Commun Nonlinear Sci Numer Simulat 53 (2017) 213–227

Contents lists available at ScienceDirect

Commun Nonlinear Sci Numer Simulat

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

Research paper

An unconditionally energy-stable second-order time-accurate scheme for the Cahn–Hilliard equation on surfaces

Yibao Li[a], Junseok Kim[b,*], Nan Wang[c]

[a] School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China
[b] Department of Mathematics, Korea University, Seoul 02841, Korea
[c] College of Science, Engineering University of CAPF, Xian 710086, China

ARTICLE INFO

Article history:
Received 10 October 2016
Revised 15 April 2017
Accepted 9 May 2017
Available online 10 May 2017

Keywords:
Cahn–Hilliard equation
Laplace–Beltrami operator
Triangular surface mesh
Unconditionally energy-stable
Mass conservation

ABSTRACT

In this paper, we propose an unconditionally energy-stable second-order time-accurate scheme for the Cahn–Hilliard equation on surfaces. The discretization is performed via a surface mesh consisting of piecewise triangles and its dual-surface polygonal tessellation. The proposed scheme, which combines a Crank–Nicolson-type scheme with a linearly stabilized splitting scheme, is second-order accurate in time. The discrete system is shown to be conservative and unconditionally energy-stable. The resulting system of discrete equations is simple to implement, and can be solved using a biconjugate gradient stabilized method. We demonstrate the performance of our proposed algorithm through several numerical experiments.

1. Introduction

In this paper, our goal is to develop a second-order accurate, unconditionally energy-stable scheme for the Cahn–Hilliard equation (CH) [1] defined on a general surface S (see Fig. 1):

$$\frac{\partial \phi(\mathbf{s},t)}{\partial t} = M\Delta_s \mu(\mathbf{s},t), \quad \mathbf{s}\in S, \ 0<t\le T, \tag{1}$$

$$\mu(\mathbf{s},t) = F'(\phi(\mathbf{s},t)) - \epsilon^2 \Delta_s \phi(\mathbf{s},t), \tag{2}$$

with the initial condition

$$\phi(\mathbf{s},0) = \phi_0(\mathbf{s}), \quad \mathbf{s}\in S, \tag{3}$$

where $\phi(\mathbf{s}, t)$ is the order parameter, which denotes the concentration of one component of a binary mixture. $\mu(\mathbf{s}, t)$ is chemical potential. $M > 0$ is mobility. Δ_s and ∇_s denote the tangential (surface) Laplacian and gradient operators, respectively. $F(\phi) = 0.25(\phi^2 - 1)^2$ is a free energy density function and ϵ is a positive constant related to interfacial thickness. For simplicity of presentation, we assume herein that surface S is closed. It is equivalent to consider the periodic boundary condition for both the order parameter and chemical potential. Note that our method can also be extended to cases involving Dirichlet and Neumann boundary conditions.

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

http://dx.doi.org/10.1016/j.cnsns.2017.05.006
1007-5704/

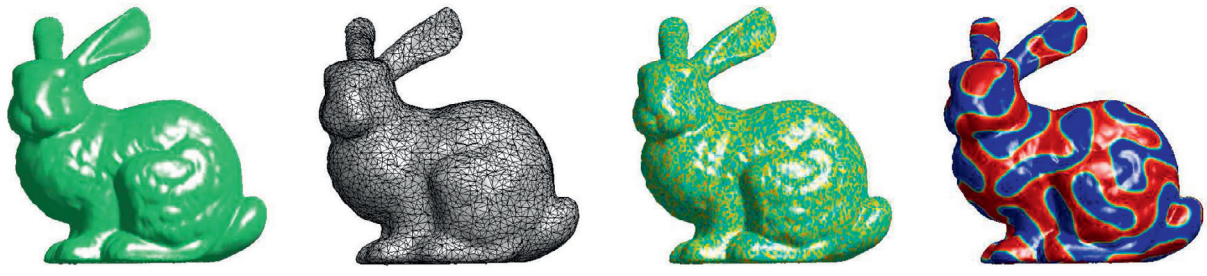

Fig. 1. Phase separation on the surface of a bunny shown as (a) bunny geometry, (b) the triangular mesh of the bunny surface, (c) the initial random phase mixture on the bunny surface, and (d) separated phases obtained by the Cahn–Hilliard equation. Note that for better visualization, the mesh points and triangles in (b) are displayed more sparsely than their actual densities.

Eqs. (1) and (2) are derived from a constrained gradient flow in the H^{-1} Hilbert space of the Helmholtz free energy functional:

$$\mathcal{E}(\phi) = \int_S \left(F(\phi) + \frac{\epsilon^2}{2} |\nabla_s \phi|^2 \right) d\mathbf{s}. \tag{4}$$

The properties of the CH equation are such that the total mass $\int_\Omega \phi \; d\mathbf{s}$ is conserved and the total energy $\mathcal{E}(\phi)$ decreases with time. That is,

$$\frac{d}{dt} \int_S \phi \; d\mathbf{s} = \int_S \phi_t \; d\mathbf{s} = M \int_S \Delta_s \mu \; d\mathbf{s} = M \int_{\partial S} \mathbf{n} \cdot \nabla_s \mu \; d\tau = 0, \tag{5}$$

$$\begin{aligned} \frac{d}{dt} \mathcal{E}(t) &= \int_S (F'(\phi)\phi_t + \epsilon^2 \nabla_s \phi \cdot \nabla_s \phi_t) d\mathbf{s} = \int_S \mu \phi_t d\mathbf{s} + \int_{\partial S} \phi_t \epsilon^2 \mathbf{n} \cdot \nabla_s \phi d\tau \\ &= M \int_S \mu \Delta_s \mu d\mathbf{s} = M \int_{\partial S} \mu \mathbf{n} \cdot \nabla_s \mu d\tau - M \int_S \nabla_s \mu \cdot \nabla_s \mu d\mathbf{s} = -M \int_S |\nabla_s \mu|^2 d\mathbf{s} \le 0. \end{aligned} \tag{6}$$

Here, ∂S is the boundary of S and $\mathbf{n}$ is the outward normal vector at the boundary. The solution $\phi(\mathbf{s}, t)$ is assumed to be smooth. Thus, Eqs. (5) and (6) can be guaranteed to hold by the boundary condition and divergence theorem. The CH equation, which describes phase separations, has been widely used in many fields such as the physical and materials sciences [2–4], multi-phase fluid dynamics [5–7], biology simulations [8,9], image inpainting processing [10], and surface/volume reconstruction [11]. Various experimental studies have shown that interesting phase separations could occur on static or dynamic surfaces; specific works include investigations of phase separation on lipid bilayer membranes [12], crystal growth on curved surfaces [13], phase separation patterns for diblock copolymers on surfaces [14,15] and phase separations within thin films [16]. The simulation of phase separation kinetics on surfaces constitutes a difficult problem. First, when solving the CH equation on a curved surface, it is difficult to compute the Laplace–Beltrami operator. Another difficulty lies in obtaining an accurate and efficient numerical scheme on surfaces that have high-order accuracy and strong numerical stability because the CH equation involves fourth-order spatial derivatives and a nonlinear term.

1.1. Definition of Laplacian on an arbitrary surface

Existing methods for estimating the Laplacian operator and solving partial differential equations on an arbitrary surface can be classified into two types: embedded narrow-band methods and direct methods. The former extends the partial differential equations to a higher-dimensional domain, then choose a narrow band around the surface and modify the differential operators so that the solution is confined to the curved surface (e.g., [17–29]). The latter methods use coordinates that are intrinsic to the surface and a surface-based mesh to discretize the differential operators, (e.g., [30–37]).

Osher and Sethian [22] and Bertalmio et al. [23] introduced a level set method for performing diffusion calculations on implicit surfaces. A surface is defined as the zero-level set of a higher-dimensional function; this surface is extended to a narrow-band domain embedding the surface. The Laplace–Beltrami operator is replaced by the standard Laplace operator, which can be discretized using an Eulerian formulation and simply computed by the finite difference formulas. It is not difficult to show the second-order convergence of the discretization. Another popular embedded narrow-band approach is the closed point method [24–27], which is applicable to many surface geometries and is easy to program using well-known standard numerical methods on a Cartesian grid. This method is based on a closest point function and supports open surfaces with boundary conditions. Recently, methods using radial functions have also been introduced [28,29].

Note that all embedded narrow-band methods require additional computational cost for arbitrary hypersurfaces with complicated topological or geometrical structures. Direct methods can reduce the additional computational cost because

the resulting discretization scheme depends only on the dimension of the continuous problem. It is well known that the Laplace–Beltrami operator is closely related to the mean curvature normal, which plays a key role in the study of geometric properties. Therefore, many methods [30–37] have been proposed for studying the Laplace–Beltrami operator on triangular or polygonal meshes because the use of such meshes is very popular in the processing of three-dimensional models. For example, Xu [31] investigated the convergence properties of several popular discretizations and proved that his two proposed methods satisfy a convergence property under certain restrictive conditions. In [33,34], Belkin et al. proposed a simple method for approximating integrals over a surface using a mesh and analyzed the proposed algorithm for approximating Laplace–Beltrami operator on a surface with a pointwise convergence guarantee for arbitrary meshes. Using the intrinsic geometric local tangential lifting method, Chen and Wu [36] proposed an intrinsic simple method for estimating the gradient, divergence, and the Laplace–Beltrami operator of a function on a regular surface. The authors also proved that the proposed method satisfies the conservation law and convergence property. Subsequently, the same authors [37] proposed a new simple discrete approximation employing the concept of local dual meshes.

The finite element technique [38,39] is another popular numerical method for solving partial differential equations [40–42]. Here, the Laplace–Beltrami operator is usually represented in terms of the tangential gradient by projecting the space function gradient in three-dimensional space onto a surface mesh consisting of a piecewise planar triangular tangent plane. However, approaches based on the finite element method are not easy to implement.

In this study, we modify the direct method described in [37] because its discretization method has the convergence property and is simple to implement. There is no requirement for the mesh to be well-centered; however mesh vertices should admit a one-ring neighborhood, because the direct method uses information from a one-ring of neighboring vertices. It should be noted that direct methods are limited by the quality of the surface mesh. In general, to obtain a good solution, a high quality mesh is required for direct methods and a finer mesh grid for embedded narrow-band methods. Owing to work in [43–46], mesh quality has been significantly improved.

1.2. Numerical scheme for the CH equation on surface

Many studies have been published regarding numerical methods for the CH equation on a flat surface [47–56]. Regarding time discretization, most of the schemes referenced above have used explicit, implicit–explicit, Crank–Nicolson, or Adams–Bashforth methods. Because the CH equation involves fourth-order spatial derivatives and a nonlinear term, explicit time schemes lead to severe time-step restrictions for stability. For a fully implicit approach, a larger time step can be employed. However, this only has first-order accuracy in time, and therefore requires small time steps to guarantee its accuracy. To obtain a high-order accurate numerical solution, the Crank–Nicolson and Adams–Bashforth methods are better choices. Because of the nonlinear term $F'(\phi)$, the usual second-order Crank–Nicolson or Adams–Bashforth schemes do not satisfy an energy law. Therefore, care should be taken to obtain a high-order time-accurate scheme with strong energy stability.

