Applied Mathematical Modelling 67 (2019) 477–490

Contents lists available at ScienceDirect

Applied Mathematical Modelling

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

An efficient linear second order unconditionally stable direct discretization method for the phase-field crystal equation on surfaces

Yibao Li[a], Chaojun Luo[a], Binhu Xia[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

ARTICLE INFO

Article history:
Received 23 May 2018
Revised 31 October 2018
Accepted 7 November 2018
Available online 14 November 2018

Keywords:
Unconditionally stable
Phase-field crystal equation
Triangular surface mesh
Laplace–Beltrami operator

ABSTRACT

We develop an unconditionally stable direct discretization scheme for solving the phase-field crystal equation on surfaces. The surface is discretized by using an unstructured triangular mesh. Gradient, divergence, and Laplacian operators are defined on triangular meshes. The proposed numerical method is second-order accurate in space and time. At each time step, the proposed computational scheme results in linear elliptic equations to be solved, thus it is easy to implement the algorithm. It is proved that the proposed scheme satisfies a discrete energy-dissipation law. Therefore, it is unconditionally stable. A fast and efficient biconjugate gradients stabilized solver is used to solve the resulting discrete system. Numerical experiments are conducted to demonstrate the performance of the proposed algorithm.

1. Introduction

In this paper, a second-order accurate scheme in space and time is proposed for the phase-field crystal (PFC) model on a general surface $\mathcal{S}$ [1,2], namely,

$$\frac{\partial \phi}{\partial t}(\mathbf{v},t) = M\Delta_\tau \mu(\mathbf{v},t), \tag{1}$$

$$\mu(\mathbf{v},t) = \phi^3(\mathbf{v},t) + (1-\epsilon)\phi(\mathbf{v},t) + \Delta_\tau \kappa(\mathbf{v},t), \tag{2}$$

$$\kappa(\mathbf{v},t) = 2\phi(\mathbf{v},t) + \Delta_\tau \phi(\mathbf{v},t), \quad \mathbf{v}\in S,\ 0<t\le T, \tag{3}$$

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

* Corresponding author.
E-mail addresses: yibaoli@xjtu.edu.cn (Y. Li), cfdkim@korea.ac.kr (J. Kim).
URL: http://gr.xjtu.edu.cn/web/yibaoli (Y. Li), http://math.korea.ac.kr/~cfdkim (J. Kim)

https://doi.org/10.1016/j.apm.2018.11.012
0307-904X/

where $\phi(\mathbf{v}, t)$ is the density field on the surface, M is the mobility, ϵ is atom thickness, Δ_τ is the Laplace–Beltrami operator, $\mu(\mathbf{v}, t)$ is the chemical potential, and $\kappa(\mathbf{v}, t)$ is an auxiliary function. For simplicity of exposition, we let $M = 1$ and periodic boundary conditions are considered if the surface boundary exists. The PFC model is based on the theory of freezing [2,3]. The mathematical model can be derived using the following free energy functional [1,2], i.e.,

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

The solution of the PFC model satisfies the total energy dissipation and total mass conservation. For various applications of the PFC model, see Provatas et al. [4]. However, it is difficult to simulate phase separation kinetics on surfaces.

First, it is difficult to obtain an efficient numerical method that is accurate and stable because the PFC equation has a nonlinear term and sixth-order spatial derivatives. An explicit time scheme has severe stability restrictions. A fully implicit time step scheme can use a relatively large time step owing to its stability. However, it is first-order accurate in time and requires sufficiently small time steps to obtain accurate numerical solutions. To remove the time step restrictions, several methods have been proposed. Wise et al. [5,6] proposed first- and second-order accurate numerical methods for the PFC equation. Zhang et al. [7] proposed an unconditionally energy stable scheme and use an adaptive time step method to reduce computational cost. Gomez and Nogueira [8] presented a second-order unconditionally stable scheme. The authors in [9] applied the operator splitting method: a closed-form solution for the linear equation and a Newton-type iterative scheme for the nonlinear equation. Glasner and Orizaga [10] developed an unconditionally stable second-order time accurate scheme. Based on radial basis functions, Dehghan and Mohammadi [11] developed a numerical meshless method for the phase field crystal equation. Li and Kim [12] proposed a stable and efficient compact fourth-order method for the phase field crystal equation in two- and three-dimensional spaces. However, the above-mentioned methods require solving a nonlinear equation. Yang and Han [13] proposed provably unconditionally stable schemes for solving the PFC equation by linearizing the nonlinear term. Generally, linear elliptic equations are solved faster and are considered easier to implement compared to nonlinear equations.

Second, it is difficult to compute the Laplace–Beltrami operator on a curved surface. Existing numerical methods can be classified into two categories: explicit methods and implicit methods. The finite element method is a representative explicit method, where the Laplace–Beltrami operator is commonly expressed by the tangential gradient [14–16]. Implicit methods extend the governing equations to a high-dimensional domain and choose a narrow band embedding of the curved surface; the equation is then solved on that narrow band domain [17–19]. However, all implicit methods incur additional computational cost. To the best of the authors' knowledge, there are few studies of the PFC equation on curved surfaces. Based on the implicit method, Lee and Kim [20] presented a first-order accuracy finite difference scheme for the PFC equation. However, the computation of the PFC model requires considerable CPU time for a long-time evolution. In general, a first-order accurate method requires very small time steps to obtain accurate numerical results. Therefore, a stable high-order numerical scheme on curved surfaces is essential.

The objective of this study is to develop an efficient direct discretization for solving the PFC equation on curved surfaces. The discretization is performed using a triangular surface mesh, therefore the gradient, divergence, and Laplacian operators are defined on the surface mesh. A direct discretization method for the Cahn–Hilliard equation on a fixed surface was recently developed in [21]. Recently, the authors extended their method on an evolving surface [22]. Compared with the finite element method, the proposed method is easy to implement. In contrast with the implicit method, the additional computational cost can be reduced because the dimension of the resulting discretization scheme is the same as the dimension of the continuous problem. Recently, several studies on the second order backward differentiation formula scheme have appeared and its applications to Allen–Cahn equation [23,24] and the Cahn–Hilliard equation [25–27]. Both the energy stability and the unique solvability were preserved. Our proposed scheme, also derived by combining a backward differentiation and a direct discretization for the time and space derivative terms, respectively, is second-order accurate in time and space. A fast and efficient biconjugate gradient stabilized solver is used to solve the resulting discrete system.

The contents of this paper are as follows. In Section 2, the second-order numerical scheme for the PFC equation is derived. In Section 3, computational experiments are performed to demonstrate the feasibility of the proposed method. In Section 4, conclusions are given.

2. Numerical solution

Some basic notations regarding triangular meshes are first introduced, and a stable scheme for the PFC equation a curved surface is then proposed.

2.1. Discretizations of the Laplace–Beltrami operator

Let $S = \{\mathbf{v}_i | 1 \le i \le N_v\}$ be a triangular discretization of $\mathcal{S}$ and $F = \{T_k | 1 \le k \le N_F\}$ be the set of triangles. For a vertex $\mathbf{v} \in S$, let $\mathbf{v}_j$ be the neighboring vertices of $\mathbf{v}$ for $j = 0, 1, \ldots, p$, where $\mathbf{v}_0 = \mathbf{v}_p$. The vertices $\mathbf{v}_j$ are labeled counterclockwise. Let T_j be the triangle with vertices $\mathbf{v}$, $\mathbf{v}_j$, $\mathbf{v}_{j+1}$, and $\mathbf{G}_j = (\mathbf{v}_j + \mathbf{v}_{j+1} + \mathbf{v})/3$ (see Fig. 1). For a small regular surface $\tilde{\mathcal{S}}$ by using Green's formula, one has

$$\int_{\tilde{\mathcal{S}}} \Delta\phi d\mathbf{v} = \int_{\partial\tilde{\mathcal{S}}} \langle \nabla\phi, \ \mathbf{n} \rangle d\partial\mathbf{v}. \tag{5}$$

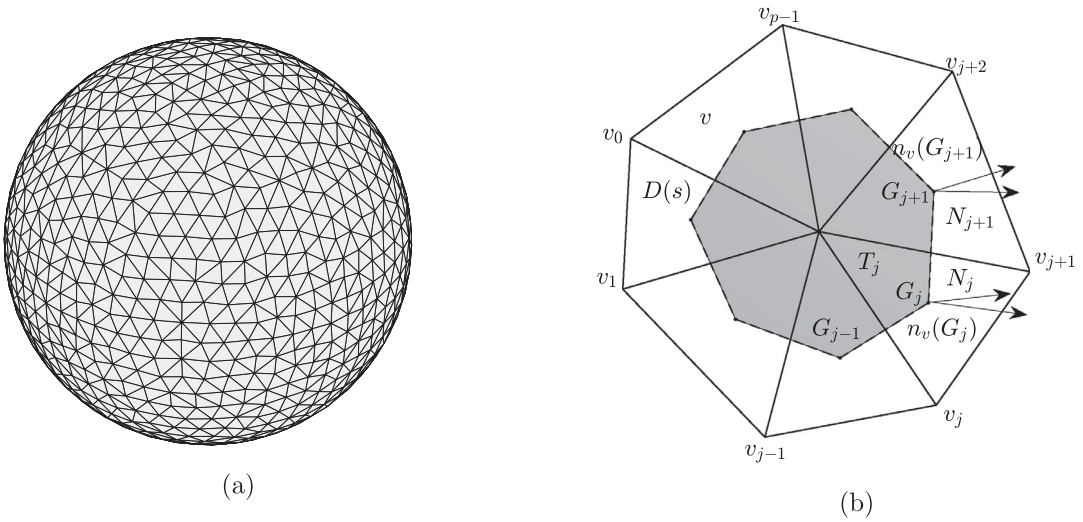

Fig. 1. (a) Triangular surface mesh. (b) Dual neighbor triangular mesh of the vertex **v** and its neighbors for evaluating the Laplace–Beltrami operators.

