^0 be an initial time step (here we set $\Delta t^0 = \Delta t_{\min}$). Then, for $n = 1, 2, \ldots$, we define the next time step to be

$$\Delta t^n = \min\left[\max\left(\frac{\lambda}{\|\phi^n - \phi^{n-1}\|_\infty}, \Delta t_{\min}\right), \Delta t_{\max}\right], \tag{30}$$

where $\|\phi^n - \phi^{n-1}\|_\infty$ is the maximum value of $\|\phi^n - \phi^{n-1}\|$ and λ is a tolerance, which is important for choosing time steps. If λ is inappropriately small, it leads to time steps close to $\Delta t_{\min}$, and if λ is too large, it produces time steps close to $\Delta t_{\max}$. It should be noted that only if $\|\phi^n - \phi^{n-1}\|_\infty \geq \lambda/\Delta t_{\max}$, the time step Δt can be chosen as $\Delta t_{\max}$, which implies that Δt may be always smaller than $\Delta t_{\max}$, if the simulations are not long enough. Next, we study the effectiveness of the adaptive time step strategy by comparing the numerical solutions obtained by using both

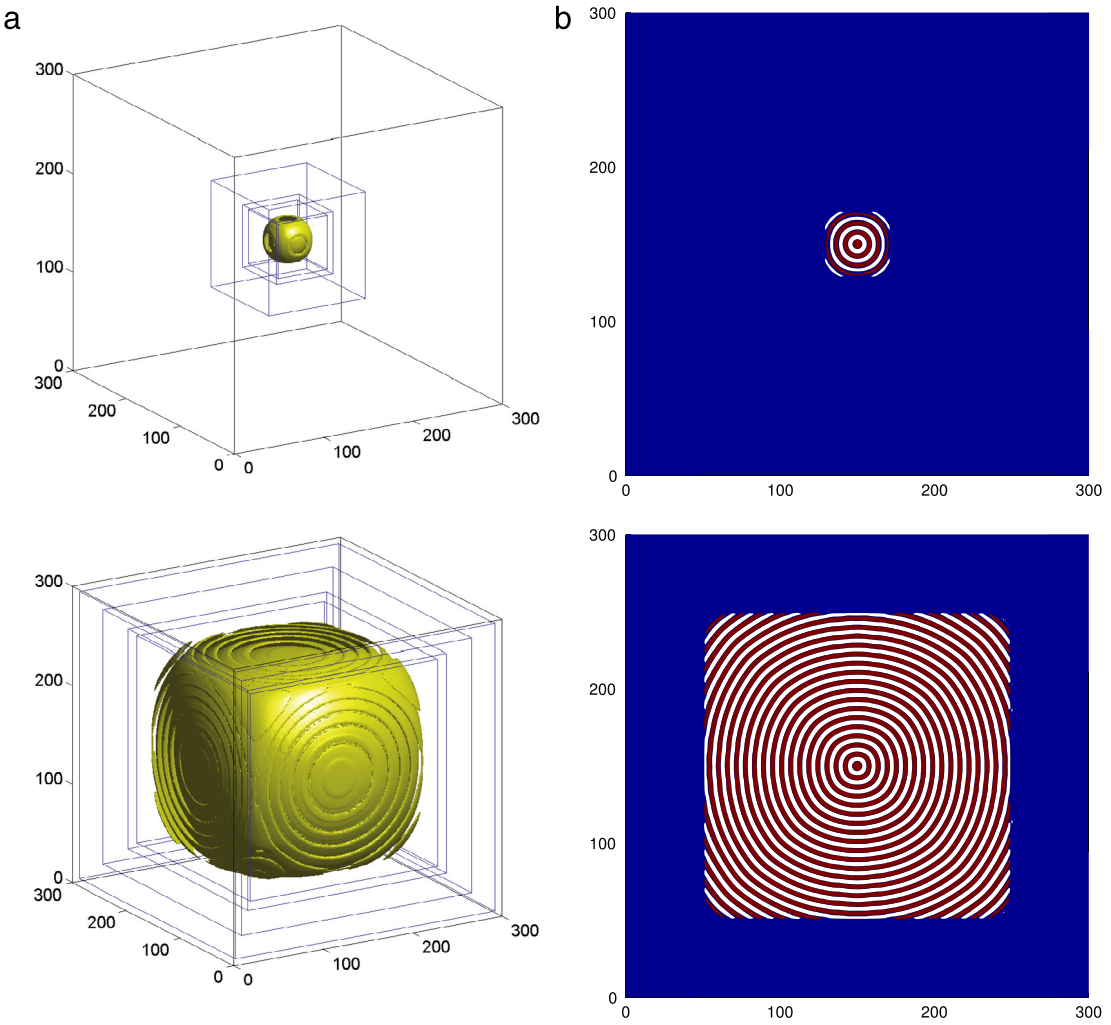


Fig. 14. Time evolution with the adaptive mesh method in three-dimensional space. (a) Isosurface of density field ϕ with the adaptive mesh, (b) slice of the density field ϕ across the plane $x = 150$. The upper and lower plots show the results at $t = 100$ and $t = 300$, respectively.

the uniform and adaptive time steps. The initial condition is the same as those in Section 4.5. Here $\epsilon = 0.2$ is chosen. In this test, we assign the following values: $\Delta t_{\min} = 1$, $\Delta t_{\max} = 50$, and $\lambda = 0.4$.