To this point, a few studies of the CH equation on a surface have presented detailed numerical algorithms. In [41], Du et al. demonstrated the well-posedness and convergence of a fully discrete finite element approximation scheme for the CH equation defined on a general surface. The proposed method has second-order accuracy, however involves time-step restrictions. A method for solving the CH equation over discrete surfaces was recently presented by Chen and Wu [36], but that paper only described the Laplace–Betltrami operator and described neither the details of the model nor the numerical algorithm for the CH equation. These details are presented in this paper, in which we also propose a simple high-order method for estimating the gradient operator. Furthermore, we believe these to be the first results concerning unconditionally stable second-order time discretizations of the CH equation on a surface.

The main objective of this study is to propose an unconditionally energy-stable second-order time-accurate scheme for the CH equation on surfaces. We demonstrate that our scheme has second-order accuracy in time and space. The discrete system can satisfy mass conservation and can use large time steps owing to its unconditional energy stability. We solve the resulting system of discrete equations using a biconjugate gradient stabilized method [57]. We demonstrate the performance of our proposed algorithm in several numerical experiments.

This paper is organized as follows. In Section 2, we present our numerical solution. Section 3 details our numerical experiments. Finally, conclusions are drawn in Section 4.

2. Numerical solution

Here, we briefly review a method of defining the discrete gradient, divergence, and Laplace–Beltrami operator on functions on a triangular mesh and propose our new high-order method. To prepare for our description, we first introduce some basic notations. Here, we closely follow the framework introduced in [36]. Let $S = (V,\ F)$ be a triangular surface mesh, where $V = \{\mathbf{v}_i | 1 \le i \le N_V\}$ and $F = \{T_k | 1 \le k \le N_F\}$ are lists of vertices and triangles, respectively. N_V and N_F are the numbers of vertices and triangles. For a vertex $\mathbf{v} \in V$, let $\mathbf{v}_j$ denote neighboring vertices of $\mathbf{v}$ for $j = 0, 1, \ldots, n$ and $\mathbf{v}_0 = \mathbf{v}_n$. Vertices $\mathbf{v}_j$ are labeled in a counterclockwise manner. Let T_j be the triangle with vertices $\mathbf{v}$, $\mathbf{v}_j$, and $\mathbf{v}_{j+1}$ (see Fig. 2). Let $\mathbf{N}(\mathbf{v}) = \sum_{j=0}^{n-1} \omega_j N_j / \| \sum_{j=0}^{n-1} \omega_j \mathbf{N}_j \|$ be the normal vector at $\mathbf{v} \in V$, where $\mathbf{N}_j$ is the unit normal to triangle T_j, $\omega_j = \|\mathbf{G}_j - \mathbf{v}\|^{-2} / \sum_{i=0}^{n-1} \|\mathbf{G}_i - \mathbf{v}\|^{-2}$, and $\mathbf{G}_j = (\mathbf{v}_j + \mathbf{v}_{j+1} + \mathbf{v})/3$. Let $P(\mathbf{v})$ be the local tangential polygon of $\mathbf{v}$, formed by vertices $\{\mathbf{0}, \tilde{\mathbf{v}}_0, \tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_{n-1}\}$, where $\tilde{\mathbf{v}}_j = \mathbf{v}_j - \mathbf{v} - \big\langle \mathbf{v}_j - \mathbf{v}, N(\mathbf{v}) \big\rangle N(\mathbf{v}),$ for $j = 0, \ldots, n-1$.

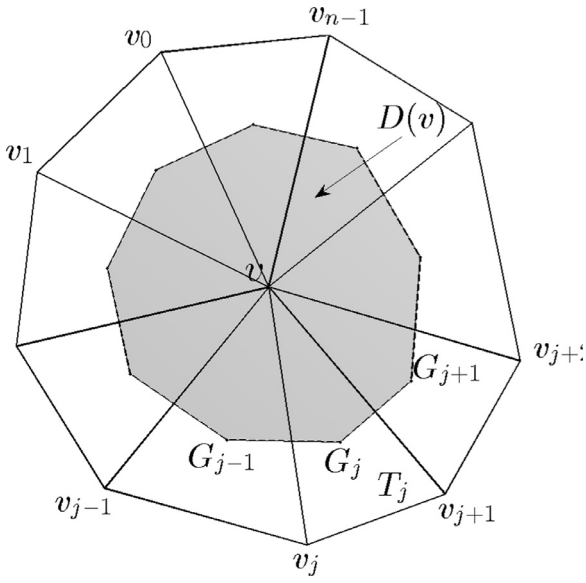

Fig. 2. A vortex $\mathbf{v}$ with several neighboring points for evaluating the gradient, divergence, and Laplace–Beltrami operators.

2.1. Discretizations of the gradient, divergence, and Laplace–Beltrami operators

We now describe the discretizations of the gradient, divergence, and Laplace–Beltrami operators of a function ϕ defined on the set of vertices V. Let $\tilde{\phi}_{\mathbf{v}}$ be a locally lifted function of ϕ on the vertices in $P(\mathbf{v})$, i.e., $\tilde{\phi}_{\mathbf{v}}(\tilde{\mathbf{v}}_j) = \phi(\mathbf{v}_j)$ and $\tilde{\phi}_{\mathbf{v}}(\mathbf{0}) = \phi(\mathbf{v})$. Let $\tilde{T}_j$ be the lifted triangle with vertices $\mathbf{0}$, $\tilde{\mathbf{v}}_j$, and $\tilde{\mathbf{v}}_{j+1}$. For $\mathbf{p} = a\tilde{\mathbf{v}}_j + b\tilde{\mathbf{v}}_{j+1} \in \tilde{T}_j$, we define $\tilde{\phi}_{\mathbf{v}}(\mathbf{p}) = a\tilde{\phi}_{\mathbf{v}}(\tilde{\mathbf{v}}_j) + b\tilde{\phi}_{\mathbf{v}}(\tilde{\mathbf{v}}_{j+1}) + (1-a-b)\tilde{\phi}_{\mathbf{v}}(\mathbf{0})$. To obtain the discrete gradient of ϕ on S at $\mathbf{v}$, we use the weighted combination method:

$$\nabla_d \phi(\mathbf{v}) = \nabla_d \tilde{\phi}_{\mathbf{v}}(\mathbf{0}) = \sum_{j=1}^{n-1} \tilde{\omega}_j \nabla_d \tilde{\phi}_{\mathbf{v}j}(\mathbf{0}), \tag{7}$$

where $\tilde{\omega}_j = \|\tilde{\mathbf{v}}_j + \tilde{\mathbf{v}}_{j+1}\|^{-2} / \sum_{i=0}^{n-1} \|\tilde{\mathbf{v}}_i + \tilde{\mathbf{v}}_{i+1}\|^{-2}$ is the weighting value. The gradient of $\phi(\mathbf{v})$ at triangle T_j is defined as $\nabla_d \tilde{\phi}_{\mathbf{v}j}(\mathbf{0}) = \alpha\tilde{\mathbf{v}}_j + \beta\tilde{\mathbf{v}}_{j+1}$, where

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \langle \tilde{\mathbf{v}}_j, \tilde{\mathbf{v}}_j \rangle & \langle \tilde{\mathbf{v}}_j, \tilde{\mathbf{v}}_{j+1} \rangle \\ \langle \tilde{\mathbf{v}}_j, \tilde{\mathbf{v}}_{j+1} \rangle & \langle \tilde{\mathbf{v}}_{j+1}, \tilde{\mathbf{v}}_{j+1} \rangle \end{pmatrix}^{-1} \begin{pmatrix} \tilde{\phi}_{\mathbf{v}}(\tilde{\mathbf{v}}_j) - \tilde{\phi}_{\mathbf{v}}(\mathbf{0}) \\ \tilde{\phi}_{\mathbf{v}}(\tilde{\mathbf{v}}_{j+1}) - \tilde{\phi}_{\mathbf{v}}(\mathbf{0}) \end{pmatrix}.$$

Let $\hat{T}_j$ be a triangle with $\mathbf{v}_j$, $\mathbf{G}_j$, and $\mathbf{G}_{j+1}$. The discrete divergence $\nabla_d \cdot \mathbf{X}$ of a vector field $\mathbf{X}$ at $\mathbf{v}$ is defined by

$$\begin{aligned} \nabla_d \cdot \mathbf{X}(\mathbf{v}) = \frac{1}{D(\mathbf{v})} \sum_{j=0}^{n-1} \frac{\|\mathbf{G}_{j+1} - \mathbf{G}_j\|}{6} \big(& 2\langle \mathbf{X}(\mathbf{G}_j),\ \mathbf{n}(\mathbf{G}_j) \rangle + 2\langle \mathbf{X}(\mathbf{G}_{j+1}),\ \mathbf{n}(\mathbf{G}_{j+1}) \rangle \\ & + \langle \mathbf{X}(\mathbf{G}_j),\ \mathbf{n}(\mathbf{G}_{j+1}) \rangle + \langle \mathbf{X}(\mathbf{G}_{j+1}),\ \mathbf{n}(\mathbf{G}_j) \rangle \big), \end{aligned}$$

where $D(\mathbf{v}) = \sum_{i=0}^{n-1} |\hat{T}_i|$, $|\hat{T}_i|$ is the area of $\hat{T}_i$, $\mathbf{X}(\mathbf{G}_j) = (\mathbf{X}(\mathbf{v}) + \mathbf{X}(\mathbf{v}_j) + \mathbf{X}(\mathbf{v}_{j+1}))/3$,

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

Finally, we have the approximating Laplace–Beltrami operator

$$\begin{aligned} \Delta_d \phi(\mathbf{v}) &= \nabla_d \cdot [\nabla_d \phi(\mathbf{v})] \\ &= \frac{1}{D(\mathbf{v})} \sum_{j=0}^{n-1} \frac{\|\mathbf{G}_{j+1} - \mathbf{G}_j\|}{6} \big(2\langle \nabla_d \phi(\mathbf{G})_j,\ \mathbf{n}(\mathbf{G}_j) \rangle + 2\langle \nabla_d \phi(\mathbf{G})_{j+1},\ \mathbf{n}(\mathbf{G}_{j+1}) \rangle \\ &\quad + \langle \nabla_d \phi(\mathbf{G})_j,\ \mathbf{n}(\mathbf{G}_{j+1}) \rangle + \langle \nabla_d \phi(\mathbf{G})_{j+1},\ \mathbf{n}(\mathbf{G}_j) \rangle \big). \end{aligned} \tag{8}$$

The definition of the Laplace–Beltrami operator in Eq. (8), requires the gradient of $\phi(\mathbf{G}_j)$ at the centroid $\mathbf{G}_j$ of each triangle, i.e., $\nabla_d \phi(\mathbf{G}_j)$. The gradient of $\phi(\mathbf{G}_j)$ can be computed as $\nabla_d \phi(\mathbf{G}_j) = (\nabla_d \phi(\mathbf{v}) + \nabla_d \phi(\mathbf{v}_j) + \nabla_d \phi(\mathbf{v}_{j+1}))/3$. However, in

this case $\nabla_d\phi(\mathbf{G})$ only has first-order accuracy. An alternative method for computing gradient $\tilde{\nabla}_d\phi(\mathbf{G})$ with second-order accuracy was proposed by Chen and Wu [36], a detailed description of which can be found in [37]:

$$\tilde{\nabla}_d\phi(\mathbf{G}_j) = A_j^{-1}\begin{pmatrix}\alpha_1(\phi(\mathbf{v})-\phi(\mathbf{G}_j))+\alpha_2(\phi(\mathbf{v}_j)-\phi(\mathbf{G}_j))+\alpha_3(\phi(\mathbf{v}_{j+1})-\phi(\mathbf{G}_j))\\ \beta_1(\phi(\mathbf{v})-\phi(\mathbf{G}_j))+\beta_2(\phi(\mathbf{v}_j)-\phi(\mathbf{G}_j))+\beta_3(\phi(\mathbf{v}_{j+1})-\phi(\mathbf{G}_j))\end{pmatrix},$$

$$\text{where } A_j = \begin{pmatrix}\alpha_1x_1+\alpha_2x_2+\alpha_3x_3 & \alpha_1y_1+\alpha_2y_2+\alpha_3y_3\\ \beta_1x_1+\beta_2x_2+\beta_3x_3 & \beta_1y_1+\beta_2y_2+\beta_3y_3\end{pmatrix}.$$