Here, **n** is the outer normal vector of the regular surface $\tilde{\mathcal{S}}$. It is easy to obtain that

$$\int_{\tilde{\mathcal{S}}} \Delta\phi d\mathbf{v} \approx D(\mathbf{v})\Delta_\tau\phi(\mathbf{v}), \tag{6}$$

where $D(\mathbf{v}) = \sum_{j=0}^{p-1} |\hat{T}_j|$ and $|\hat{T}_j|$ is the area of $\hat{T}_j$, which is a triangle with vertices **v,** $\mathbf{G}_j$, and $\mathbf{G}_{j+1}$. Δ_τ is a discrete Laplace–Beltrami operator. Furthermore,

$$\begin{aligned}\int_{\partial\tilde{\mathcal{S}}} \langle\nabla\phi,\ \mathbf{n}\rangle d\partial\mathbf{v} &\approx \sum_{j=0}^{p-1}(\|\mathbf{G}_{j+1}-\mathbf{G}_j\|)\int_0^1 \Big\langle q\nabla_d\phi(\mathbf{G}_j)+(1-q)\nabla_d\phi(\mathbf{G}_{j+1}),\ q\mathbf{n}_\mathbf{v}(\mathbf{G}_j)+(1-q)\mathbf{n}_\mathbf{v}(\mathbf{G}_{j+1})\Big\rangle dq\\ &= \sum_{j=0}^{p-1}\frac{\|\mathbf{G}_{j+1}-\mathbf{G}_j\|}{6}\Big(2\big\langle\nabla_d\phi(\mathbf{G}_j),\ \mathbf{n}_\mathbf{v}(\mathbf{G}_j)\big\rangle+2\big\langle\nabla_d\phi(\mathbf{G}_{j+1}),\ \mathbf{n}_\mathbf{v}(\mathbf{G}_{j+1})\big\rangle\\ &\quad+\big\langle\nabla_d\phi(\mathbf{G}_j),\ \mathbf{n}_\mathbf{v}(\mathbf{G}_{j+1})\big\rangle+\big\langle\nabla_d\phi(\mathbf{G}_{j+1}),\ \mathbf{n}_\mathbf{v}(\mathbf{G}_j)\big\rangle\Big).\end{aligned} \tag{7}$$

Here, $\|\cdot\|$ and $\langle\cdot,\cdot\rangle$ denote the normal and inner vector product, respectively. The normals $\mathbf{n}_\mathbf{v}(\mathbf{G}_j)$ and $\mathbf{n}_\mathbf{v}(\mathbf{G}_{j+1})$ are defined as

$$\mathbf{n}_\mathbf{v}(\mathbf{G}_j) = \frac{(\mathbf{G}_{j+1}-\mathbf{G}_j)\times\mathbf{N}_j}{\|(\mathbf{G}_{j+1}-\mathbf{G}_j)\times\mathbf{N}_j\|} \text{ and } \mathbf{n}_\mathbf{v}(\mathbf{G}_{j+1}) = \frac{(\mathbf{G}_{j+1}-\mathbf{G}_j)\times\mathbf{N}_{j+1}}{\|(\mathbf{G}_{j+1}-\mathbf{G}_j)\times\mathbf{N}_{j+1}\|}. \tag{8}$$

Here $\mathbf{N}_j$ and $\mathbf{N}_{j+1}$ are the normal vectors of the triangles T_j and T_{j+1}, respectively. $\nabla_d\phi(\mathbf{G}_j)$ is the approximate surface gradient of $\phi(\mathbf{G}_j)$ at the centroid $\mathbf{G}_j$ of each triangle. Eqs. (5)–(7) suggest that the discretization of the Laplace–Beltrami operator can be defined by

$$\begin{aligned}\Delta_\tau\phi(\mathbf{v}) &= \frac{1}{D(\mathbf{v})}\sum_{j=0}^{p-1}\frac{\|\mathbf{G}_{j+1}-\mathbf{G}_j\|}{6}\Big(2\big\langle\nabla_d\phi(\mathbf{G}_j),\ \mathbf{n}_\mathbf{v}(\mathbf{G}_j)\big\rangle+2\big\langle\nabla_d\phi(\mathbf{G}_{j+1}),\ \mathbf{n}_\mathbf{v}(\mathbf{G}_{j+1})\big\rangle\\ &\quad+\big\langle\nabla_d\phi(\mathbf{G}_j),\ \mathbf{n}_\mathbf{v}(\mathbf{G}_{j+1})\big\rangle+\big\langle\nabla_d\phi(\mathbf{G}_{j+1}),\ \mathbf{n}_\mathbf{v}(\mathbf{G}_j)\big\rangle\Big).\end{aligned} \tag{9}$$

A second-order-accurate method is used to compute $\nabla_d\phi(\mathbf{G}_j)$. Using a Taylor expansion, we obtain

$$\begin{cases}\phi(\mathbf{v})-\phi(\mathbf{G}_j) = \big\langle\nabla_d\phi(\mathbf{G}_j),\mathbf{v}-\mathbf{G}_j\big\rangle+0.5\Delta_d\phi(\mathbf{G}_j)\|\mathbf{v}-\mathbf{G}_j\|^2+O(\|\mathbf{v}-\mathbf{G}_j\|^2),\\ \phi(\mathbf{v})_j-\phi(\mathbf{G}_j) = \big\langle\nabla_d\phi(\mathbf{G}_j),\mathbf{v}_j-\mathbf{G}_j\big\rangle+0.5\Delta_d\phi(\mathbf{G}_j)\|\mathbf{v}_j-\mathbf{G}_j\|^2+O(\|\mathbf{v}-\mathbf{G}_j\|^2),\\ \phi(\mathbf{v}_{j+1})-\phi(\mathbf{G}_j) = \big\langle\nabla_d\phi(\mathbf{G}_j),\mathbf{v}-\mathbf{G}_j\big\rangle+0.5\Delta_d\phi(\mathbf{G}_j)\|\mathbf{v}_{j+1}-\mathbf{G}_j\|^2+O(\|\mathbf{v}_{j+1}-\mathbf{G}_j\|^2).\end{cases} \tag{10}$$

We assume that $\nabla_d\phi(\mathbf{G}_j)$ has the following form

$$\nabla_d\phi(\mathbf{G}_j) = \alpha_j(\mathbf{v}_j-\mathbf{G}_j)+\beta_j(\mathbf{v}_{j+1}-\mathbf{G}_j), \tag{11}$$

where α_j and β_j are two constants. Then, substituting Eq. (11) into Eq. (10) yields

$$\begin{pmatrix} \alpha_j \\ \beta_j \\ 0.5\Delta_d\phi(\mathbf{G}_j) \end{pmatrix} = B_j^{-1} \begin{pmatrix} \phi(\mathbf{v}) - \phi(\mathbf{G}_j) \\ \phi(\mathbf{v}_j) - \phi(\mathbf{G}_j) \\ \phi(\mathbf{v}_{j+1}) - \phi(\mathbf{G}_j) \end{pmatrix} \tag{12}$$

$$B_j = \begin{pmatrix} \langle \mathbf{v}_j - \mathbf{G}_j, \mathbf{v} - \mathbf{G}_j \rangle & \langle \mathbf{v}_{j+1} - \mathbf{G}_j, \mathbf{v} - \mathbf{G}_j \rangle & \langle \mathbf{v} - \mathbf{G}_j, \mathbf{v} - \mathbf{G}_j \rangle \\ \langle \mathbf{v}_j - \mathbf{G}_j, \mathbf{v}_j - \mathbf{G}_j \rangle & \langle \mathbf{v}_{j+1} - \mathbf{G}_j, \mathbf{v}_j - \mathbf{G}_j \rangle & \langle \mathbf{v}_j - \mathbf{G}_j, \mathbf{v}_j - \mathbf{G}_j \rangle \\ \langle \mathbf{v}_j - \mathbf{G}_j, \mathbf{v}_{j+1} - \mathbf{G}_j \rangle & \langle \mathbf{v}_{j+1} - \mathbf{G}_j, \mathbf{v}_{j+1} - \mathbf{G}_j \rangle & \langle \mathbf{v}_{j+1} - \mathbf{G}_j, \mathbf{v}_{j+1} - \mathbf{G}_j \rangle \end{pmatrix}. \tag{13}$$

It is not difficult to prove that $\Delta_d\phi(\mathbf{G}_j)$ is zero and B_j^{-1} exists, which implies that $\nabla_d\phi(\mathbf{G}_j)$ has second-order accuracy in space. Therefore, combing Eqs. (9)–(13), we can compute the Laplace–Beltrami operator (9) with second order accuracy. The Laplace–Beltrami operator $\Delta_\tau\phi(\mathbf{v})$ at the vertex $\mathbf{v}$ is rewritten as

$$\Delta_\tau\phi(\mathbf{v}_i) = \mathcal{L}_i \cdot \begin{pmatrix} \phi(\mathbf{v}_1) \\ \phi(\mathbf{v}_2) \\ \vdots \\ \phi(\mathbf{v}_{N_v}) \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} \Delta_\tau\phi(\mathbf{v}_1) \\ \Delta_\tau\phi(\mathbf{v}_2) \\ \vdots \\ \Delta_\tau\phi(\mathbf{v}_{N_v}) \end{pmatrix} = \begin{pmatrix} \mathcal{L}_1 \\ \mathcal{L}_2 \\ \vdots \\ \mathcal{L}_{N_v} \end{pmatrix} \cdot \begin{pmatrix} \phi(\mathbf{v}_1) \\ \phi(\mathbf{v}_2) \\ \vdots \\ \phi(\mathbf{v}_{N_v}) \end{pmatrix},$$

where $\mathcal{L}_i$ is an $1 \times N_v$ matrix for $i = 1,\ 2,\ \ldots,\ N_v$ that can be obtained by Eqs. (9)–(13). Furthermore, the $N_v \times N_v$ matrix $\mathcal{L}$ can be defined as $\mathcal{L} = (\mathcal{L}_1,\ \mathcal{L}_2,\ \ldots,\ \mathcal{L}_{N_v})'$. For more detail, the reader is referred to [21].