Fig. 15 compares the comparisons of the simulation results obtained by using uniform and adaptive time steps. All the simulations are run until time $T = 12\,000$. Note that the results obtained by using a uniform time step are obtained by $\Delta t = \Delta t_{\min} = 1$. Fig. 16(a) presents the energy evolution using adaptive time steps. For comparison purposes, the energy evolution obtained by using the uniform time step is plotted together with that of the adaptive time step. These results indicate that they are in good agreement. Fig. 16(b) shows the time step evolution. As can be seen, at the early stage, which corresponds to the quick decay of energy, the time step is small as expected from our proposed adaptive time step method. The time step subsequently becomes larger, since in the latter case, the energy evolves on a very slow time scale and the numerical solution approaches its steady solution.

For the three-dimensional case, we consider a similar problem with the same initial conditions given in Section 4.1. Fig. 17 compares the numerical results obtained by using both the uniform and adaptive time steps. Fig. 18(a) and (b) show the energy evolution and the used time step, respectively. Clearly, the numerical results obtained by using the adaptive time step are qualitatively in good agreement with those obtained with the uniform time step. The efficiency of our proposed adaptive time step method is apparent from the CPU time required for the two methods, as listed in Table 5. We can see that the computational efficiency is significantly improved by using the adaptive time step.