Here, $\phi(\mathbf{G}_j) = (\phi(\mathbf{v})+\phi(\mathbf{v}_j)+\phi(\mathbf{v}_{j+1}))/3$ and (x_1, y_1) are the orthonormal coordinates for vectors $\mathbf{v}-\mathbf{G}_j$, i.e., $\mathbf{v}-\mathbf{G}_j = x_1\mathbf{e}_1 + y_1\mathbf{e}_2$. Here, $\mathbf{e}_1$ and $\mathbf{e}_2$ form an orthonormal basis for the triangle plane T_j, which contains the three vertexes $\mathbf{v}$, $\mathbf{v}_j$, and $\mathbf{v}_{j+1}$. In addition, (x_2, y_2) and (x_3, y_3) are the orthonormal coordinates for the vectors $\mathbf{v}_j-\mathbf{G}_j$ and $\mathbf{v}_{j+1}-\mathbf{G}_j$, respectively, and can be computed in the same manner. Furthermore, α_1, α_2, α_3, β_1, β_2, and β_3 satisfy the following relations:

$$\begin{pmatrix}x_1y_1 & x_2y_2 & x_3y_3\\ x_1^2 & x_2^2 & x_3^2\\ y_1^2 & y_2^2 & y_3^2\end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\\ \alpha_3\end{pmatrix} = \begin{pmatrix}0\\ 0\\ 0\end{pmatrix} \text{ and } \begin{pmatrix}x_1y_1 & x_2y_2 & x_3y_3\\ x_1^2 & x_2^2 & x_3^2\\ y_1^2 & y_2^2 & y_3^2\end{pmatrix}\begin{pmatrix}\beta_1\\ \beta_2\\ \beta_3\end{pmatrix} = \begin{pmatrix}0\\ 0\\ 0\end{pmatrix}.$$

In this manner, $\tilde{\nabla}_d\phi(\mathbf{G}_j)$ is computed with second-order accuracy. For further details, we refer the reader to Chen and Wu [36], 37]. However, the computation of $\tilde{\nabla}_d\phi(\mathbf{G}_j)$ is not simple. The solutions of the above equation are not unique; thus, care should be taken to choose suitable values of α_1, α_2, α_3, β_1, β_2, and β_3 to ensure that matrix A_j has full rank. To remove this difficulty and compute the gradient of $\phi(\mathbf{G}_j)$ with high accuracy, we propose a second-order-accurate method. Starting with a Taylor expansion, we can obtain the following:

$$\begin{cases}\phi(\mathbf{v})-\phi(\mathbf{G}_j) = \langle\nabla_d\phi(\mathbf{G}_j), \mathbf{v}-\mathbf{G}_j\rangle + 0.5\Delta_d\phi(\mathbf{G}_j)\|\mathbf{v}-\mathbf{G}_j\|^2 + O(\|\mathbf{v}-\mathbf{G}_j\|^2),\\ \phi(\mathbf{v}_j)-\phi(\mathbf{G}_j) = \langle\nabla_d\phi(\mathbf{G}_j), \mathbf{v}_j-\mathbf{G}_j\rangle + 0.5\Delta_d\phi(\mathbf{G}_j)\|\mathbf{v}_j-\mathbf{G}_j\|^2 + O(\|\mathbf{v}_j-\mathbf{G}_j\|^2),\\ \phi(\mathbf{v}_{j+1})-\phi(\mathbf{G}_j) = \langle\nabla_d\phi(\mathbf{G}_j), \mathbf{v}_{j+1}-\mathbf{G}_j\rangle + 0.5\Delta_d\phi(\mathbf{G}_j)\|\mathbf{v}_{j+1}-\mathbf{G}_j\|^2 + O(\|\mathbf{v}_{j+1}-\mathbf{G}_j\|^2).\end{cases} \tag{9}$$

The Taylor expansion theorem used in Eq. (9) can also be similarly found in [37]. Because $\nabla_d\phi(\mathbf{G}_j)$, $\mathbf{v}_j-\mathbf{G}_j$, and $\mathbf{v}_{j+1}-\mathbf{G}_j$ are in the same triangle plane, we can assume that $\nabla_d\phi(\mathbf{G}_j)$ takes the following form

$$\nabla_d\phi(\mathbf{G}_j) = \alpha_j(\mathbf{v}_j-\mathbf{G}_j) + \beta_j(\mathbf{v}_{j+1}-\mathbf{G}_j), \tag{10}$$

where α_j and β_j are constants. Then, plugging Eq. (10) into Eq. (9) and combining the six relations together, we arrive at

$$\begin{pmatrix}\alpha_j\\ \beta_j\\ 0.5\Delta_d\phi(\mathbf{G}_j)\end{pmatrix} = B_j^{-1}\begin{pmatrix}\phi(\mathbf{v})-\phi(\mathbf{G}_j)\\ \phi(\mathbf{v}_j)-\phi(\mathbf{G}_j)\\ \phi(\mathbf{v}_{j+1})-\phi(\mathbf{G}_j)\end{pmatrix}, \tag{11}$$

$$\text{where } B_j = \begin{pmatrix}\langle\mathbf{v}_j-\mathbf{G}_j, \mathbf{v}-\mathbf{G}_j\rangle & \langle\mathbf{v}_{j+1}-\mathbf{G}_j, \mathbf{v}-\mathbf{G}_j\rangle & \|\mathbf{v}-\mathbf{G}_j\|^2\\ \langle\mathbf{v}_j-\mathbf{G}_j, \mathbf{v}_j-\mathbf{G}_j\rangle & \langle\mathbf{v}_{j+1}-\mathbf{G}_j, \mathbf{v}_j-\mathbf{G}_j\rangle & \|\mathbf{v}_j-\mathbf{G}_j\|^2\\ \langle\mathbf{v}_j-\mathbf{G}_j, \mathbf{v}_{j+1}-\mathbf{G}_j\rangle & \langle\mathbf{v}_{j+1}-\mathbf{G}_j, \mathbf{v}_{j+1}-\mathbf{G}_j\rangle & \|\mathbf{v}_{j+1}-\mathbf{G}_j\|^2\end{pmatrix}. \tag{12}$$

After calculating α_j and β_j and substituting them into Eq. (10), we can compute $\nabla_d\phi(\mathbf{G}_j)$. Note that it is not difficult to prove that matrix B_j has full rank; therefore, B_j^{-1} exists. Furthermore, it follows from Eq. (11) that $\Delta_d\phi(\mathbf{G}_j)$ is zero. The proof is not presented here. Thus, using Eqs. (10)–(12), we find that $\nabla_d\phi(\mathbf{G}_j)$ has second-order accuracy in space. Then, we obtain vector field $\nabla_d\phi(\mathbf{G}_j)$ defined on all triangle centroids, which allows us to define the approximating Laplace–Beltrami operator (8). Furthermore, by observing Eqs. (8) and (10)–(12), we can rewrite the Laplace–Beltrami operator $\Delta_d\phi(\mathbf{v})$ at vertex $\mathbf{v}$ as

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

where $\mathcal{L}_i$ is a $1\times N_v$ matrix for $i = 1, 2, \ldots, N_v$, which can be obtained by Eqs. (8), and (10)–(12). Let $\mathcal{L} = (\mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_{N_V})^T$ be an $N_V\times N_V$ matrix, which is called the Laplace matrix of surface S in this paper.

2.2. Proposed scheme

For the sake of convenience, let $M = 1$. Let ϕ_i^n be the approximation of $\phi(\mathbf{v}_i, n\Delta t)$ at vertex $\mathbf{v}_i$, 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. To obtain a high-order numerical solution, we apply a Crank–Nicolson discretization to Eqs. (1) and (2). Then, we have the following discrete system of equations:

$$\frac{\phi^{n+1}-\phi^n}{\Delta t} = \Delta_d\mu^{n+\frac{1}{2}}, \tag{13}$$

$$\mu^{n+\frac{1}{2}} = \frac{1}{2}(F'(\phi^{n+1}) + F'(\phi^n)) - \frac{\epsilon^2}{2}(\Delta_d\phi^{n+1} + \Delta_d\phi^n). \tag{14}$$

It is well known that owing to the nonlinear term $F'(\phi)$, the usual second-order Crank–Nicolson scheme does not satisfy an energy law. To construct a second-order scheme that does satisfy an energy law, we divide the nonlinear terms $F'(\phi^{n+1})$ into $F'(\phi^{n+1}) - \lambda\phi^{n+1}$ and $\lambda\phi^{n+1}$ and then linearize $F'(\phi^{n+1}) - \lambda\phi^{n+1}$ using information from the $n-1$ and n levels, i.e., $F'(\phi^{n+1}) - \lambda\phi^{n+1} = 2(F'(\phi^n) - \lambda\phi^n) - (F'(\phi^{n-1}) - \lambda\phi^{n-1})$. Here, λ is a positive stabilizing parameter. Thus, we obtain the following equations:

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \Delta_d\mu^{n+\frac{1}{2}}, \tag{15}$$

$$\mu^{n+\frac{1}{2}} = \frac{3}{2}(F'(\phi^n) - \lambda\phi^n) - \frac{1}{2}(F'(\phi^{n-1}) - \lambda\phi^{n-1}) + \frac{\lambda}{2}(\phi^{n+1} + \phi^n) - \frac{\epsilon^2}{2}(\Delta_d\phi^{n+1} + \Delta_d\phi^n). \tag{16}$$

The above scheme is of second-order accuracy. A detailed proof of the stability of the above scheme will be presented in Section 2.3. Note that a similar fully implicit and second-order accurate scheme can be found in [58], where the authors directly solved the nonlinear term $F'(\phi^{n+1})$ using a Newton-type iterative method. To solve the linear discrete system (15) and (16) at the implicit time level, we use a biconjugate gradient stabilized method [57]. To condense this discussion, we only describe the preprocessing step by rewriting Eqs. (15) and (16) as follows:

$$\begin{pmatrix} \mathcal{I} & \Delta t\mathcal{L} \\ -\lambda/2\mathcal{I} + \epsilon^2/2\mathcal{L} & \mathcal{I} \end{pmatrix} \begin{pmatrix} \phi_1^{n+1} \\ \vdots \\ \phi_{N_V}^{n+1} \\ \mu_1^{n+\frac{1}{2}} \\ \vdots \\ \mu_{N_V}^{n+\frac{1}{2}} \end{pmatrix} = \begin{pmatrix} \phi_1^n \\ \vdots \\ \phi_{N_V}^n \\ \frac{3}{2}(F'(\phi_1^n) - \lambda\phi_1^n) - \frac{1}{2}(F'(\phi_1^{n-1}) - \lambda\phi_1^{n-1}) + \frac{\lambda}{2}\phi_1^n \\ \vdots \\ \frac{3}{2}(F'(\phi_{N_V}^n) - \lambda\phi_{N_V}^n) - \frac{1}{2}(F'(\phi_{N_V}^{n-1}) - \lambda\phi_{N_V}^{n-1}) + \frac{\lambda}{2}\phi_{N_V}^n \end{pmatrix} + \begin{pmatrix} \mathbf{0} & -\epsilon^2/2\mathcal{L} \end{pmatrix} \begin{pmatrix} \phi_1^n \\ \vdots \\ \phi_{N_V}^n \\ \phi_1^n \\ \vdots \\ \phi_{N_V}^n \end{pmatrix}. \tag{17}$$