2.2. Numerical scheme for the PFC model on the surface

Let ϕ_i^n be the numerical approximation of $\phi(\mathbf{v}_i,\ n\Delta t)$, where Δt is the time step. To obtain a high-order numerical solution, a stable backward differentiation scheme can be applied to Eqs. (1)–(3):

$$\frac{3\phi_i^{n+1} - 4\phi_i^n + \phi_i^{n-1}}{2\Delta t} = \Delta_\tau\mu_i^{n+1},\ \ i = 1, \ldots, N_s, \tag{14}$$

$$\mu_i^{n+1} = (\phi_i^{n+1})^3 + (1-\epsilon)\phi_i^{n+1} + \Delta_\tau\kappa_i^{n+1}, \tag{15}$$

$$\kappa_i^{n+1} = 2\tilde{\phi}_i^{n+1} + \Delta_\tau\phi_i^{n+1}. \tag{16}$$

Here, $\tilde{\phi}_i^{n+1} = 2\phi_i^n - \phi_i^{n-1}$. It is obvious that Eqs. (14)–(16) are second-order accurate in time and space. As $(\phi_i^{n+1})^3$ is nonlinear, the Newton-type iterative scheme in [12] can be applied, and thereby the nonlinear systems can be solved at each time step. To reduce computational cost, the nonlinear term $(\phi^{n+1})^3$ is linearized as

$$\begin{aligned} (\phi_i^{n+1})^3 &= (\tilde{\phi}_i^{n+1})^3 + 3(\tilde{\phi}_i^{n+1})^2(\phi_i^{n+1} - \tilde{\phi}_i^{n+1}) + O((\phi_i^{n+1} - \tilde{\phi}_i^{n+1})^2) \\ &= 3(\tilde{\phi}_i^{n+1})^2\phi_i^{n+1} - 2(\tilde{\phi}_i^{n+1})^3 + O((\phi_{tt_i}{}^{n+1}\Delta t)^2), \end{aligned} \tag{17}$$

thereby retaining the second-order accuracy in time. Therefore, keeping the second-order accuracy in time and space, we can rewrite Eqs. (14)–(16) as

$$\frac{3\phi_i^{n+1} - 4\phi_i^n + \phi_i^{n-1}}{2\Delta t} = \Delta_\tau\mu_i^{n+1},\ \ i = 1, \ldots, N_s, \tag{18}$$

$$\mu_i^{n+1} = 3(\tilde{\phi}_i^{n+1})^2\phi_i^{n+1} - 2(\tilde{\phi}_i^{n+1})^3 + (1-\epsilon)\phi_i^{n+1} + \Delta_\tau\kappa_i^{n+1}, \tag{19}$$

$$\kappa_i^{n+1} = 2\tilde{\phi}_i^{n+1} + \Delta_\tau\phi_i^{n+1}. \tag{20}$$

In Section 2.5, a detailed proof will be provided for the unconditional stability of the proposed form. In this study, we set $\phi_i^{-1} = \phi_i^0$. However, for later computations, the discrete system (18)–(20) is indeed second-order accurate. Let us rewrite

Eqs. (18)–(20) as

$$\begin{pmatrix} 3\mathcal{I}/2 & -\Delta t\mathcal{L} & \mathbf{0} \\ -3\mathcal{D}(\tilde{\phi}^{n+1}) - (1-\epsilon)\mathcal{I} & \mathcal{I} & -\mathcal{L} \\ -\mathcal{L} & \mathbf{0} & \mathcal{I} \end{pmatrix} \begin{pmatrix} \phi_1^{n+1} \\ \vdots \\ \phi_{N_v}^{n+1} \\ \mu_1^{n+1} \\ \vdots \\ \mu_{N_s}^{n+1} \\ \kappa_1^{n+1} \\ \vdots \\ \kappa_{N_v}^{n+1} \end{pmatrix} = \begin{pmatrix} 2\phi_1^n - \phi_1^{n-1}/2 \\ \vdots \\ 2\phi_{N_v}^n - \phi_{N_v}^{n-1}/2 \\ -2(\tilde{\phi}^{n+1})_1^3 \\ \vdots \\ -2(\tilde{\phi}^{n+1})_{N_s}^3 \\ 2\tilde{\phi}_1^{n+1} \\ \vdots \\ 2\tilde{\phi}_{N_v}^{n+1} \end{pmatrix} \tag{21}$$

Here, $\mathcal{I}$ is the $N_v \times N_v$ identity matrix and $\mathbf{0}$ is the $N_v \times N_v$ zero matrix. $\mathcal{D}(\tilde{\phi}^{n+1})$ is a $N_v \times N_v$ diagonal matrix with diagonal element $(\tilde{\phi}^{n+1})^2$. To solve the discrete system (21), we use a biconjugate gradient stabilized method [28].

2.3. Total mass conservation property

It is now proved that the proposed numerical scheme satisfies the total mass conservation property, that is,

$$\sum_{\mathbf{v}_i \in S} \phi_i^{n+1} D(\mathbf{v}_i) = \sum_{\mathbf{v}_i \in S} \phi_i^n D(\mathbf{v}_i). \tag{22}$$

By multiplying Eq. (18) by $D(\mathbf{v})$ and summing by parts, we can obtain that

$$\begin{aligned} &\sum_{\mathbf{v}_i \in S} 3\phi_i^{n+1} D(\mathbf{v}_i) - 4\sum_{\mathbf{v}_i \in S} \phi_i^n D(\mathbf{v}_i) + \sum_{\mathbf{v}_i \in S} \phi_i^{n-1} D(\mathbf{v}_i) \\ &= 2\Delta t \sum_{\mathbf{v}_i \in S} \Delta_\tau \mu_i^{n+\frac{1}{2}} D(\mathbf{v}_i) \\ &= \Delta t \sum_{\mathbf{v}_i \in S} \left(\sum_{j=0}^{p-1} \frac{\|\mathbf{G}_{j+1} - \mathbf{G}_j\|}{3} \Big(2\langle \nabla_d \mu(\mathbf{G}_j),\ \mathbf{n}(\mathbf{G}_j) \rangle + 2\langle \nabla_d \mu(\mathbf{G}_{j+1}),\ \mathbf{n}(\mathbf{G}_{j+1}) \rangle \right. \right. \\ &\quad \left. \left. + \langle \nabla_d \mu(\mathbf{G}_j),\ \mathbf{n}(\mathbf{G}_{j+1}) \rangle + \langle \nabla_d \mu(\mathbf{G}_{j+1}),\ \mathbf{n}(\mathbf{G}_j) \rangle \Big) \right) \end{aligned} \tag{23}$$

Let $\overline{\mathbf{G}_j\mathbf{G}_{j+1}}$ denote the edge of $\mathbf{G}_j\mathbf{G}_{j+1}$. It is obvious that $\overline{\mathbf{G}_j\mathbf{G}_{j+1}}$ belongs to the dual neighbor triangular mesh of the vertices $\mathbf{s}$ and $\mathbf{v}_{j+1}$ (see Fig. 1). It should be noted that in Eq. (23), the sum in the last term is for the vertex $\mathbf{v}$. Thus, this sum can be rewritten as

$$\begin{aligned} &\sum_{\mathbf{v}_i \in S} 3\phi_i^{n+1} D(\mathbf{v}_i) - 4\sum_{\mathbf{v}_i \in S} \phi_i^n D(\mathbf{v}_i) + \sum_{\mathbf{v}_i \in S} \phi_i^{n-1} D(\mathbf{v}_i) \\ &= \Delta t \sum_{\overline{\mathbf{G}_j\mathbf{G}_{j+1}} \in \tilde{S}} \left(\frac{\|\mathbf{G}_{j+1} - \mathbf{G}_j\|}{3} \Big(2\langle \nabla_d \mu(\mathbf{G}_j),\ \mathbf{n}_{\mathbf{v}}(\mathbf{G}_j) \rangle + 2\langle \nabla_d \mu(\mathbf{G}_{j+1}),\ \mathbf{n}_{\mathbf{v}}(\mathbf{G}_{j+1}) \rangle \right. \right. \\ &\quad + \langle \nabla_d \mu(\mathbf{G}_j),\ \mathbf{n}_{\mathbf{v}}(\mathbf{G}_{j+1}) \rangle + \langle \nabla_d \mu(\mathbf{G}_{j+1}),\ \mathbf{n}_{\mathbf{v}}(\mathbf{G}_j) \rangle \\ &\quad + 2\langle \nabla_d \phi(\mathbf{G})_j,\ \mathbf{n}_{\mathbf{v}_{j+1}}(\mathbf{G}_j) \rangle + 2\langle \nabla_d \mu(\mathbf{G}_{j+1}),\ \mathbf{n}_{\mathbf{v}_{j+1}}(\mathbf{G}_{j+1}) \rangle \\ &\quad \left. \left. + \langle \nabla_d \mu(\mathbf{G}_j),\ \mathbf{n}_{\mathbf{v}_{j+1}}(\mathbf{G}_{j+1}) \rangle + \langle \nabla_d \mu(\mathbf{G}_{j+1}),\ \mathbf{n}_{\mathbf{v}_{j+1}}(\mathbf{G}_j) \rangle \Big) \right). \end{aligned} \tag{24}$$