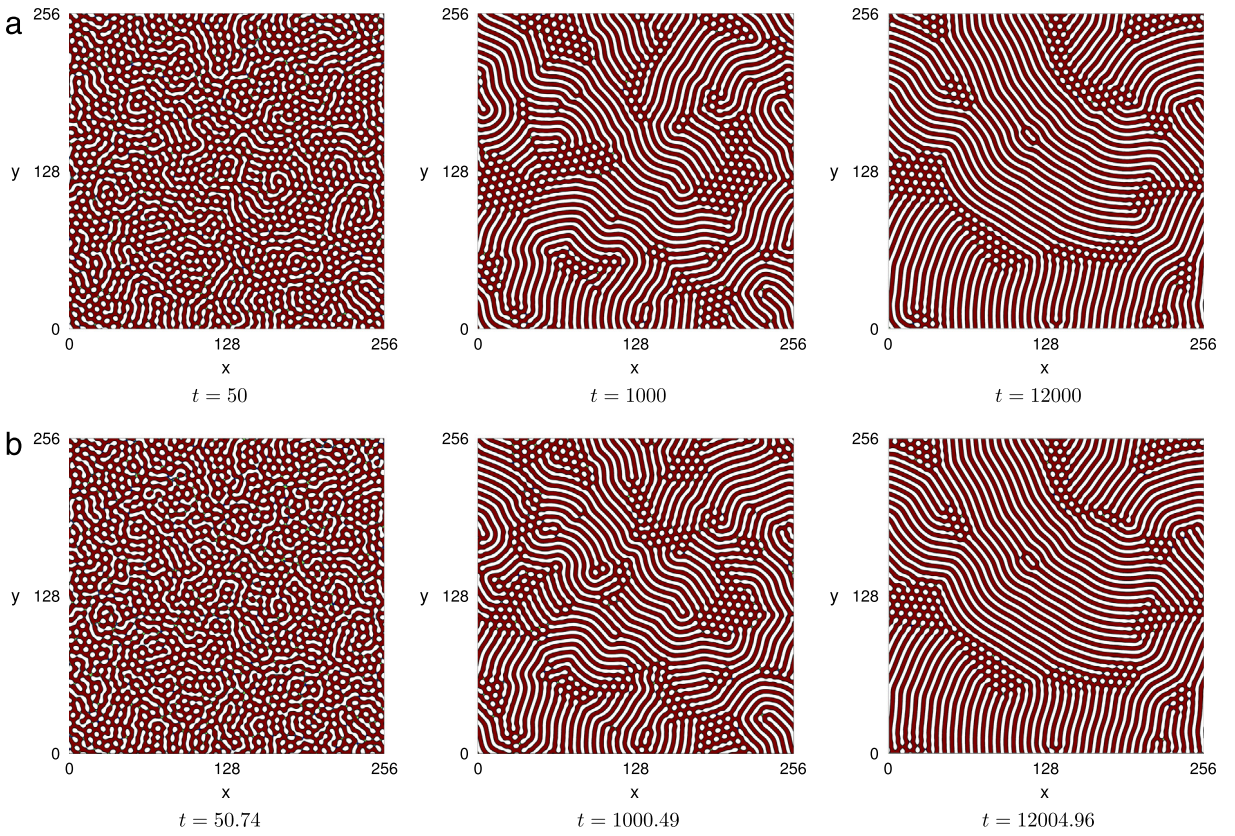


Fig. 15. Density field ϕ computed with the uniform time step (a) and adaptive time step (b) in two-dimensional space. The times are shown below each figure.

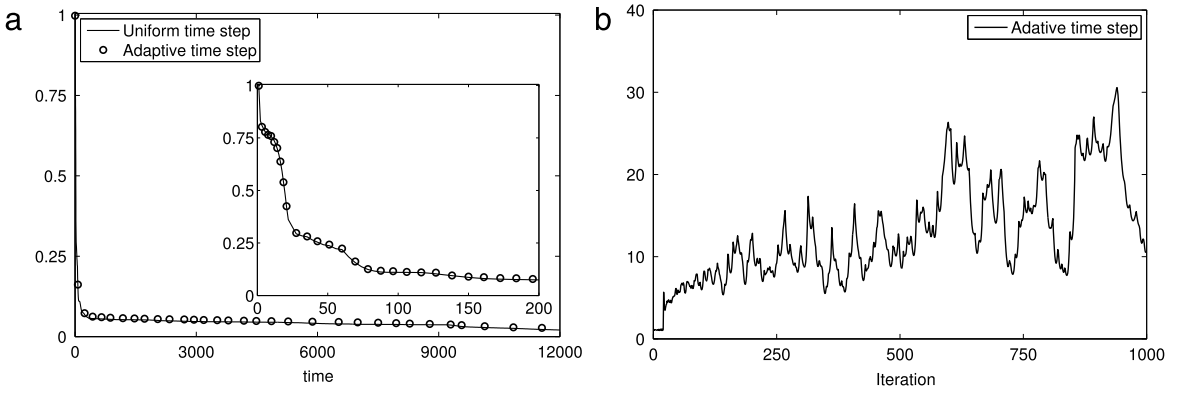


Fig. 16. Evolution of (a) energy by using both uniform and adaptive time steps. For improved visualization, the circle symbols are displayed more sparsely than the real density. Here the total energies are normalized by the total energy at the initial time. (b) Adaptive time step.

Table 5
Comparison of the used CPU time (hours) between the uniform and adaptive time steps.

Case	Mesh size	Uniform time step	Adaptive time step	$\frac{\text{Uniform time step}}{\text{Adaptive time step}}$
$2D$	256×256	6.722	0.524	12.83
$3D$	$128 \times 128 \times 128$	34.45	4.05	8.51