Here, $\mathcal{I}$ and $\mathbf{0}$ are the $N_V \times N_V$ identity matrix and zero matrix, respectively. If the triangular surface mesh is without a boundary, we can directly obtain the solution for ϕ by solving Eq. (17) using a biconjugate gradient stabilized method. If the surface is open with a Neumann boundary condition, then we split the total collection of vertices into two groups: those in the surface boundary and those that are not. Following rearrangements, we assume that $\mathbf{v}_i$, $i = 1, 2, \ldots, m$ is the surface mesh and $\mathbf{v}_i$, $i = m, m+1, \ldots, N_V$ is in the boundary of the surface. Then, the boundary points satisfy $\mathbf{n} \cdot \nabla_d\phi(\mathbf{v}_i) = 0$. Here, we can compute $\Delta_d\phi(\mathbf{v}) = \nabla_d \cdot (\nabla_d\phi(\mathbf{v})) = \mathbf{n} \cdot \nabla_d\phi(\mathbf{v}_i) = 0$ on the surface boundary instead of solving the original equation. Therefore, to consider the Neumann boundary condition of the CH equation, we can write Eqs. (15) and (16) as

$$\begin{pmatrix} \mathcal{I}^* & \Delta t\mathcal{L} \\ -\lambda/2\mathcal{I}^* + \epsilon^2/2\mathcal{L}^I & \mathcal{I}^* \end{pmatrix} \begin{pmatrix} \phi_1^{n+1} \\ \vdots \\ \phi_m^{n+1} \\ \phi_{m+1}^{n+1} \\ \vdots \\ \phi_{N_V}^{n+1} \\ \mu_1^{n+\frac{1}{2}} \\ \vdots \\ \mu_m^{n+\frac{1}{2}} \\ \mu_{m+1}^{n+\frac{1}{2}} \\ \vdots \\ \mu_{N_V}^{n+\frac{1}{2}} \end{pmatrix} = \begin{pmatrix} \phi_1^n \\ \vdots \\ \phi_m^n \\ 0 \\ \vdots \\ 0 \\ \frac{3}{2}(F'(\phi_1^n) - \lambda\phi_1^n) - \frac{1}{2}(F'(\phi_1^{n-1}) - \lambda\phi_1^{n-1}) + \frac{\lambda}{2}\phi_1^n \\ \vdots \\ \frac{3}{2}(F'(\phi_m^n) - \lambda\phi_m^n) - \frac{1}{2}(F'(\phi_m^{n-1}) - \lambda\phi_m^{n-1}) + \frac{\lambda}{2}\phi_m^n \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} \mathbf{0} & -\epsilon^2/2\mathcal{L} \end{pmatrix} \begin{pmatrix} \phi_1^n \\ \vdots \\ \phi_m^n \\ \phi_{m+1}^n \\ \vdots \\ \phi_{N_V}^n \\ \phi_1^n \\ \vdots \\ \phi_m^n \\ \phi_{m+1}^n \\ \vdots \\ \phi_{N_V}^n \end{pmatrix}. \tag{18}$$

Here,

$$\mathcal{I}^* = \begin{pmatrix} \mathcal{I}^I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}$$

is an $N_V \times N_V$ identity matrix and $\mathcal{I}^I$ is an $m \times m$ identity matrix. Finally, we summarize our process as follows: (i) If the triangle surface is without a boundary, then we use Eq. (17). (ii) If the triangle surface has a Neumann boundary condition,

then we use Eq. (18). Note that our method can be extended to cases with Dirichlet and periodic boundary conditions as well.

2.3. Unconditional energy stability

Before we proceed, we must define inner products and provide summation-by-parts formulas for deriving and analyzing the proposed numerical schemes. Define the discrete l_2 inner product at the vertex and centroid of the triangle face by

$$(\phi,\psi)_d=\sum_i^{N_V}\phi_i\psi_i D_i \quad \text{and} \quad (\nabla_d\phi,\nabla_d\psi)_d=\sum_i^{N_V}\left[D_i\sum_{j=0}^{n-1}\nabla_d\phi(\mathbf{G}_{ij})\cdot\nabla_d\psi(\mathbf{G}_{ij})\right].$$

Here, $D_i=D(\mathbf{v}_i)$ for $i=1,2,\ldots,N_V$. We can also define the discrete norm as $\|\phi\|_d^2=(\phi,\phi)_d$ and $\|\nabla_d\phi\|_d^2=(\nabla_d\phi,\nabla_d\phi)_d$. As in the semi-discrete case, we define the discrete total energy functional by

$$\mathcal{E}^d(\phi^n)=(F(\phi^n),\mathbf{1})_d+\frac{\epsilon^2}{2}(\nabla_d\phi^n,\nabla_d\phi^n)_d. \tag{19}$$

Here, $\mathbf{1}$ is a vector with all entries equal to 1. Furthermore, we define the discrete pseudo energy as

$$\tilde{\mathcal{E}}^d(\phi^{n+1},\phi^n)=\mathcal{E}^d(\phi^{n+1})+\frac{1}{4}(\lambda+1+3(\hat{\varphi}^n)^2-6(\hat{\psi}^{n+1})^2)\|\phi^{n+1}-\phi^n\|_d^2. \tag{20}$$

Here, $\hat{\psi}^{n+1}$ and $\hat{\varphi}^n$ are constants satisfying

$$\left(F'(\phi^n),\ \phi^{n+1}-\phi^n\right)_d=\left(F(\phi^{n+1})-F(\phi^n),\ \mathbf{1}\right)_d-\frac{1}{2}F''(\hat{\psi}^{n+1})\|\phi^{n+1}-\phi^n)\|_d^2, \tag{21}$$

$$\left(\frac{1}{2}F'(\phi^n)-\frac{1}{2}F'(\phi^{n-1}),\phi^{n+1}-\phi^n\right)_d=\frac{1}{2}F''(\hat{\psi}^n)\left(\phi^n-\phi^{n-1},\phi^{n+1}-\phi^n\right)_d. \tag{22}$$

In this work, we restrict our attention to the order parameter ϕ which is bounded. That is, there exists a constant $\mathcal{M}\geq 1$ such that $|\phi|\leq\mathcal{M}$. We note that this condition can be satisfied by many physically relevant potentials by restricting the growth of $F(\phi)$ to quadratic for $|\phi|\geq\mathcal{M}$ [59]. However, because the CH equation does not satisfy the maximum principle [60] and $F(\phi)$ exhibits quartic growth at infinity, it is very difficult to analytically prove that the solution for the CH equation is bounded. In [59,61], the maximum norm of the solution for the CH equation was proved to be bounded with a truncated potential. We refer the interested reader to Shen and Yang [59] and Condette et al. [61]. Furthermore, it should be noted that if the maximum norm of initial condition ϕ_0 is bounded by 1, then the solution for the CH equation with original double-well potential is also bounded based on our numerical experiments. Now, we prove the unconditional energy stability of these schemes under the condition $\lambda\geq 6\mathcal{M}^2-1$. First, we state Lemma 1 without proof, which can be obtained simply by summation-by-parts formulas and the discrete first and second Green's identities. We refer the interested reader to Gilbarg and Trudinger [62] for a similar proof for Lemma 1.

Lemma 1. *Let ϕ and ψ be defined on surface S, which is assumed to be closed. Then, $(\phi,\Delta_d\psi)_d=(\psi,\Delta_d\phi)_d=-(\nabla_d\psi,\nabla_d\phi)_d$.*

Lemma 2. *Under the condition that $\lambda\geq 6\mathcal{M}^2-1$, three solutions $(\phi^{n+1},\phi^n,\phi^{n-1})$ of the conservative scheme (15) and (16) satisfy:*

$$\mathcal{E}^d(\phi^{n+1})\leq\tilde{\mathcal{E}}^d(\phi^{n+1},\phi^n). \tag{23}$$

Proof. By Eqs. (19) and (20), we can show that

$$\mathcal{E}^d(\phi^{n+1})-\tilde{\mathcal{E}}^d(\phi^{n+1},\phi^n)=-\frac{1}{4}(\lambda+1+3(\hat{\varphi}^n)^2-6(\hat{\psi}^{n+1})^2)\|\phi^{n+1}-\phi^n\|_d^2\leq 0. \tag{24}$$

which proves Lemma 2. □

Theorem 1. *If ϕ^{n+1}, ϕ^n, and ϕ^{n-1} are the solutions of Eqs. (15) and (16), then under the condition $\lambda\geq 6\mathcal{M}^2-1$, the following energy law holds for any time step Δt:*

$$\tilde{\mathcal{E}}^d(\phi^{n+1},\phi^n)\leq\tilde{\mathcal{E}}^d(\phi^n,\phi^{n-1}). \tag{25}$$

Proof. By multiplying Eq. (15) by $\mu^{n+\frac{1}{2}}$ and summing by parts, we obtain

$$(\phi^{n+1}-\phi^n,\mu^{n+\frac{1}{2}})_d=\Delta t(\Delta_d\mu^{n+\frac{1}{2}},\ \mu^{n+\frac{1}{2}})_d=-\Delta t\|\nabla_d\mu^{n+\frac{1}{2}}\|_d^2.$$

By multiplying Eq. (16) by $\phi^{n+1}-\phi^n$ and summing by parts, we have

$$\left(\frac{\epsilon^2}{2}\Delta_d\phi^{n+1}+\frac{\epsilon^2}{2}\Delta_d\phi^n,\ \phi^{n+1}-\phi^n\right)_d=-\frac{\epsilon^2}{2}\|\nabla_d\phi^{n+1}\|_d^2+\frac{\epsilon^2}{2}\|\nabla_d\phi^n\|_d^2, \tag{26}$$

$$\frac{\lambda}{2}\left(\phi^{n+1}-2\phi^{n}+\phi^{n-1},\phi^{n+1}-\phi^{n}\right)_d=\frac{\lambda}{2}\left(\|\phi^{n+1}-\phi^{n}\|_d^2-\|\phi^{n}-\phi^{n-1}\|_d^2+\|\phi^{n+1}-2\phi^{n}+\phi^{n-1}\|_d^2\right), \tag{27}$$

$$\begin{aligned}\left(F'(\phi^n),\ \phi^{n+1}-\phi^n\right)_d&=\left(F(\phi^{n+1})-F(\phi^n),\ \mathbf{1}\right)_d-\left(\frac{1}{2}F''(\psi^{n+1})(\phi^{n+1}-\phi^n),\phi^{n+1}-\phi^n\right)_d\\&=\left(F(\phi^{n+1})-F(\phi^n),\ \mathbf{1}\right)_d-\frac{1}{2}F''(\hat{\psi}^{n+1})\|\phi^{n+1}-\phi^n\|_d^2.\end{aligned} \tag{28}$$