As all vertices are labeled counterclockwise, Eq (8) implies that

$$\begin{aligned} \mathbf{n}_{\mathbf{v}_{j+1}}(\mathbf{G}_j) &= \frac{(\mathbf{G}_j - \mathbf{G}_{j+1}) \times \mathbf{N}_j}{\|(\mathbf{G}_j - \mathbf{G}_{j+1}) \times \mathbf{N}_j\|} = -\mathbf{n}_{\mathbf{v}}(\mathbf{G}_j) \quad \text{and} \\ \mathbf{n}_{\mathbf{v}_{j+1}}(\mathbf{G}_{j+1}) &= \frac{(\mathbf{G}_j - \mathbf{G}_{j+1}) \times \mathbf{N}_{j+1}}{\|(\mathbf{G}_j - \mathbf{G}_{j+1}) \times \mathbf{N}_{j+1}\|} = -\mathbf{n}_{\mathbf{v}}(\mathbf{G}_{j+1}). \end{aligned} \tag{25}$$

Substituting Eq. (25) into Eq. (24) yields

$$3\sum_{\mathbf{v}_i \in S} \phi_i^{n+1} D(\mathbf{v}_i) - 3\sum_{\mathbf{v}_i \in S} \phi_i^n D(\mathbf{v}_i) = \sum_{\mathbf{v}_i \in S} \phi_i^n D(\mathbf{v}_i) - \sum_{\mathbf{v}_i \in S} \phi_i^{n-1} D(\mathbf{v}_i). \tag{26}$$

As $\phi^0 = \phi^{-1}$, we have

$$3\sum_{\mathbf{v}_i\in S}\phi_i^1 D(\mathbf{v}_i) - 3\sum_{\mathbf{v}_i\in S}\phi_i^0 D(\mathbf{v}_i) = \sum_{\mathbf{v}_i\in S}\phi_i^0 D(\mathbf{v}_i) - \sum_{\mathbf{v}_i\in S}\phi_i^{-1} D(\mathbf{v}_i) = 0,$$

which implies

$$\sum_{\mathbf{v}_i\in S}\phi_i^2 D(\mathbf{v}_i) = \sum_{\mathbf{v}_i\in S}\phi_i^1 D(\mathbf{v}_i). \tag{27}$$

Therefore, we have the chain of equalities

$$\sum_{\mathbf{v}_i\in S}\phi_i^{n+1} D(\mathbf{v}_i) = \sum_{\mathbf{v}_i\in S}\phi_i^{n} D(\mathbf{v}_i) = \cdots = \sum_{\mathbf{v}_i\in S}\phi_i^{1} D(\mathbf{v}_i) = \sum_{\mathbf{v}_i\in S}\phi_i^{0} D(\mathbf{v}_i), \tag{28}$$

which suggests that the proposed numerical scheme conserves the total mass.

2.4. Unique solvability

Eqs. (18)–(20) are uniquely solvable. The discrete inner product and centroid of the triangle face are defined by

$$(\phi,\psi)_d = \sum_{\mathbf{v}_i\in S}\phi_i\psi_i D(\mathbf{v}_i) \quad \text{and} \quad (\nabla_d\phi,\nabla_d\psi)_d = \sum_{\mathbf{v}_i\in S}\left[D(\mathbf{v}_i)\sum_{j=0}^{n-1}\nabla_d\phi(\mathbf{G}_j)\cdot\nabla_d\psi(\mathbf{G}_j)\right].$$

The discrete norm is defined as

$$\|\phi\|_d^2 = (\phi,\phi)_d, \quad \|\nabla_d\phi\|_d^2 = (\nabla_d\phi,\nabla_d\phi)_d, \quad \text{and} \quad \|\phi\|_{-1,d} = \sqrt{\left(\phi,(-\Delta_\tau)^{-1}\phi\right)_d}.$$

By taking the L_2 inner product of Eqs. (18)–(20) with $(-\Delta_\tau)^{-1}\psi$, we obtain

$$\begin{aligned}
0 &= \left(\frac{3\phi^{n+1}-4\phi^n+\phi^{n-1}}{2\Delta t},(-\Delta_\tau)^{-1}\psi\right)_d + \left(3(\tilde{\phi}_i^{n+1})^2\phi_i^{n+1} - 2(\tilde{\phi}_i^{n+1})^3,\psi\right)_d \\
&\quad +(1-\epsilon)\left(\phi^{n+1},\psi\right)_d + \left(2\Delta_\tau\tilde{\phi}^{n+1} + \Delta_\tau^2\phi^{n+1},\psi\right)_d \\
&= \frac{3}{2\Delta t}\left(\phi^{n+1},(-\Delta_\tau)^{-1}\psi\right)_d + \left(3(\tilde{\phi}^{n+1})^2\phi^{n+1} + (1-\epsilon)\phi^{n+1} + \Delta_\tau^2\phi^{n+1},\psi\right)_d \\
&\quad +\frac{1}{2\Delta t}\left(-4\phi^n+\phi^{n-1},(-\Delta_\tau)^{-1}\psi\right)_d + \left(-2(\tilde{\phi}^{n+1})^3 + 2\Delta_\tau\tilde{\phi}^{n+1},\psi\right)_d.
\end{aligned} \tag{29}$$

Let

$$G(\phi) := \frac{3}{4\Delta t}\|\phi\|_{-1,d}^2 + \frac{3}{2}\|\tilde{\phi}^{n+1}\phi\|_d^2 + \frac{1-\epsilon}{2}\|\phi\|_d^2 + \frac{1}{2}\|\Delta_\tau\phi\|_d^2 + F_1(\phi^n,\phi^{n-1},\phi) - \left(\delta_\phi F_2(\tilde{\phi}^{n+1}),\phi\right)_d.$$

Here, $F_1(\phi^n,\phi^{n-1},\phi) = ((-4\phi^n+\phi^{n-1})/(2\Delta t),(-\Delta_\tau)^{-1}\phi)_d - (2(\tilde{\phi}^{n+1})^3,\phi)_d$ is a bounded linear functional, $F_2(\phi) = \|\nabla_d\phi\|_d^2$ is strictly convex, and δ_ϕ is the discrete variation with respect to ϕ. As $G(\phi)$ can be written as the difference of two purely convex functionals, it has a unique minimizer by the convex splitting algorithm. The scheme (18)–(20) is equivalent to $\delta_\phi G(\phi^{n+1}) = 0$. Therefore, its solution exists and is unique.

2.5. Unconditional energy stability

Theorem. *If ϕ^{n+1}, ϕ^n, and ϕ^{n-1} are the solutions of Eqs. (18)–(20), then the scheme (18)–(20) is unconditionally stable and*

$$\mathcal{E}^d(\phi^{n+1},\phi^n) \le \mathcal{E}^d(\phi^n,\phi^{n-1}). \tag{30}$$

Here,

$$\begin{aligned}
\mathcal{E}^d(\phi^{n+1},\phi^n) &= 2\|\tilde{\phi}^{n+1}(\phi^{n+1}-\phi^n)\|_d^2 + \frac{1}{2}\left(\|\tilde{\phi}^{n+1}\phi^{n+1}\|_d^2 + \|\tilde{\phi}^{n+1}(2\phi^{n+1}-\phi^n)\|_d^2\right) \\
&\quad +\frac{1-\epsilon}{2}\left(\|\phi^{n+1}\|_d^2 + \|2\phi^{n+1}-\phi^n\|_d^2\right) + \frac{1}{2}\left(\|\Delta_\tau\phi^{n+1}\|_d^2 + \|2\Delta_\tau\phi^{n+1}-\Delta_\tau\phi^n\|_d^2\right) \\
&\quad -\left(\|\nabla_d\phi^{n+1}\|_d^2 + \|2\nabla_d\phi^{n+1}-\nabla_d\phi^n\|_d^2 - 2\|\nabla_d\phi^{n+1}-\nabla_d\phi^n\|_d^2\right).
\end{aligned} \tag{31}$$

and $\mathcal{E}^d(\phi^n,\phi^{n-1})$ is defined likewise.