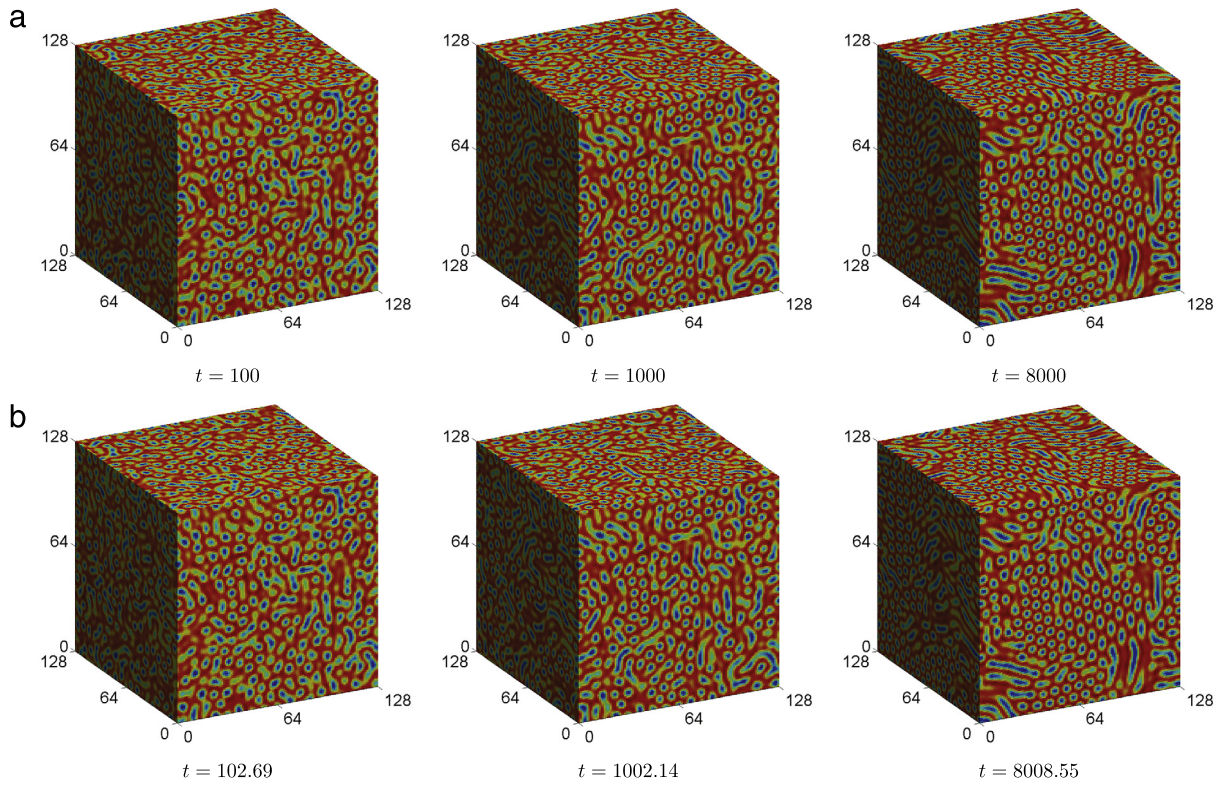


Fig. 17. Density field ϕ computed with the uniform time step (a) and adaptive time step (b) in three-dimensional space. The times are shown below each figure.

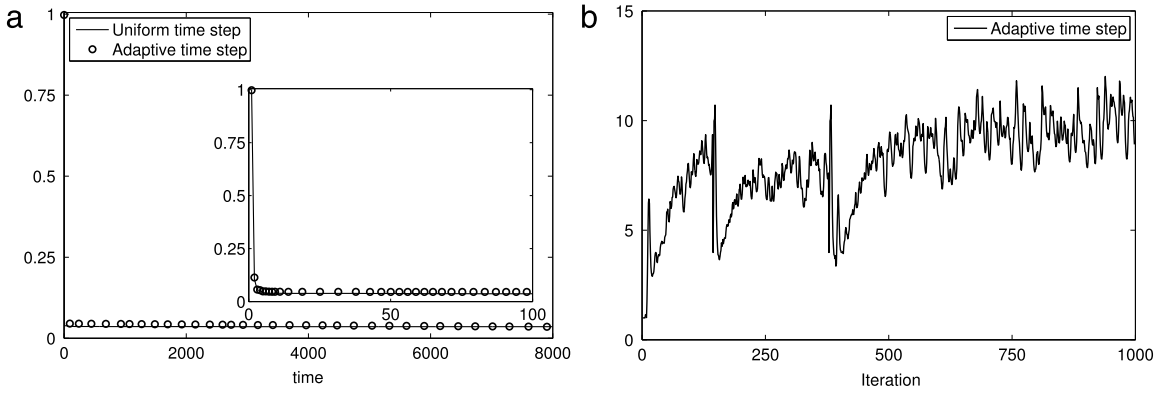


Fig. 18. Energy evolution using (a) the uniform and adaptive time steps in three-dimensional space. Note that for improved visualization, the circle symbols are displayed more sparsely than the real density. Here the total energies are normalized by the total energy at the initial time. (b) The adaptive time step.