Here, we have used the Taylor expansion and mean value theorem for definite integrals in Eq. (28). Note that a set of constant values ψ^{n+1} and $\hat{\psi}^{n+1}$ exist because of the Taylor expansion and the mean value theorem, respectively. Similarly, we can also find that

$$\begin{aligned}&\left(\frac{1}{2}F'(\phi^n)-\frac{1}{2}F'(\phi^{n-1}),\phi^{n+1}-\phi^n\right)_d=\left(\frac{1}{2}F''(\psi^n)(\phi^n-\phi^{n-1}),\phi^{n+1}-\phi^n\right)_d\\&=\left(\frac{1}{2}F''(\psi^n)(\phi^n-\phi^{n-1}),\phi^n-\phi^{n-1}\right)_d+\left(\frac{1}{2}F''(\psi^n)(\phi^{n+1}-2\phi^n+\phi^{n-1}),\phi^n-\phi^{n-1}\right)_d\\&=\frac{1}{2}F''(\hat{\psi}^n)\left(\phi^n-\phi^{n-1},\phi^n-\phi^{n-1}\right)_d+\frac{1}{2}F''(\hat{\varphi}^n)\left(\phi^{n+1}-2\phi^n+\phi^{n-1},\phi^n-\phi^{n-1}\right)_d\\&=\frac{1}{2}F''(\hat{\psi}^n)\|\phi^n-\phi^{n-1}\|_d^2-\frac{F''(\hat{\varphi}^n)}{4}\left(\|\phi^{n+1}-\phi^n\|_d^2-\|\phi^n-\phi^{n-1}\|_d^2-\|\phi^{n+1}-2\phi^n+\phi^{n-1}\|_d^2\right)\end{aligned} \tag{29}$$

Here, ψ^n is constant for the Taylor expansion and $\hat{\varphi}^n$ is constant for the mean value theorem.

$$\begin{aligned}&\frac{1}{2}F''(\hat{\psi}^{n+1})\|\phi^{n+1}-\phi^n\|_d^2-\frac{1}{2}F''(\hat{\psi}^n)\|\phi^n-\phi^{n-1}\|_d^2-\frac{F''(\hat{\varphi}^n)}{4}(\|\phi^{n+1}-\phi^n\|_d^2-\|\phi^n-\phi^{n-1}\|_d^2\\&-\|\phi^{n+1}-2\phi^n+\phi^{n-1}\|_d^2)-\frac{\lambda}{4}\left(\|\phi^{n+1}-\phi^n\|_d^2-\|\phi^n-\phi^{n-1}\|_d^2+\|\phi^{n+1}-2\phi^n+\phi^{n-1}\|_d^2\right)\\&=\frac{\lambda+1-3(\hat{\varphi}^n)^2}{4}\|\phi^{n+1}-2\phi^n+\phi^{n-1}\|_d^2\\&\quad+\frac{\lambda+1+3(\hat{\varphi}^n)^2-6(\hat{\psi}^{n+1})^2}{4}\|\phi^{n+1}-\phi^n\|_d^2-\frac{\lambda+1+3(\hat{\varphi}^n)^2-6(\hat{\psi}^n)^2}{4}\|\phi^n-\phi^{n-1}\|_d^2\end{aligned}$$

By combining the above relations and using the definition of energy (19), we have that

$$\begin{aligned}\tilde{\mathcal{E}}^d(\phi^{n+1},\phi^n)-\tilde{\mathcal{E}}^d(\phi^n,\phi^{n-1})&=\frac{\epsilon^2}{2}\|\nabla_d\phi^{n+1}\|_d^2-\frac{\epsilon^2}{2}\|\nabla_d\phi^n\|_d^2+\left(F(\phi^{n+1})-F(\phi^n),\ \mathbf{1}\right)_d\\&\quad+\frac{\lambda+1+3(\hat{\varphi}^n)^2-6(\hat{\psi}^{n+1})^2}{4}\|\phi^{n+1}-\phi^n\|_d^2-\frac{\lambda+1+3(\hat{\varphi}^n)^2-6(\hat{\psi}^n)^2}{4}\|\phi^n-\phi^{n-1}\|_d^2\\&=-\Delta t\|\nabla_d\mu^{n+\frac{1}{2}}\|_d^2-\frac{\lambda+1-3(\hat{\varphi}^n)^2}{4}\|\phi^{n+1}-2\phi^n+\phi^{n-1}\|_d^2\le 0.\end{aligned} \tag{30}$$

Here, we have used the conditions that $\lambda\ge 6\mathcal{M}^2-1$ and $\lambda\ge 3(\hat{\varphi}^n)^2-1$. □

Theorem 2. *Suppose that $\{\phi^n,\mu^{n-\frac{1}{2}}\}_{n=1}^N$ is a sequence of solution pairs for the conservative scheme (15) and (16) with the initial values ϕ^0 and ϕ^{-1}, where $\phi^0=\phi^{-1}\in[-1,1]$. If $\lambda\ge 6\mathcal{M}^2-1$, then we have that*

$$\mathcal{E}^d(\phi^{n+1})\le\tilde{\mathcal{E}}^d(\phi^{n+1},\phi^n)\le\mathcal{E}^d(\phi^0). \tag{31}$$

Proof. By the last Theorem 1 and $\phi^0=\phi^{-1}\in[-1,1]$, we have a chain of inequalities,

$$\mathcal{E}^d(\phi^{n+1})\le\tilde{\mathcal{E}}^d(\phi^{n+1},\phi^n)\le\tilde{\mathcal{E}}^d(\phi^n,\phi^{n-1})\le\cdots\le\tilde{\mathcal{E}}^d(\phi^0,\phi^{-1})=\mathcal{E}^d(\phi^0).$$

Thus, the discrete version of the original energy is bounded and non-increasing in time, which follows from the above by the energy of the initial condition. Therefore, our proposed method is unconditionally energy-stable with a suitable stabilizing parameter λ.

The following points should be noted: (1) Our proposed scheme (15) and (16) is a three-level scheme that requires setting initial step ϕ^{-1} to guarantee second-order accuracy for every time step. In this study, $\phi^{-1}=\phi^0$ is simple, and lowers the accuracy in the first time step. However, for later calculations our scheme is indeed second-order accurate. Therefore, for a lengthy simulation, second-order accuracy with respect to time and space can be observed. (2) We can easily design a fully Crank–Nicolson scheme that is a second-order two-level scheme and has better stability. However, this requires solving a nonlinear equation at each time step. Compared to the fully Crank–Nicolson scheme, only an elliptic equation with constant coefficients is solved at each time step in our scheme, which makes it easy to implement. (3) In our proof, stabilizing parameter λ is chosen to be $\lambda\ge 6\mathcal{M}^2-1$ to make the energy strictly non-increasing. However, in practical simulation λ may only need to be set to $\lambda=3\mathcal{M}^2-1$ or even $\lambda=2$ for the initial condition $\phi^0\in[-1,1]$ because the absolute values of $\hat{\psi}^{n+1}$, $\hat{\psi}^n$, and $\hat{\varphi}^n$ may be much smaller than $\mathcal{M}$ in Eqs. (24) and (30). □

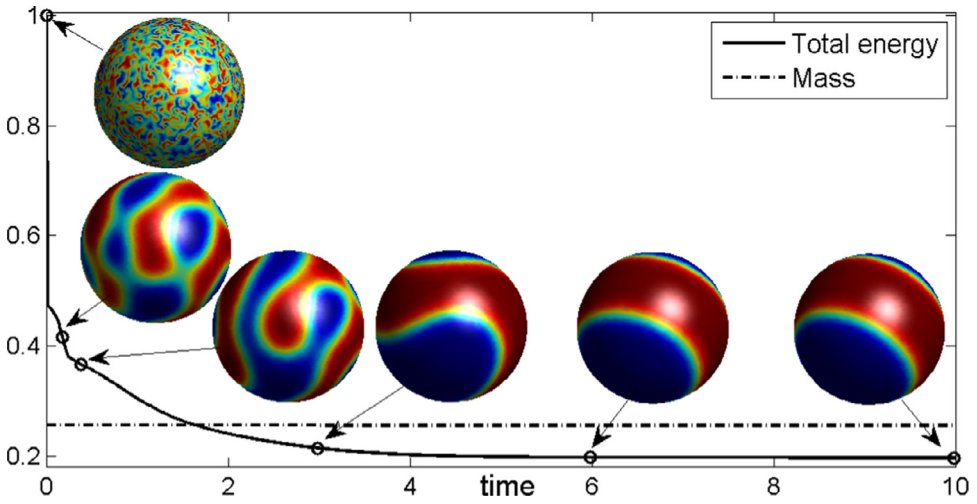

Fig. 3. The non-increasing discrete energy and mass conservation for our proposed method. The inset small figures represent the morphology of the phase field at the indicated times. Note that we have normalized the total energy by the total energy at the initial time.

2.4. Mass conservation

The total mass conservation is an important property of the CH equation. Here, we prove that our proposed method satisfies the mass conservation property, i.e., $(\phi^{n+1}, \mathbf{1})_d = (\phi^n, \mathbf{1})_d$. By combining Eq. (15) and the discrete version of integration by parts, we can obtain that

$$(\phi^{n+1}, \mathbf{1})_d - (\phi^n, \ \mathbf{1})_d = \Delta t(\Delta_d \mu^{n+\frac{1}{2}}, \mathbf{1})_d = \Delta t(\nabla_d \mu^{n+\frac{1}{2}}, \ \nabla_d \mathbf{1})_d = 0,$$

where we have used the Neumann boundary conditions for chemical potential and telescoping cancelation. Thus, our proposed method satisfies total mass conservation.

3. Numerical results

In this section, we perform the following numerical experiments to test the non-increasing discrete energy and mass conservation, the stability of the proposed scheme, its relation between time and space, spinodal decomposition on a poor-quality surface mesh, spinodal decomposition on an open surface with various boundary conditions, and microphase separation patterns on surfaces, and we perform a convergence test for time and space. Unless otherwise specified, $\epsilon = 0.1$ and $\lambda = 5$. To generate an oriented triangular mesh connecting with the surface points, we employ the distmesh algorithm [43].

3.1. Non-increasing discrete energy and mass conservation

We start with a numerical simulation to demonstrate the non-increasing discrete energy and mass conservation of our proposed method. The initial condition is set to $\phi(x, y, z, 0) = \text{rand}(x, y, z)$ on a unit sphere surface, where rand(x, y, z) is a random number between -1 and 1. The edge of every triangle on the sphere surface is almost the same as $h = 0.05$. The calculation is run until $t = 10$ with time step $\Delta t = 0.01$. The CH equation not only models phase separation, but also coarsening an already phase-separated solution, as shown in Fig. 3. From Fig. 3, we also observe that the total discrete energy is non-increasing and the average mass concentration of the numerical solution remains constant.