Proof. By taking the L_2 inner product of Eq. (18) and $-2\Delta t\mu^{n+1}$, we obtain

$$-(3\phi^{n+1}-4\phi^{n}+\phi^{n-1},\mu^{n+1})_d=2\Delta t\|\nabla_d\mu^{n+1}\|_d^2. \tag{32}$$

By multiplying Eq. (19) by $3\phi^{n+1}-4\phi^{n}+\phi^{n-1}$, we have

$$\begin{aligned}&\left(3\phi^{n+1}-4\phi^{n}+\phi^{n-1},\mu^{n+1}\right)_d\\&=\left(2(\tilde{\phi}^{n+1})^2(\phi^{n+1}-\tilde{\phi}^{n+1})+(\tilde{\phi}^{n+1})^2\phi^{n+1}+(1-\epsilon)\phi^{n+1}+\Delta_\tau\kappa^{n+1},\ 3\phi^{n+1}-4\phi^{n}+\phi^{n-1}\right)_d.\end{aligned} \tag{33}$$

The terms in the resultant Eq. (33) are computed as follows:

$$\begin{aligned}&\left(2(\tilde{\phi}^{n+1})^2(\phi^{n+1}-\tilde{\phi}^{n+1}),\ 3\phi^{n+1}-4\phi^{n}+\phi^{n-1}\right)_d\\&=\left(2(\tilde{\phi}^{n+1})^2(\phi^{n+1}-2\phi^{n}+\phi^{n-1}),\ 2(\phi^{n+1}-\phi^{n})+(\phi^{n+1}-2\phi^{n}+\phi^{n-1})\right)_d\\&=2\|\tilde{\phi}^{n+1}(\phi^{n+1}-\phi^{n})\|_d^2-2\|\tilde{\phi}^{n+1}(\phi^{n}-\phi^{n-1})\|_d^2+4\|\tilde{\phi}^{n+1}(\phi^{n+1}-2\phi^{n}+\phi^{n-1})\|_d^2\end{aligned} \tag{34}$$

$$\begin{aligned}\left((\tilde{\phi}^{n+1})^2\phi^{n+1},\ 3\phi^{n+1}-4\phi^{n}+\phi^{n-1}\right)_d=\frac{1}{2}\Big(&\|\tilde{\phi}^{n+1}\phi^{n+1}\|_d^2-\|\tilde{\phi}^{n+1}\phi^{n}\|_d^2+\|\tilde{\phi}^{n+1}(2\phi^{n+1}-\phi^{n})\|_d^2\\&-\|\tilde{\phi}^{n+1}(2\phi^{n}-\phi^{n-1})\|_d^2+\|\tilde{\phi}^{n+1}(\phi^{n+1}-2\phi^{n}+\phi^{n-1})\|_d^2\Big)\end{aligned} \tag{35}$$

$$\begin{aligned}(1-\epsilon)\left(\phi^{n+1},\ 3\phi^{n+1}-4\phi^{n}+\phi^{n-1}\right)_d=\frac{1-\epsilon}{2}\Big(&\|\phi^{n+1}\|_d^2-\|\phi^{n}\|_d^2+\|2\phi^{n+1}-\phi^{n}\|_d^2-\|2\phi^{n}-\phi^{n-1}\|_d^2\\&+\|\phi^{n+1}-2\phi^{n}+\phi^{n-1}\|_d^2\Big)\end{aligned} \tag{36}$$

$$\left(\Delta_\tau\kappa^{n+1},\ 3\phi^{n+1}-4\phi^{n}+\phi^{n-1}\right)_d=\left(\kappa^{n+1},\ 3\Delta_\tau\phi^{n+1}-4\Delta_\tau\phi^{n}+\Delta_\tau\phi^{n-1}\right)_d. \tag{37}$$

By taking the L_2 inner product of Eq. (20) and $3\Delta_\tau\phi^{n+1}-4\Delta_\tau\phi^{n}+\Delta_\tau\phi^{n-1}$, we obtain

$$\left(\kappa^{n+1},3\Delta_\tau\phi^{n+1}-4\Delta_\tau\phi^{n}+\Delta_\tau\phi^{n-1}\right)_d=\left(2\tilde{\phi}^{n+1}+\Delta_\tau\phi^{n+1},\ 3\Delta_\tau\phi^{n+1}-4\Delta_\tau\phi^{n}+\Delta_\tau\phi^{n-1}\right)_d. \tag{38}$$

The terms in the resultant Eq. (38) are computed as follows:

$$\begin{aligned}&\left(2(2\phi^{n}-\phi^{n-1}),\ 3\Delta_\tau\phi^{n+1}-4\Delta_\tau\phi^{n}+\Delta_\tau\phi^{n-1}\right)_d\\&=-\left(\|\nabla_d\phi^{n+1}\|_d^2+\|2\nabla_d\phi^{n+1}-\nabla_d\phi^{n}\|_d^2-2\|\nabla_d\phi^{n+1}-\nabla_d\phi^{n}\|_d^2\right)\\&\quad+\left(\|\nabla_d\phi^{n}\|_d^2+\|2\nabla_d\phi^{n}-\nabla_d\phi^{n-1}\|_d^2-2\|\nabla_d\phi^{n}-\nabla_d\phi^{n-1}\|_d^2\right)\\&\quad+3\|\nabla_d\phi^{n}-2\nabla_d\phi^{n}+\nabla_d\phi^{n-1}\|_d^2\end{aligned} \tag{39}$$

$$\begin{aligned}&\left(\Delta_\tau\phi^{n+1},\ 3\Delta_\tau\phi^{n+1}-4\Delta_\tau\phi^{n}+\Delta_\tau\phi^{n-1}\right)_d\\&=\frac{1}{2}\Big(\|\Delta_\tau\phi^{n+1}\|_d^2-\|\Delta_\tau\phi^{n}\|_d^2+\|2\Delta_\tau\phi^{n+1}-\Delta_\tau\phi^{n}\|_d^2\\&\quad-\|2\Delta_\tau\phi^{n}-\Delta_\tau\phi^{n-1}\|_d^2+\|\Delta_\tau\phi^{n+1}-2\Delta_\tau\phi^{n}+\Delta_\tau\phi^{n-1}\|_d^2\Big).\end{aligned} \tag{40}$$

By combining Eqs. (32)–(40) and using the definition of energies (30) and (31), we obtain

$$\begin{aligned}\mathcal{E}^d(\phi^{n+1},\phi^{n})-\mathcal{E}^d(\phi^{n},\phi^{n-1})=&-2\Delta t\|\nabla_d\mu^{n+1}\|_d^2-4\|\tilde{\phi}^{n+1}(\phi^{n+1}-2\phi^{n}+\phi^{n-1})\|_d^2\\&-\frac{1}{2}\|\tilde{\phi}^{n+1}(\phi^{n+1}-2\phi^{n}+\phi^{n-1})\|_d^2-\frac{1-\epsilon}{2}\|\phi^{n+1}-2\phi^{n}+\phi^{n-1}\|_d^2\\&-3\|\nabla_d\phi^{n}-2\nabla_d\phi^{n}+\nabla_d\phi^{n-1}\|_d^2-\frac{1}{2}\|\Delta_\tau\phi^{n+1}-2\Delta_\tau\phi^{n}+\Delta_\tau\phi^{n-1}\|_d^2.\end{aligned} \tag{41}$$

which implies the desired result (30). The proof is complete. □

Table 1
Numerical errors and calculated convergence rates with various mesh grids. Here, $\Delta t = 0.05$ is used. The reference solution is obtained using $h = 0.05$.

h	2		1		0.5		0.25
l_2-error	1.976E−2		4.636E−3		1.231E−3		3.231E−4
Rate		2.09		1.91		1.93	

Table 2
Numerical errors and calculated convergence rates. Here $h = 0.125$ is fixed. The reference solution is obtained with $\Delta t = 0.0625$.

Δt	2		1		0.5		0.25
l_2-error	2.452E−3		5.616E−4		1.422E−4		3.701E−5
Rate		2.13		1.98		1.94	

3. Numerical results

Herein, numerical results are presented. Unless otherwise specified, we use $\epsilon = 0.25$.

3.1. Convergence test

The convergence rate predicted by the proposed scheme is verified. If the L_2 norm of the residual is less than a given tolerance (1E−12), then the iteration of the proposed scheme is terminated. The surface of a sphere with radius 30 is chosen as a test surface. The initial density field is taken as

$$\phi^0(\mathbf{v}) = 0.1 + 0.05\cos(2\pi x)\cos(2\pi y)\cos(2\pi z), \tag{42}$$

where $\mathbf{v} = (x, y, z)$. The system is evolved up to final time $T = 5$ with $\Delta t = 0.05$. The mesh grids are taken to be $h = 2,\ 1,$ 0.5, and 0.25. ϕ^{ref} is taken as a reference numerical solution, which is obtained with a fine space grid $h = 0.05$. For a grid $\mathbf{v}_i$ in the coarse surface mesh, three reference cells ($\mathbf{v}_p^{\text{ref}}$, $\mathbf{v}_q^{\text{ref}}$, and $\mathbf{v}_r^{\text{ref}}$) neighboring it can be obtained in the reference triangular surface mesh. By taking the linear interpolation operator, the weightings ζ_i, η_i, and θ_i are obtained, which satisfy $\mathbf{v}_i = \zeta_i\mathbf{v}_p^{\text{ref}} + \eta_i\mathbf{v}_q^{\text{ref}} + \theta_i\mathbf{v}_r^{\text{ref}}$. With the obtained weightings, the error of a grid can be defined as the L_2-norm of the difference between that grid and the mean of the neighboring reference solutions, that is, $e_{hi} := \phi_i - (\zeta_i\phi_p^{\text{ref}} + \eta_i\phi_q^{\text{ref}} + \theta_i\phi_r^{\text{ref}})$. $\log_2(\|e_h\|_2/\|e_{\frac{h}{2}}\|_2)$ is the convergence rate. Table 1 lists the numerical errors and rates of convergence. The results suggest that the scheme is second-order accurate.

To study the accuracy of the proposed scheme with respect to time, $\Delta t = 2,\ 1,$ 0.5, and 0.25 are taken. Here $h = 0.125$ is fixed. The numerical reference solution ϕ^{ref} is obtained with a very fine time step $\Delta t = 0.0625$. $\log_2(\|e_{\Delta t}\|_2/\|e_{\frac{\Delta t}{2}}\|_2)$ is the convergence rate. All numerical solutions are computed up to time $T = 200$. Table 2 lists the errors and the second-order convergence rates.

3.2. Stability of the proposed scheme

It is difficult to obtain a stable numerical scheme for the PFC equation because it involves a nonlinear term and sixth-order spatial derivatives. To demonstrate the stability of the proposed scheme (18)–(20), a numerical experiment is conducted using large time steps, namely, $\Delta t = 10$ and 100. The initial condition and parameters are the same. Here, $h = 0.5$ is used. All computations are run up to final time $T = 1000$. The computational solutions at $T = 1000$ with $\Delta t = 10$ and 100 are shown in Fig. 2. To confirm the accuracy of numerical solution, the reference solutions ϕ^{ref}, which are obtained with $\Delta t = 1$, are considered. Fig. 2(d) shows the temporal evolution of the energy and the mass of the density field ϕ. This plot shows that the mass is conserved and the energy is non-increasing. It should be noted that by observing the evolution of the energy, the evolution of the PFC equation has several time scales. Initially, it rapidly evolves and exhibits slow evolution at a later time. To save the computational cost, an adaptive time step technique [12,29] can be used to compute the PFC equation.