5. Conclusions

In this paper, we presented a high-order accurate compact scheme for the phase field crystal model in two- and three-dimensional space. The proposed scheme, derived by combining a compact formula and a second-order backward differentiation for the time derivative term, is second-order accurate in time and fourth-order accurate in space. Furthermore, a nonlinearly stabilized splitting scheme is used and thus a larger time step can be allowed. We employ a fast and efficient nonlinear multigrid solver to solve the resulting system. The discrete system can preserve

total mass. In particular, we implemented the compact scheme in the adaptive mesh refinement framework. The results of various numerical experiments were presented and confirmed the accuracy, stability, and efficiency of our proposed method.

Acknowledgments

Y.B. Li is supported by Natural Science Basic Research Plan in Shaanxi Province of China (2016JQ1024), by National Natural Science Foundation of China (Nos. 11601416, 11631012). 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). The authors are grateful to the reviewers whose valuable suggestions and comments significantly improved the quality of this paper.

References

[1] K.R. Elder, M. Katakowski, M. Haataja, M. Grant, Modeling elasticity in crystal growth, Phys. Rev. Lett. 88 (24) (2002) 245701.
[2] K.R. Elder, M. Grant, Modeling elastic and plastic deformations in nonequilibrium processing using phase field crystals, Phys. Rev. E 70 (5) (2004) 051605.
[3] U.M.B. Marconi, P. Tarazona, Dynamic density functional theory of liquids, J. Comput. Phys. 110 (16) (1999) 8032–8044.
[4] N. Provatas, J.A. Dantzig, B. Athreya, P. Chan, P. Stefanovic, N. Goldenfeld, K.R. Elder, Using the phase-field crystal method in the multiscale modeling of microstructure evolution, JOM 59 (7) (2007) 83–90.
[5] M. Elsey, B. Wirth, A simple and efficient scheme for phase field crystal simulation, ESAIM Math. Model. Numer. Anal. 47 (5) (2013) 1413–1432.
[6] Z. Zhang, Y. Ma, Z. Qiao, An adaptive time-stepping strategy for solving the phase field crystal model, J. Comput. Phys. 249 (2013) 204–215.
[7] R. Guo, Y. Xu, Local discontinuous galerkin method and high order semi-implicit scheme for the phase field crystal equation, SIAM J.Sci. Comput. 38 (1) (2016) A105–A127.
[8] J.D. Clayton, J. Knap, Phase field modeling and simulation of coupled fracture and twinning in single crystals and polycrystals, Comput. Methods Appl. Mech. Engrg. 312 (2016) 447–467.
[9] M. Dehghan, V. Mohammadi, The numerical simulation of the phase field crystal (PFC) and modified phase field crystal (MPFC) models via global and local meshless methods, Comput. Methods Appl. Mech. Engrg. 298 (2016) 453–484.
[10] H.G. Lee, J.S. Kim, A simple and efficient finite difference method for the phase-field crystal equation on curved surfaces, Comput. Methods Appl. Mech. Engrg. 307 (2016) 32–43.
[11] R. Backofen, A. Rätz, A. Voigt, Nucleation and growth by a phase field crystal (PFC) model, Phil. Mag. Lett. 87 (11) (2007) 813–820.
[12] M. Cheng, J.A. Warren, An efficient algorithm for solving the phase field crystal model, J. Comput. Phys. 227 (12) (2008) 6241–6248.
[13] S.M. Wise, C. Wang, J.S. Lowengrub, An energy stable and convergent finite-difference scheme for the phase field crystal equation, SIAM J. Numer. Anal. 47 (3) (2009) 2269–2288.
[14] Z. Hu, S.M. Wise, C. Wang, J.S. Lowengrub, Stable and efficient finite-difference nonlinear-multigrid schemes for the phase field crystal equation, J. Comput. Phys. 228 (15) (2009) 5323–5339.
[15] H. Gomez, X. Nogueira, An unconditionally energy-stable method for the phase field crystal equation, Comput. Methods Appl. Mech. Engrg. 249 (2012) 52–61.
[16] H.G. Lee, J. Shin, J.-Y. Lee, First and second order operator splitting methods for the phase field crystal equation, J. Comput. Phys. 299 (2015) 82–91.
[17] P. Vignal, L. Dalcin, D.L. Brown, N. Collier, V.M. Calo, An energy-stable convex splitting for the phase-field crystal equation, Comput. Struct. 158 (2015) 355–368.
[18] K. Glasner, S. Orizaga, Improving the accuracy of convexity splitting methods for gradient flow equations, J. Comput. Phys. 315 (2016) 52–64.
[19] C. Lee, D. Jeong, J. Shin, Y.B. Li, J.S. Kim, A fourth-order spatial accurate and practically stable compact scheme for the Cahn–Hilliard equation, Physica A 409 (2014) 17–28.
[20] Y.B. Li, H.-G. Lee, B. Xia, J.S. Kim, A compact fourth-order finite difference scheme for the three-dimensional Cahn–Hilliard equation, Comput. Phys. Comm. 200 (2016) 108–116.
[21] M. Dehghan, V. Mohammadi, The numerical solution of Cahn-Hilliard (CH) equation in one, two and three-dimensions via globally radial basis functions (GRBFs) and RBFs-differential quadrature (RBFs-DQ) methods, Eng. Anal. Bound. Elem. 51 (2015) 74–100.
[22] Y. Li, D. Jeong, J. Shin, J. Kim, A conservative numerical method for the Cahn–Hilliard equation with Dirichlet boundary conditions in complex domains, Comput. Math. Appl. 65 (2013) 102–115.
[23] M. Dehghan, V. Mirzaei, A numerical method based on the boundary integral equation and dual reciprocity methods for one-dimensional Cahn-Hilliard equation, Eng. Anal. Bound. Elem. 33 (4) (2009) 522–528.
[24] M. Dehghan, Finite difference procedures for solving a problem arising in modeling and design of certain optoelectronic devices, Math. Comput. Simulation 71 (1) (2006) 16–30.
[25] G. Sutmann, B. Steffen, High-order compact solvers for the three-dimensional Poisson equation, J. Comput. Appl. Math. 187 (2006) 142–170.
[26] J. Swift, P.C. Hohenberg, Hydrodyamic fluctuations at the convective instability, Phys. Rev. A 15 (1) (1977) 319.
[27] G. Maurizio, H. Wu, Well-posedness and long-time behavior for the modified phase-field crystal equation, Math. Models Methods Appl. Sci. 24 (14) (2014) 2743–2783.