3.2. Stability of the proposed scheme

Because the CH equation contains fourth-order spatial derivatives and a nonlinear term, an explicit time scheme leads to severe time-step restrictions for stability. To demonstrate the stability of our proposed scheme (15) and (16), we perform a numerical experiment using a large time step, $\Delta t = 10000$. The initial condition and parameters are set as in Section 3.1. The calculations are run until time $T = 100000$ and the numerical solutions are presented in Fig. 4. Fig. 4 also shows the mass and total energy evolutions, which suggest that our proposed scheme is indeed unconditionally stable. Note that the much larger time step causes the solver of the biconjugate gradient stabilized method to stagnate, thus the order parameter is always zero, i.e., $\phi \equiv 0$. Therefore, in this test we use the biconjugate gradient method to force the numerical error to converge to the desired tolerance. As mentioned above, our proposed scheme is unconditionally stable and allows the use of large time steps. However, large time steps may cause less accurate results, since our scheme has second-order accuracy in time and space. Meanwhile, a smaller time step results in higher computational costs. To choose the best time step for

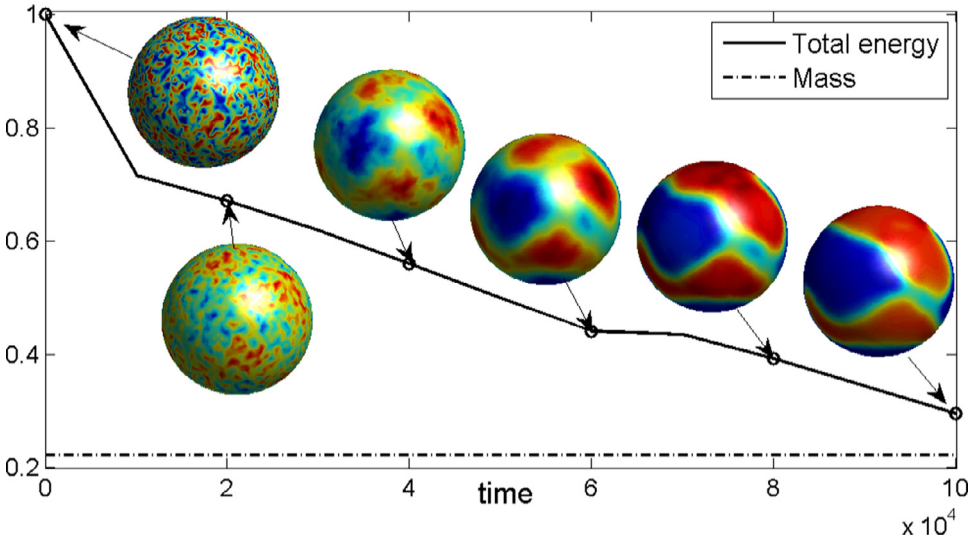

Fig. 4. Stability of the proposed scheme. The inset small figures show the morphology of the phase field at the indicated times. Note that we have normalized the total energy by the total energy at the initial time.

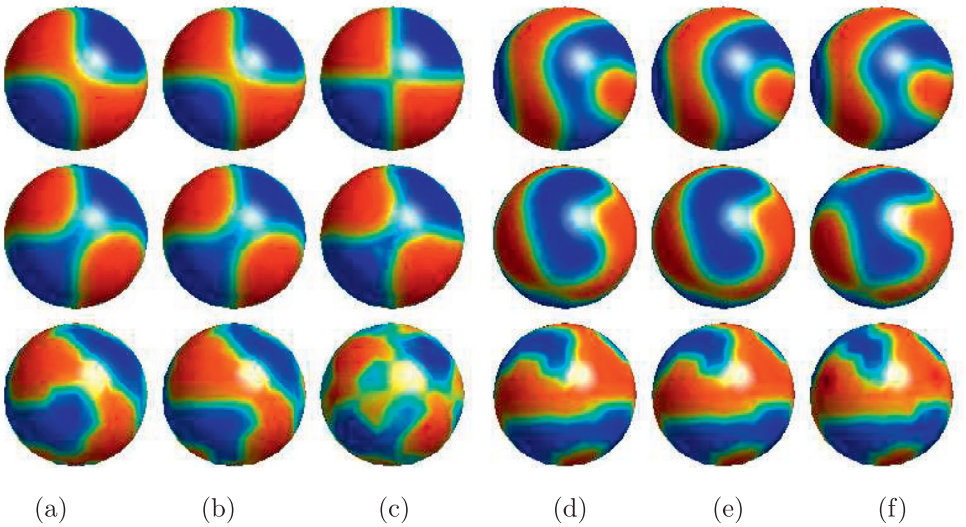

Fig. 5. Plots of ϕ at time $t = 0.8$ with different time steps and space. From top to bottom, these are the results with $h = 0.025$, $h = 0.05$, and $h = 0.1$. (a–c) and (d–f) are for initial conditions $\phi(x, y, z, 0) = \sin(2\pi x)\sin(2\pi y)\sin(2\pi z)$ and $\phi(x, y, z, 0) = \text{rand}(x, y, z)$, respectively. (a) and (d) correspond to $\Delta t = 0.01$ h, (b) and (e) to $\Delta t = 0.1$ h, (c) and (f) to $\Delta t = 0.5$ h.

a given space, we use a set of time steps $\Delta t = 0.01$ h, $\Delta t = 0.1$ h, and $\Delta t = 0.5$ h for $h = 0.025$, $h = 0.05$, and $h = 0.1$, respectively. In this test, two initial conditions are $\phi(x, y, z, 0) = \sin(2\pi x)\sin(2\pi y)\sin(2\pi z)$ and $\phi(x, y, z, 0) = \text{rand}(x, y, z)$. All simulations are performed until $T = 0.8$, with the results presented in Fig. 5. We find that the results with $\Delta t = 0.1$ h are in good agreement with the results for the smaller time step $\Delta t = 0.01$ h. However, these differ from the results for the larger time step $\Delta t = 0.5$ h. Therefore, to maintain our proposed scheme's accuracy and reduce computational costs, an appropriate value for Δt is $0.1h_{\min}$. Here, $h_{\min}$ represents the smallest edge of the triangle. It should be noted that as shown in Section 3.1, the temporal evolution of CH equation has multiple time scales. For phase separation simulation, an initial random perturbation evolves on a fast time scale, and later coarsening evolves on a very slow time scale. Therefore, we can use a small time step to capture the fast dynamics, and then use a large time step to reduce the computational time. An adaptive time step method, which was proposed in our previous papers [63,64], can be straightforwardly applied to simulate the Cahn–Hilliard equation on surfaces, since our proposed scheme is unconditionally stable and allows the use of large time steps.

Fig. 6. Convergence of numerical results with refined temporal grids. The results, from left to right, were obtained with $\Delta t =$ 1.563E−3, 7.813E−4, 3.906E−4, 1.953E−4, and $\Delta t =$ 2.441E−5. (a) Plots of ϕ at time $T = 0.0125$. (b) Plots of the difference between that grid and the reference solution cell, i.e., $\phi - \phi^{\text{ref}}$ at time $T = 0.0125$.

Table 1
Error and convergence results with various time steps. Here $h = 0.05$ is fixed. The numerical reference solution is obtained with a very fine temporal grid $\Delta t =$ 2.441E−5 at time $T = 0.0125$.

Δt	1.563E−3		7.813E−4		3.906E−4		1.953E−4
Sphere surface: l_2-error	2.058E−4		4.862E−4		1.206E−5		2.980E−5
Sphere surface: Rate		2.08		2.01		2.02	
Bunny surface: l_2−error	1.078E−2		2.791E−3		7.074E−4		1.673E−4
Bunny surface: Rate		1.95		1.97		2.08	

3.3. Convergence test

In this test, we want to verify the convergence predicted by our proposed method. To assess the effects of the geometry and structure of the mesh grid, we employ the surface of a sphere as a test geometry, and generate a set of high-quality mesh grids using the mesh generator. Here, we refer to a mesh grid as high-quality if the edge of every triangle is almost the same. Note that our method has few requirements on the mesh quality. To demonstrate this, we perform a test in Section 3.4. To obtain the convergence rate for temporal discretization, we choose a set of decreasing time steps $\Delta t =$ 1.563E−3, 7.813E−4, 3.906E−4, and 1.953E−4 and compare the solutions at the same time T. Here, the space step size $h = 0.05$ is fixed. Because there is no closed-form analytical solution for this problem, we consider a numerical reference solution, ϕ^{ref}, which is obtained with a very fine time step $\Delta t =$ 2.441E−5. Numerical solutions are computed until time $T = 0.0125$ with initial condition $\phi(x, y, z, 0) = \sin(2\pi x)\sin(2\pi y)\sin(2\pi z)$ on a unit sphere surface and a bunny surface. In Fig. 6(a), we show the solutions on the bunny surface at time $T = 0.0125$. For better visualization, we show the difference between that grid and the reference solution cell, i.e., $\phi - \phi^{\text{ref}}$ at time $T = 0.0125$ in Fig. 6(b). As shown in Fig. 6, the convergence of the results under temporal refinements is evident.

We also define the error of a grid as the discrete l_2-norm of the difference between that grid and the reference solution cell. The rate of convergence is defined as the ratio of successive errors: $\log_2(\|e_{\Delta t}\|_2 / \|e_{\frac{\Delta t}{2}}\|_2)$. The errors and rates of convergence obtained using these definitions are presented in Table 1. Second-order accuracy with respect to time is observed, as expected from the discretization.

We define the error of a grid as the discrete l_2-norm of the difference between that grid and the average of the reference solution cells neighboring it as follows: $e_{hi} := \phi_{di} - (\zeta_i \phi_p^{\text{ref}} + \eta_i \phi_q^{\text{ref}} + \theta_i \phi_r^{\text{ref}})$. Here, p, q, and r are fine reference grid indexes in a triangle. Furthermore, ζ_i, η_i, and θ_i satisfy $\mathbf{v}_{di} = \zeta_i \mathbf{v}_p^{\text{ref}} + \eta_i \mathbf{v}_q^{\text{ref}} + \theta_i \mathbf{v}_r^{\text{ref}}$. The rate of convergence is defined as the ratio of successive errors: $\log_2(\|e_h\|_2 / \|e_{\frac{h}{2}}\|_2)$. The errors and rates of convergence obtained using these definitions are presented in Table 2. Here, all simulations are performed until $T = 0.0125$ with fixed time step $\Delta t =$ 9.766E-5. In addition, the numerical reference solution at time $T = 0.0125$ is obtained using a very fine space grid $h = 0.025$. Second-order accuracy with respect to space is observed in Table 2.

3.4. Spinodal decomposition on a poor-quality surface mesh

In general, a poor-quality mesh causes the discrete Laplace–Beltrami operator over triangular surfaces to become non-convergent. Furthermore, the non-convergent Laplace–Beltrami operator results in the unstable solution of the CH equation.