3.3. Comparison study on the dynamics between flat and non-flat surfaces

This example presents a comparison of the PFC dynamics between flat and non-flat surfaces. For the flat surface, a square domain $(0, 100)\times(0, 100)$ is used. The mesh is composed of 128^2 elements. For the non-flat surface, a spherical surface with radius $100\sqrt{4\pi}$ is chosen, so that the areas of the two surfaces are equal. On the spherical surface, a uniform computational mesh composed of 16000 elements is defined. As initial condition, the following expression is used to define a crystal lattice: $\phi^0(\mathbf{v}) = 0.15 + 0.05\text{rand}(\mathbf{v})$, where rand($\mathbf{v}$) is a random number between -1 and 1. The time step is $\Delta t = 1$. We use periodic boundary conditions for the flat surface. Fig. 3 shows the time evolution of the energy for the PFC on the flat and

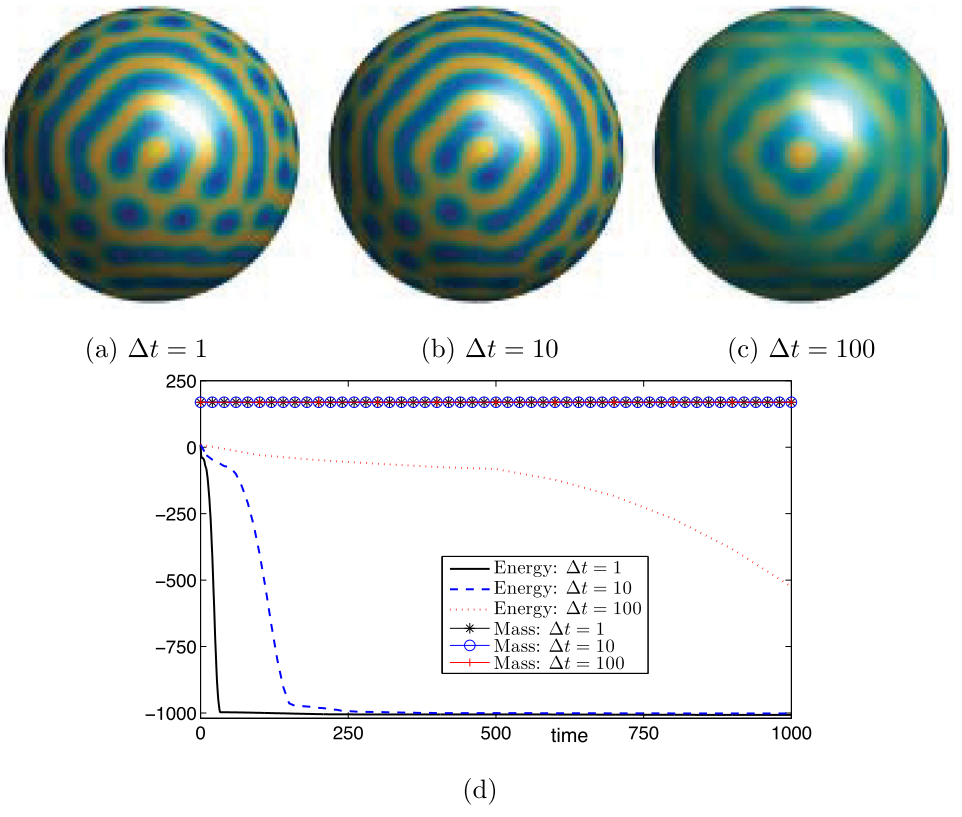

Fig. 2. (a–c) Density field ϕ with different time steps at time $t = 1000$. The used time steps are listed below each figure. (d) Evolutions of energy and mass with three different time steps.

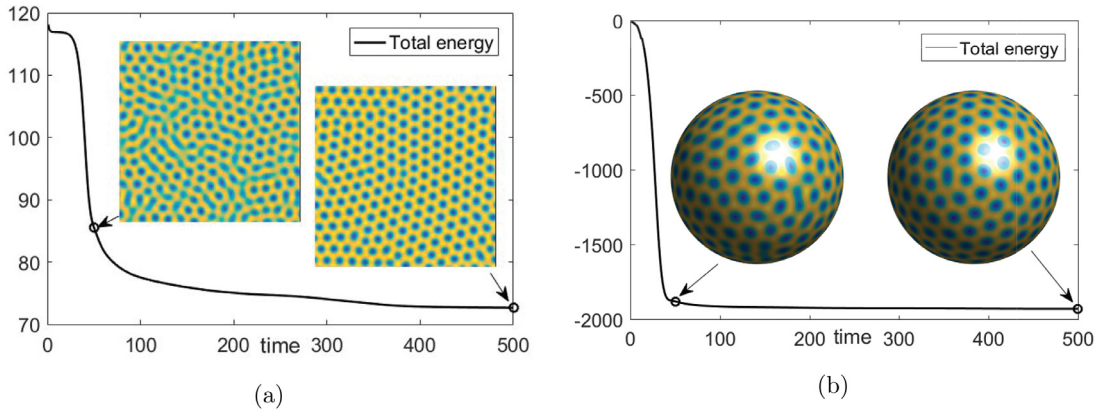

Fig. 3. Time evolution of PFC growth on flat surface (a) and non-flat surface (b).

the non-flat surfaces. It is observed that the energy decreases at all times. Furthermore, on the non-flat surface, the energy decreases faster and the radius of the hexagonal crystal is larger than those obtained on the flat surface.

3.4. Comparison with previous method

In [20], Lee and Kim presented a finite difference method for the PFC equation on curved surfaces. First, they employed a narrow band and extended the PFC equation on the surface to the 3D narrow band domain. Subsequently, they used the standard discrete Laplacian operator. Lee and Kim performed a simulation on a spherical surface with a radius 64 and initial data $\phi^0(\mathbf{v}) = 0.15 + 0.05\text{rand}(\mathbf{v})$. A time step $\Delta t = 1$, a grid size $h = 1$, and a narrow band domain with a thickness

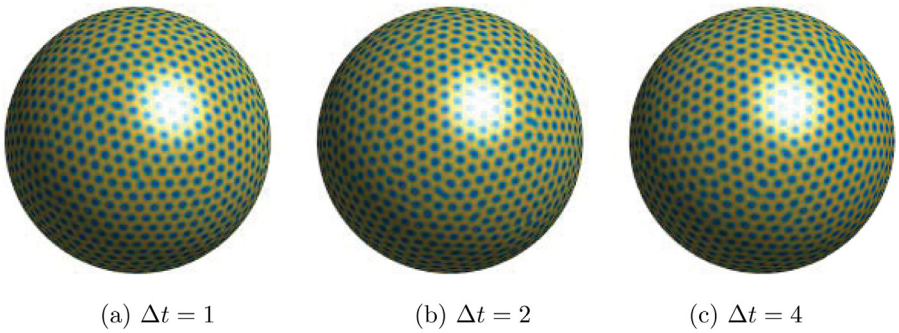

Fig. 4. Phase field ϕ with different time steps at time $t = 5000$. The time steps used are listed below each figure.

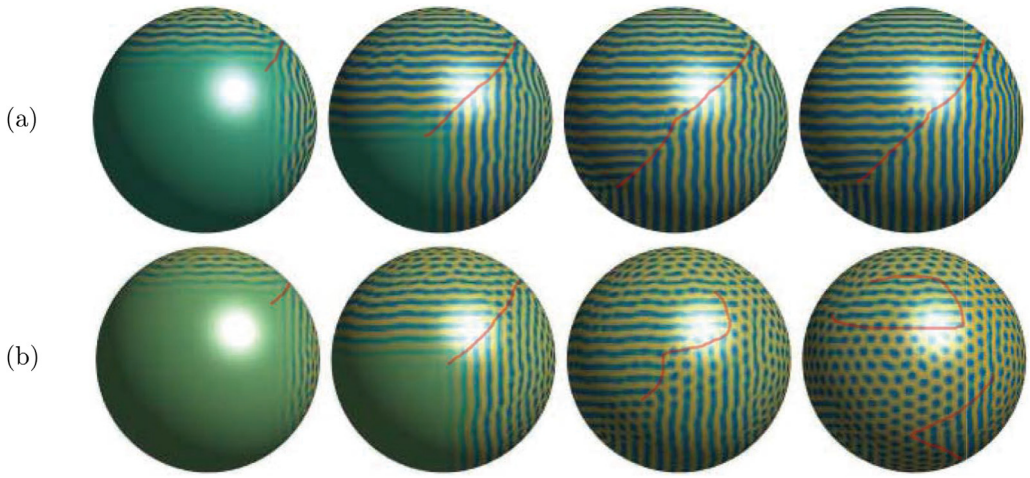

Fig. 5. (a) and (b) are the density field ϕ with $\phi_{ave} = 0.05$ and $\phi_{ave} = 0.15$, respectively. Two random perturbations are placed on the pole of the spherical surface as nucleation seeds. From left to right, the time is $t = 50$, $t = 100$, $t = 200$, and $t = 500$. It should be noted that the red solid lines are guides for finding the grain boundaries.

of $2.2\sqrt{3}h$ were chosen. To compare the present results with those from [20], the same initial condition and equivalent edge length are used.