[28] A. Miranville, Sixth-order Cahn–Hilliard systems with dynamic boundary conditions, Math. Methods Appl. Sci. 38 (6) (2015) 1127–1145.
[29] A. Miranville, On the phase-field-crystal model with logarithmic nonlinear terms, Rev. R. Acad. Cienc. Exactas Fís. Nat. Ser. A Mat. RACSAM 110 (1) (2016) 145–157.
[30] U. Trottenberg, C. Oosterlee, A. Schüller, Multigrid, Academic Press, London, 2001.
[31] J.S. Kim, K. Kang, J. Lowengrub, Conservative multigrid methods for Cahn–Hilliard fluids, J. Comput. Phys. 193 (2004) 511–543.
[32] R. Backofen, K. Barmak, K.E. Elder, A. Voigt, Capturing the complex physics behind universal grain size distributions in thin metallic films, Acta Mater. 64 (2014) 72–77.
[33] A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome, A conservative adaptive projection method for the variable density incompressible Navier–Stokes equations, J. Comput. Phys. 142 (1998) 1–46.
[34] M. Sussman, A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome, An adaptive level set approach for incompressible two-phase flows, J. Comput. Phys. 148 (1999) 81–124.
[35] Y.B. Li, J.S. Kim, Phase-field simulations of crystal growth with adaptive mesh refinement, Int. J. Heat Mass Transfer 55 (2012) 7926–7932.
[36] Y.B. Li, D. Jeong, J.S. Kim, Adaptive mesh refinement for simulation of thin film flows, Meccanica 49 (2014) 239–252.
[37] Y.B. Li, Y.H. Choi, J.S. Kim, Computationally efficient adaptive time step method for the Cahn–Hilliard equation, in review, Comput. Math. Appl. (2017).