Table 2
Error and convergence results with various mesh grids. Here, $\Delta t = 9.766\text{E}{-5}$ is fixed. The numerical reference solution is obtained using a very fine space grid $h = 0.025$ at time $T = 0.0125$.

h	0.2		0.1		0.05
Sphere surface: l_2-error	6.063E−2		1.604E−2		3.767E−3
Sphere surface: Rate		1.92		2.09	
Bunny surface: l_2-error	8.553E−2		2.467E−2		6.563E−3
Bunny surface: Rate		1.79		1.91	

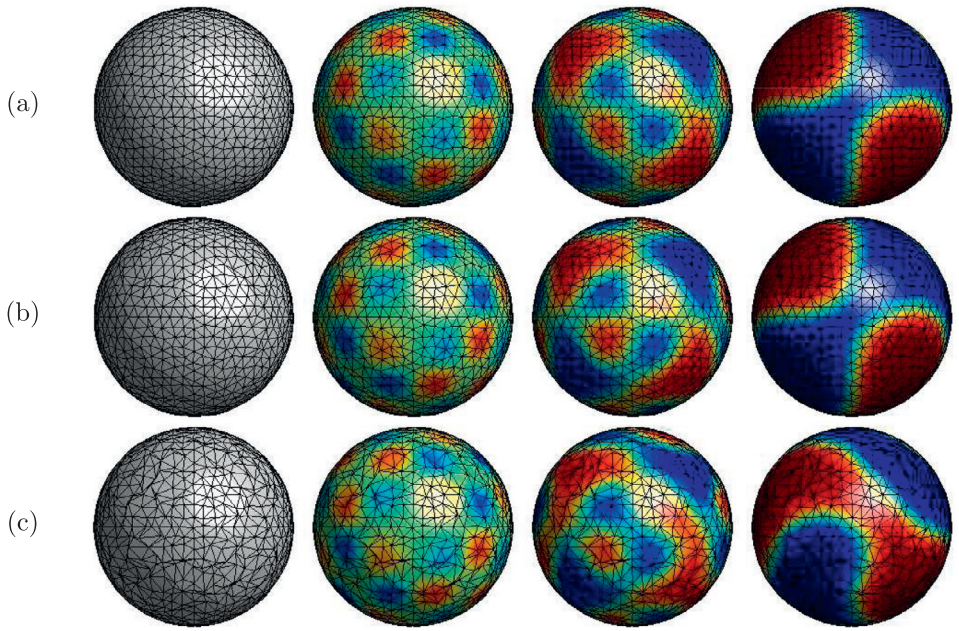

Fig. 7. Comparisons of (a) high-quality, (b) 10% poor-quality and (c) 20% poor-quality surface meshes. From left to right, these show the surface mesh structure and the plots of ϕ at time $t = 0.2$, $t = 0.4$, and $t = 1$, respectively.

To investigate the efficiency of our proposed method, we perform this simulation on high-quality, 10% poor-quality, and 20% poor-quality surface meshes, which are illustrated in Fig. 7(a–c), respectively. Here, the 10% poor-quality surface mesh is obtained by introducing 10% random noise to the high-quality mesh. The 20% poor-quality surface mesh is defined in a similar fashion. Here, $\Delta t = 0.01$ is used in Fig. 7(a) and (b). To assess the effects of the time-step size in Fig. 7(c), we have used $\Delta t = 0.001$. From left to right in Fig. 7, we illustrate the structure of the surface mesh, and the plots of ϕ at time $t = 0.2$, $t = 0.4$, and $t = 1$, respectively. The agreement between results obtained from the poor- and high-quality surface meshes over a short time is clear. These results also suggest that our proposed method can perform well on poor-quality surface meshes. Furthermore, with 10% random noise, the result using our method guarantees a mesh-independent solution. From this test, we also conclude that our approach can be applied straightforwardly to an adaptive mesh refinement framework.

3.5. Spinodal decomposition on an open surface

Spinodal decomposition is a mechanism by which a solution of two or more components separates into different phases [1]. Here, we consider this problem for the CH equation using the Neumann boundary and a plus-one Dirichlet boundary, i.e., $\phi(\mathbf{x}, t) = 1$ on the boundary of Costa's minimal surface. The initial condition is set to $\phi(x, y, z, 0) = 0.5\text{rand}(x, y, z)$. Note that to obtain a comparison in the same scaling, we normalize the length of the spatial domain of Costa's minimal surface to 2. Unless otherwise specified, we perform the same operation on the other surfaces throughout the remainder of this paper. Simulations are run until time $T = 2$, with time step $\Delta t = 0.01$. Here, we set $\epsilon = 0.05$. Fig. 8(a) and (b) illustrate the evolutions of spinodal decomposition with the Neumann and Dirichlet boundary conditions, respectively. These results suggest that our proposed method can handle an open surface and perform well with various boundary conditions. Furthermore, compared with the Neumann boundary condition, the plus-one Dirichlet boundary condition results in the phases gathering more quickly.

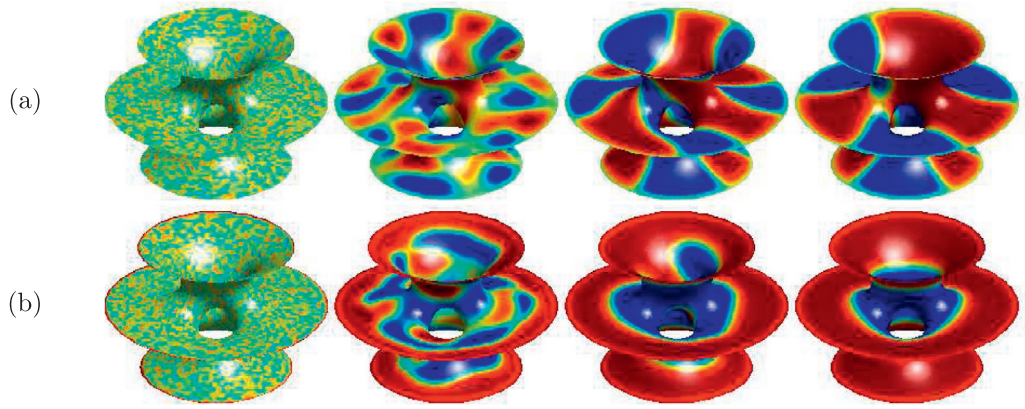

Fig. 8. Evolutions with (a) Neumann and (b) Dirichlet boundary conditions on Costa's minimal surface. From left to right, the plots show ϕ at time $t = 0$, 0.1, 0.4, and 2.

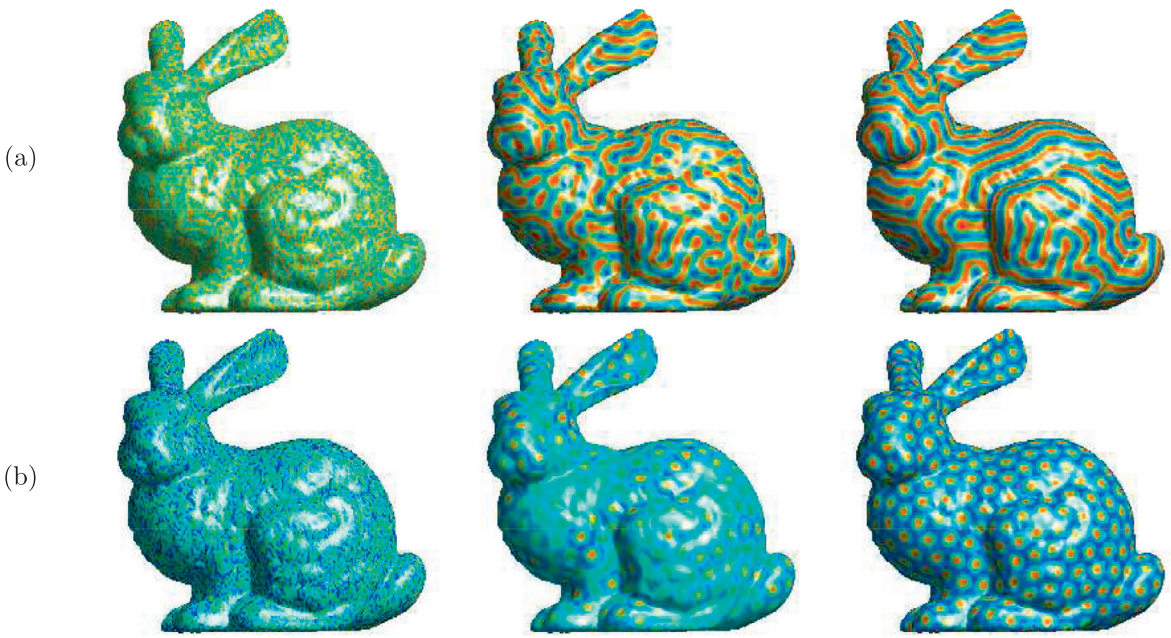

Fig. 9. Microphase separation patterns on a bunny surface with different initial conditions. (a) $\phi(x, y, z, 0) = 0.5\text{rand}(x, y, z)$ and (b) $\phi(x, y, z, 0) = -0.3 + 0.5\text{rand}(x, y, z)$. From left to right, these are the results at time $t = 0$, 0.4, and, 2.

3.6. Microphase separation patterns on surfaces

The directed self-assembly of block copolymers in thin films is an emerging technology for nanoscale patterning (see [65] and the references therein). A diblock copolymer consists of two blocks, each formed of a different type of monomer, which are chemically joined [66]. When the temperature is lowered below a critical point, the two sequences become incompatible, and the copolymer melt undergoes phase separation. This results in the occurrence of periodic structures such as lamellae, spheres, cylinders, and gyroids [67,68]. The following nonlocal Cahn–Hilliard equation has been proposed for studying microphase separation patterns in diblock copolymers on curved surfaces [15]:

$$\frac{\partial \phi}{\partial t}(\mathbf{s}, t) = \Delta_s \mu(\mathbf{s}, t) - \alpha(\phi(\mathbf{s}, t) - \bar{\phi}), \tag{32}$$

$$\mu(\mathbf{s}, t) = F'(\phi(\mathbf{s}, t)) - \epsilon^2 \Delta_s \phi(\mathbf{s}, t), \tag{33}$$

where α is inversely proportional to the square of the total chain length of the copolymer and $\bar{\phi}$ is the spatial mean value of the order parameter. Our proposed scheme can be extended to discretize Eqs. (32) and (33) as follows:

$$\frac{\phi^{n+1}-\phi^{n}}{\Delta t} = \Delta_d \mu^{n+\frac{1}{2}} - \alpha\left(\frac{\phi^{n+1}+\phi^{n}}{2} - \bar{\phi}\right), \tag{34}$$

$$\mu^{n+\frac{1}{2}} = \frac{3}{2}(F'(\phi^{n}) - \lambda\phi^{n}) - \frac{1}{2}(F'(\phi^{n-1}) - \lambda\phi^{n-1}) + \frac{\lambda}{2}(\phi^{n+1}+\phi^{n}) - \frac{\epsilon^2}{2}(\Delta_d\phi^{n+1} + \Delta_d\phi^{n}). \tag{35}$$

Fig. 9(a) and (b) present the results of microphase separation patterns on a bunny surface with the initial conditions: $\phi(x,y,z,0) = 0.5\text{rand}(x,y,z)$ and $\phi(x,y,z,0) = -0.3 + 0.5\text{rand}(x,y,z)$, respectively. From left to right, these are the results at time $t = 0,\ 0.4,$ and 2. Here, we employ the time step $\Delta t = 0.01$, $\epsilon = 0.05$, and $\alpha = 50$. Lamellar and hexagonal patterns can be observed. The results suggest that our algorithm performs well and results in high visual quality.

4. Conclusions

In this paper, we have proposed an unconditionally energy-stable second-order time-accurate scheme for the Cahn–Hilliard equation on surfaces. Discretization is performed via a surface mesh consisting of piecewise triangles and its dual-surface polygonal tessellation. The proposed scheme, which combines Crank–Nicolson and linearly stabilized splitting schemes, is second-order accurate in time. We have proven that the discrete system satisfies mass conservation and unconditional energy stability. The resulting system of discrete equations is solved using a biconjugate gradient stabilized method. We performed several numerical experiments, testing the non-increasing of discrete energy and mass conservation, the stability of the proposed scheme, time and space convergence, the relation between time and space in the proposed scheme, spinodal decomposition on a poor-quality surface mesh, spinodal decomposition on an open surface with different boundary conditions, and microphase separation patterns on surfaces. All these numerical experiments demonstrated that our proposed algorithm is efficient.