Fig. 4(a)–(c) shows the phase field ϕ at time $t = 5000$ with $\Delta t = 1$, 2, and 4, respectively. As can be observed, the numerical results are similar to those in ([20], Fig. 5). The agreement between the results with different time steps suggests that owing to its second-order accuracy, the proposed scheme can use a slightly lager time step to obtain equivalent numerical results. The grid size used here is 64008, which is approximately one-quarter of the size of the grid used in [20].

3.5. Dynamics of polycrystals and grain boundaries on a sphere surface

The proposed scheme is now used to compute the growth of a polycrystal and the grain boundary dynamics in a supercooled liquid on a surface. This experiment demonstrates the applicability of the proposed method to a physical problem. A system on a spherical surface with radius 50 is simulated. As nucleation seeds, two random perturbations are placed on the pole of the spherical surface with the following expression:

$$\phi^0(\mathbf{v}) = \begin{cases} \phi_{ave} + 0.1\,\mathrm{rand}(\mathbf{v}), & \text{if } z > 45, \\ \phi_{ave} + 0.3\,\mathrm{rand}(\mathbf{v}), & \text{if } x > 45, \\ \phi_{ave}, & \text{otherwise.} \end{cases}$$

Here, ϕ_{ave} is the average of the density field ϕ and $\Delta t = 1$. Fig. 5(a) and (b) shows snapshots of the crystal microstructure with $\phi_{ave} = 0.05$ and $\phi_{ave} = 0.15$, respectively. The computational time is shown below each figure. It can be seen that as time evolves the crystallites grow and form grain boundaries. As two different initial configurations are considered, the two crystallites evolve with a different orientation and striped pattern. Depending on the average of density field ϕ, there are different patterns, e.g., striped (Fig. 5(a)) and hexagonal (Fig. 5(b)). It should be noted that the red solid lines in Fig. 5 are guides for finding the grain boundaries. The evolution of the total energy with different average of the density field ϕ is shown in Fig. 6. It can be seen that these energies are non-increasing.

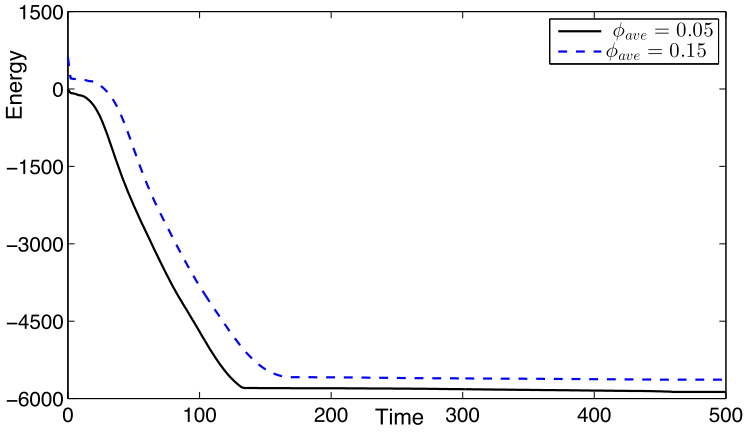

Fig. 6. Evolution of energy with different ϕ_{ave}.

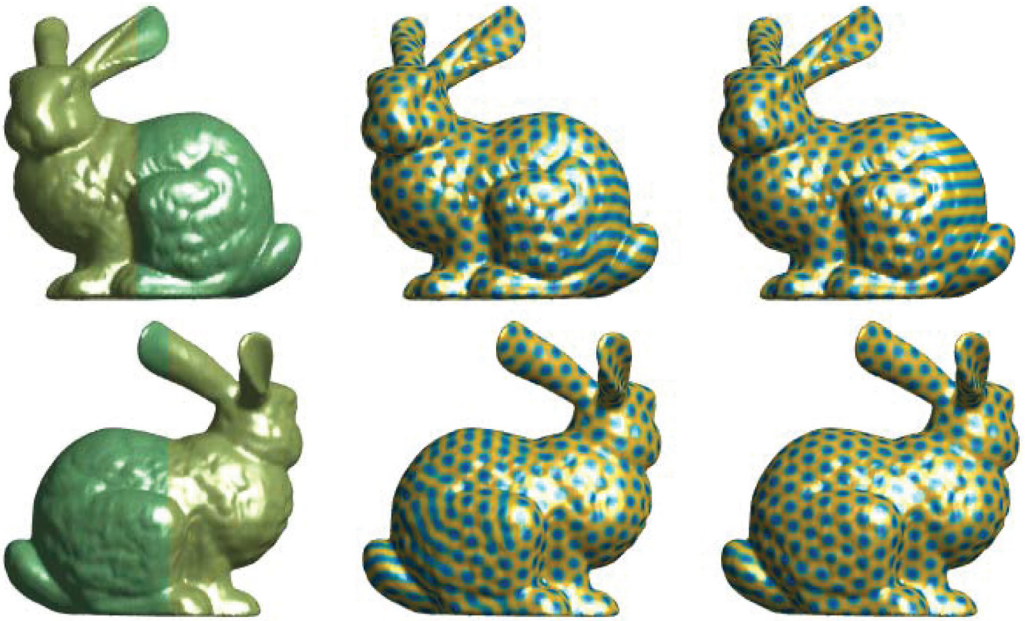

Fig. 7. PFC growth on a bunny surface. The first and second row show the results from different view points. The computational time is given below each figure.

3.6. PFC growth on a bunny surface

In this section, the PFC equation will be solved on a bunny surface, which is set in a box $(0, 100) \times (0, 100) \times (0, 80)$ with the following initial condition:

$$\phi^0(\mathbf{v}) = \begin{cases} 0.05 + 0.1\,\mathrm{rand}(\mathbf{v}), & \text{if } x < 50, \\ 0.15 + 0.1\,\mathrm{rand}(\mathbf{v}), & \text{otherwise.} \end{cases}$$

The simulation is run up to time $T = 1000$ with $\Delta t = 1$. Fig. 7(a)–(c) shows the results of crystal growth on the bunny surface at time $t = 0,\ 100,$ and 1000, respectively. The first and second row show the results from different view points. A mixture of lamellar and hexagonal patterns is observed. These results confirm that the proposed algorithm performs well on complex surfaces.

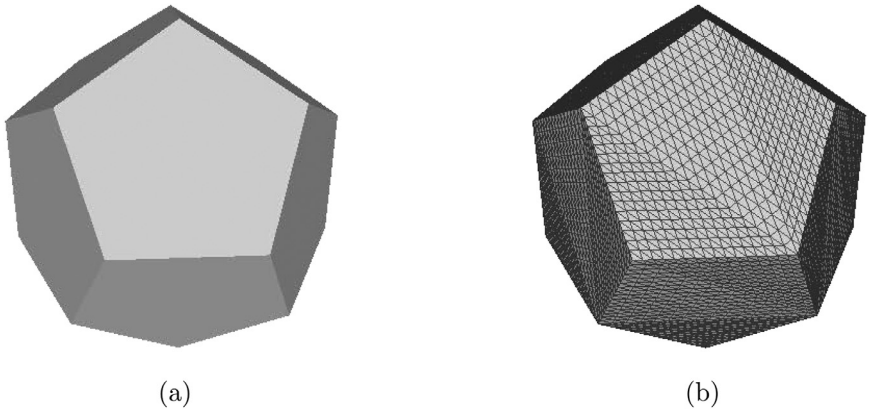

Fig. 8. (a) Dodecahedron. (b) Triangular mesh on the Dodecahedron surface.

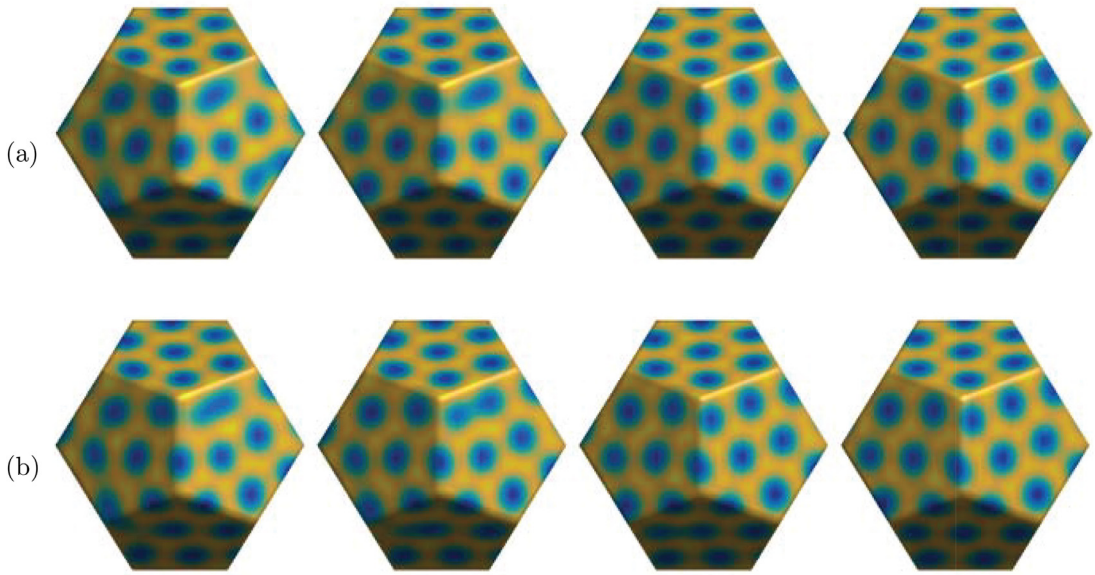

Fig. 9. Density field ϕ on a sharp surface. (a) Dodecahedron surface with 4610 vertices. (b) Dodecahedron surface with 18, 442 vertices. From left to right, the time is $t = 100$, $t = 200$, $t = 400$, and $t = 1000$, respectively.

3.7. PFC growth on a sharp surface

In this section, the evolution of the density field will be considered on a Dodecahedron surface having sharp feature (see Fig. 8). The Dodecahedron is located in the box $(-15, 15)^3$. The initial configuration of ϕ is a random perturbation from $\phi_{ave} = 0.05$. The simulation was performed up to time $t = 1000$ with $\Delta t = 1$. Fig. 9(a) and (b) shows snapshots of ϕ with 4610 vertices and 18, 442 vertices, respectively. From left to right, the time is $t = 100$, 200, 400, and 1000. The convergence results obtained by using coarse and finer mesh confirm that the proposed method performs well on sharp surface.

3.8. Dynamics of crystal growth on a non-uniform surface mesh

In this test, it will be demonstrated that the proposed method can be used with an adaptive mesh. Fig. 10 shows the non-uniform surface mesh structure. The mean curvature of the surface provides an indicator for determining a region to be refined. The domain near the left and right ends is refined with a finer mesh. In the middle domain, a coarse mesh is used. The parameter ϵ is related to temperature [30]. The initial condition is given as $\phi^0(\mathbf{v}) = 0.1 + 0.1\text{rand}(\mathbf{v})$. The density field at time $t = 1000$ with $\epsilon = 0.1$ and 1 are shown in the first row and second row of Fig. 11, respectively. Here, $\Delta t = 1$ is used. Depending on the value of the parameter ϵ, different patterns are obtained. That is, a smaller ϵ results in a hexagonal pattern. As ϵ increases, the pattern becomes striped. The third and fourth rows in Fig. 11 show the density field ϕ with $\epsilon = 0.1$ and 1, respectively, on the uniform mesh at time $T = 1000$. Here, the nodes used in the uniform mesh are 2.42 as many as in the non-uniform mesh. The agreement between the results using the non-uniform and uniform mesh is obvious.

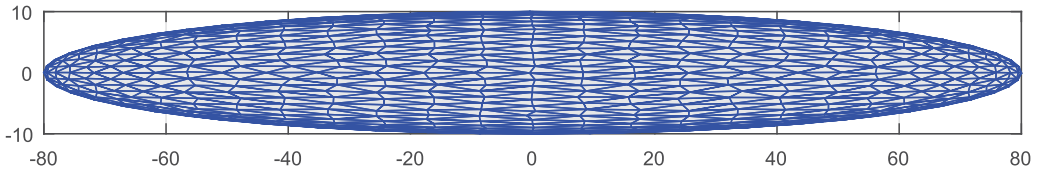

Fig. 10. Non-uniform surface mesh structure. For better visualization, the surface mesh is displayed more sparsely than in reality.

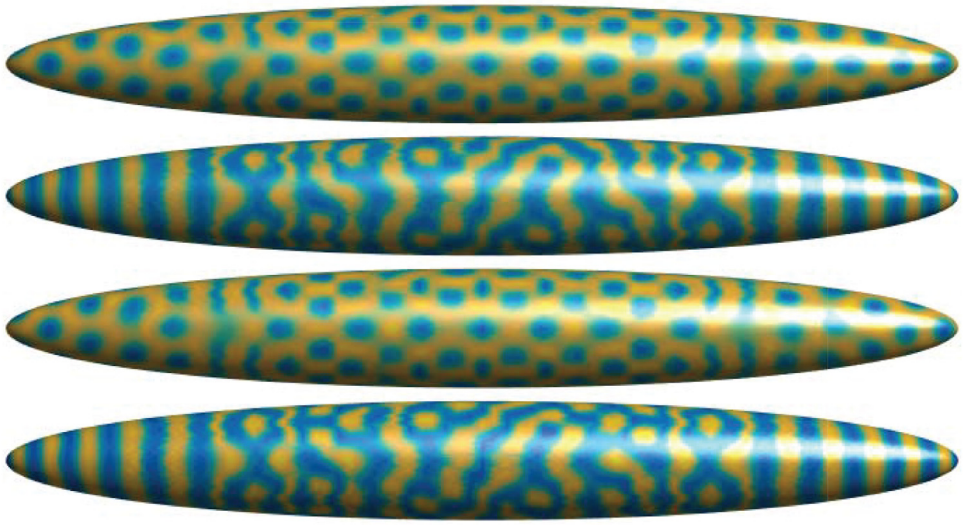

Fig. 11. Density field ϕ at time $T = 1000$. The first two rows show the solutions on the non-uniform grid with $\epsilon = 1$ and $\epsilon = 0.1$, respectively. The second two rows show the solutions on the uniform grid with $\epsilon = 1$ and $\epsilon = 0.1$, respectively.

4. Conclusions

An efficient direct discretization method was proposed for solving the PFC equation on curved surfaces. The proposed method consists of a backward differentiation and a linearly stabilized splitting scheme. It is second-order accurate in space and time. The proposed scheme is easy to implement. It is proved that the proposed scheme is unconditionally stable. Numerical experiments such as time and space convergence, stability of the proposed scheme, dynamics of polycrystals and grain boundaries on a sphere surface, dynamics of crystal growth on a non-uniform surface mesh, and PFC growth on a bunny surface were performed. The computational tests confirmed the efficiency of the proposed method.

Acknowledgment

Y.B. Li is supported by National Natural Science Foundation of China (Nos. 11601416, 11631012, and 11771348). The corresponding author (J.S. Kim) is supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1A1B03933243). The authors thank the reviewers for their constructive and helpful comments on the revision of this article.

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] 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.
[6] 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.
[7] 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.
[8] H. Gomez, X. Nogueira, An unconditionally energy-stable method for the phase field crystal equation, Comput. Methods Appl. Mech. Eng. 249 (2012) 52–61.
[9] 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.
[10] K. Glasner, S. Orizaga, Improving the accuracy of convexity splitting methods for gradient flow equations, J. Comput. Phys. 315 (2016) 52–64.

[11] 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. Eng. 298 (2016) 453–484.
[12] Y. Li, J. Kim, An efficient and stable compact fourth-order finite difference scheme for the phase field crystal equation, Comput. Methods Appl. Mech. Eng. 319 (2017) 194–216.
[13] X. Yang, D. Han, Linearly first-and second-order, unconditionally energy stable schemes for the phase field crystal model, J. Comput. Phys. 330 (2017) 1116–1134.
[14] T. Witkowski, R. Backofena, A. Voigt, The influence of membrane bound proteins on phase separation and coarsening in cell membranes, Phys. Chem. Chem. Phys. 14 (2012) 14509–14515.
[15] C.M. Elliott, T. Ranner, Evolving surface finite element method for the Cahn–Hilliard equation, Numer. Math. 129 (2015) 483–534.
[16] M.H. Gfrerer, M. Schanz, A high-order FEM with exact geometry description for the Laplacian on implicitly defined surfaces, Int. J. Numer. Methods Eng. 16 (2018) 1–16.
[17] G. Dziuk, C.M. Elliott, An Eulerian approach to transport and diffusion on evolving implicit surfaces, Comput. Visual. Sci. 13 (2010) 17–28.
[18] S. Leung, J. Lowengrub, H.K. Zhao, A grid based particle method for solving partial differential equations on evolving surfaces and modeling high order geometrical motion, J. Comput. Phys. 230 (2011) 2540–2561.
[19] D. Jeong, J. Kim, Microphase separation patterns in Diblock copolymers on curved surfaces using a nonlocal Cahn–Hilliard equation, Eur. Phys. J. E 38 (2015) 1–7.
[20] H.G. Lee, J. Kim, A simple and efficient finite difference method for the phase-field crystal equation on curved surfaces, Comput. Methods Appl. Mech. Eng. 307 (2016) 32–43.
[21] Y. Li, J. Kim, N. Wang, An unconditionally energy-stable second-order time-accurate scheme for the Cahn–Hilliard equation on surfaces, Commun. Nonlinear. Sci. Numer. Simul. 53 (2017) 213–227.
[22] Y. Li, X. Qi, J. Kim, Direct discretization method for the Cahn–Hilliard equation on an evolving surface, J. Sci. Comput. 77 (2) (2018) 1147–1163.
[23] F. Guillén-González, G. Tierra, Second order schemes and time-step adaptivity for Allen–Cahn and Cahn–Hilliard models, Comput. Math. Appl. 68 (8) (2014) 821–846.
[24] Y. Li, S. Guo, Triply periodic minimal surface using a modified Allen–Cahn equation, Appl. Math. Comput. 295 (2017) 84–94.
[25] Y. Li, H.G. Lee, B. Xia, J. Kim, A compact fourth-order finite difference scheme for the three-dimensional Cahn–Hilliard equation, Comput. Phys. Commun. 200 (2016) 108–116.
[26] D. Li, Z. Qiao, On second order semi-implicit fourier spectral methods for 2D Cahn–Hilliard equations, J. Sci. Comput. 70 (2017) 301–341.
[27] Y. Yan, W. Chen, C. Wang, S.M. Wise, A second-order energy stable BDF numerical scheme for the Cahn–Hilliard equation, Commun. Comput. Phys. 23 (2) (2018) 572–602.
[28] H.A. van der Vorst, BI-CGSTAB: a fast and smoothly converging variant of BI-CG for the solution of nonsymmetriclinear systems, SIAM J. Sci. Stat. Comput. 13 (2) (1992) 631–644.
[29] Y. Li, Y. Choi, J. Kim, Computationally efficient adaptive time step method for the Cahn–Hilliard equation, Comput. Math. Appl. 73 (8) (2017) 1855–1864.
[30] 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.