Acknowledgment

This work was funded by Natural Science Basic Research Plan in Shaanxi Province of China(2016JQ1024), by National Natural Science Foundation of China(No. 11601416). The corresponding author (J.S. Kim) was supported by Korea University Future Research Grant. The authors greatly appreciate the reviewers for their constructive comments and suggestions, which have improved the quality of this paper.

References

[1] Cahn JW, Hilliard JE. Free energy of a non-uniform system i. interfacial free energy. J Chem Phys 1958;28:258–67.
[2] Archer AJ, Evans R. Dynamical density functional theory and its application to spinodal decomposition. J Chem Phys 2004;121(9):4246–54.
[3] Marconi UMB, Tarazona P. Dynamic density functional theory of fluids. J Chem Phys 1999;110(16):8032–44.
[4] Yang S-D, Lee HG, Kim JS. A phase-field approach for minimizing the area of triply periodic surfaces with volume constraint. Comput Phys Commun 2010;181:1037–46.
[5] Kim J. Phase-field models for multi-component fluid flows. Commun Comput Phys 2012;12:613–61.
[6] Li Y, Choi J-I, Kim J. A phase-field fluid modeling and computation with interfacial profile correction term. Commun Nonlinear Sci Numer Simul 2016;30:84–100.
[7] Han D, Wang X. A second order in time, uniquely solvable, unconditionally stable numerical scheme for cahn–hilliard–navier–stokes equation. J Comput Phys 2015;290:139–56.
[8] Armstrong NJ, Painter KJ, Sherratt JA. A continuum approach to modelling cell-cell adhesion. J Theor Biol 2006;243(1):98–113.
[9] Wise S, Lowengrub J, Frieboes H, Cristini V. Three-dimensional multispecies nonlinear tumor growth: I. model and numerical method. J Theor Biol 2008;253:524–43.
[10] Bertozzi AL, Esedoglu S, Gillette A. Inpainting of binary images using the cahn–hilliard equation. IEEE Trans Image Process 2007;16:285–91.
[11] Li Y, Shin J, Choi Y, Kim JS. Three-dimensional volume reconstruction from slice data using phase-field models. Comput Vis Image Und 2015;137:115–24.
[12] Baumgart T, Hess S, Webb W. Imaging coexisting fluid domains in biomembrane models coupling curvature and line tension. Nature 2003;425:821–4.
[13] Bausch A, et al. Grain boundary scars and spherical crystallography. Science 2003;299:1716–18.
[14] Tang P, Qiu F, Zhang H, Yang Y. Phase separation patterns for diblock copolymers on spherical surfaces: a finite volume method. Phys Rev E 2005;72:016710.
[15] Jeong D, Kim JK. Microphase separation patterns in diblock copolymers on curved surfaces using a nonlocal cahn–hilliard equation. Eur Phys J E 2015;38:1–7.
[16] Sens P, Safran SA. Inclusions induced phase separation in mixed lipid film. Eur Phys J E 2000;1:237–48.
[17] Rätz A, Voigt A. PDE's on surfaces—a diffuse interface approach. Commun Math Sci 2006;4:575–90.
[18] Witkowski T, Backofena R, Voigt A. The influence of membrane bound proteins on phase separation and coarsening in cell membranes. Phys Chem Chem Phys 2012;14. 14509C14515.
[19] Elliott CM, Stinner B. Modeling and computation of two phase geometric biomembranes using surface finite elements. J Comput Phys 2010;229:6585–612.
[20] Nitschke I, Voigt A, Wensch J. A finite element approach to incompressible flow on manifolds. J Fluid Mech 2012;708:418–38.
[21] Aland S, Rätz A, Röger M, Voigt A. Buckling instability of viral capsids-a continuum approach. Multiscale Model Simul 2012;10:82–110.
[22] Osher S, Sethian J. Fronts propagating with curvature dependent speed: algorithms based on hamilton-jacobi formulations. J Comput Phys 1988;79:12–49.
[23] Bertalmio M, Cheng L-T, Osher S, Sapiro G. Variational problems and partial differential equations on implicit surfaces. J Comput Phys 2001;174:759–80.

[24] Ruuth S, Merriman B. A simple embedding method for solving partial differential equations on surfaces. J Comput Phys 2008;227(3):1943–61.
[25] Macdonald C, Brandman J, Ruuth S. Solving eigenvalue problems on curved surfaces using the closest point method. J Comput Phys 2011;230(22):7944–56.
[26] Macdonald CB, Ruuth SJ. The implicit closest point method for the numerical solution of partial differential equations on surfaces. SIAM J Sci Comput 2009;31:4330–50.
[27] März T, Macdonald CB. Calculus on surfaces with general closest point functions. SIAMJ Numer Anal 2012;50:3303–28.
[28] Fuselier EJ, Wright GB. A high-order kernel method for diffusion and reaction-diffusion equations on surfaces. J Sci Comput 2013;56:535–65.
[29] Piret C. The orthogonal gradients method: a radial basis functions method for solving partial differential equations on arbitrary surfaces. J Comput Phys 2012;231:4662–75.
[30] Meyer M, Desbrun M, Schröder P, Barr AH. Discrete differential-geometry operators for triangulated 2-manifolds. Math Vis 2002;3:52–8.
[31] Xu G. Discrete laplace-beltrami operators and their convergence, comput. Aided Geom Design 2004;21:767–84.
[32] Wu J-Y, Chen S-G, Chi M-H. A local tangential lifting differential method for triangular meshes. Math Comput Simul 2010;80:2386–402.
[33] Belkin M, Sun J, Wang Y. Discrete laplace operator on meshed surfaces. In: Proceeding SCG '08 proceedings of the twenty-fourth annual symposium on computational geometry; 2008. p. 278–87.
[34] Belkin M, Sun J, Wang Y. Constructing laplace operator from point clouds in r^d. In: Proceedings of the twentieth annual ACM-SIAM symposium on discrete algorithms; 2009. p. 1031–40.
[35] Wu J-Y, Chi M-H, Chen S-G. A new intrinsic numerical method for PDE on surfaces. Int J Comput Math 2012;89:54–79.
[36] Chen S-G, Wu J-Y. Discrete conservation laws on curved surfaces II: a dual approach. SIAM J Sci Comput 2014;36(4):1813–30.
[37] Chen S-G, Wu J-Y. Discrete conservation laws on curved surfaces. SIAM J Sci Comput 2013;36(2):719–39.
[38] Zingales M, Failla G. The finite element method for fractional non-local thermal energy transfer in non-homogeneous rigid conductors. Commun Nonlinear Sci Numer Simul 2015;29:116–27.
[39] Zhang R, Zhang Q, Song H. An efficient finite element method for pricing american multi-asset put options. Commun Nonlinear Sci Numer Simul 2015;29:25–36.
[40] Dziuk G, Elliott C. Surface finite elements for parabolic equations. J Comput Math 2007;25:385–407.
[41] Du Q, Ju L, Tian L. Finite element approximation of the cahn–hilliard equation on surfaces. Comput Methods Appl Mech Engrg 2011;200:2458–70.
[42] Chernyshenko AY, Olshanskii MA. An adaptive octree finite element method for PDEs posed on surfaces. Comput Methods Appl Mech Engrg 2015;291:146–72.
[43] Persson PO, Stangt G. A simple mesh generator in MATLAB. SIAM Rev Soc Ind Appl Math 2004;46:329–45.
[44] Du Q, Gunzburger M, Ju L. Constrained centroidal voronoi tessellations on general surfaces. SIAM J Sci Comput 2003;24:1488–506.
[45] Talischi C, Paulino GH, Pereira A, Menezes IFM. Polymesher: a general-purpose mesh generator for polygonal elements written in matlab. Struct Multidiscip Optim 2012;45(3):309–28.
[46] Sun F, Choi Y-K, Wang W, Yan D-M, Liu Y, Lévy B. Obtuse triangle suppression in anisotropic meshes. Comp Aided Geom Design 2011;28:537–48.
[47] Chen LQ. Phase-field models for microstructure evolution. Ann Rev Mater Res 2002;32:113–40.
[48] Dai S, Du Q. Motion of interfaces governed by the cahn–hilliard equation with highly disparate diusion mobility. SIAM J Appl Math 2012;72:1818–41.
[49] Furihata D. A stable and conservative finite difference scheme for the cahn–hilliard equation. Numer Math 2001;87:675–99.
[50] He Y, Liu Y, Tang T. On large time-stepping methods for the cahn–hilliard equation. Appl Numer Math 2007;57:616–28.
[51] Gomez H, Hughes TJR. Provably unconditionally stable, second-order time-accurate, mixed variational methods for phase-field models. J Comput Phys 2011;230:5310–27.
[52] Kay D, Welford R. A multigrid finite element solver for the cahn–hilliard equation. J Comput Phys 2006;212:288–304.
[53] Lee HG, Kim J. A second-order accurate non-linear difference scheme for the n-component cahn–hilliard system. Physica A 2008;387:4787–99.
[54] Li Y, Jeong D, Shin J, Kim JS. A conservative numerical method for the cahn–hilliard equation with dirichlet boundary conditions in complex domains. Comput Math Appl 2013;65:102–15.
[55] Li Y, Lee HG, Xia B, Kim J. A compact fourth-order finite difference scheme for the three-dimensional cahn–hilliard equation. Comput Phys Commun 2016;200:108–16.
[56] Li Y, Choi J-I, Kim J. Multi-component cahn–hilliard system with different boundary conditions in complex domains. J Comput Phys 2016;323:1–16.
[57] van der Vorst HA. BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetriclinear systems. SIAM J Sci Stat Comput 1992;13(2):631–44.
[58] Kim JS, Kang K, Lowengrub J. Conservative multigrid methods for ternary cahn–hilliard systems. Comm Math Sci 2004;2(1):53–77.
[59] Shen J, Yang X. Numerical approximations of allen–cahn and cahn–hilliard equations. Discrete Contin Dyn Syst 2010;28(4):1669–91.
[60] Caffarelli LA, Muler NE. An l_∞ bound for solutions of the cahn–hilliard equation. Arch Rational Mech Anal 1995;133:129–44.
[61] Condette N, Melcher C, Suli E. Spectral approximation of pattern-forming nonlinear evolution equations with double-well potentials of quadratic growth. Math Comput 2011;80(273):205–23.
[62] Gilbarg D, Trudinger NS. Elliptic partial differential equations of second order. Berlin: Springer-Verlag; 2001.
[63] Li Y, Choib Y, Kim J. Computationally efficient adaptive time step method for the cahnCHilliard equation. Comput Math Appl 2017;73(8):1855–64.
[64] Li Y, Kim J. An efficient and stable compact fourth-order finite difference scheme for the phase field crystal equation. Comput Methods Appl Mech Engrg 2017;319:194–216.
[65] Campbell IP, Lau GJ, Feaver JL, Stoykovich MP. Network connectivity and long-range continuity of lamellar morphologies in block copolymer thin films. Macromolecules 2012;45:1587–94.
[66] Singh C, Goulian M, Liu AJ, Fredrickson GH. Phase behavior of semiflexible diblock copolymers. Macromolecules 1994;27:2974–86.
[67] Liu F, Goldenfeld N. Dynamics of phase separation in block copolymer melts. Phys Rev A 1989;39:4805.
[68] Choksi R, Peletier MA, Williams JF. On the phase diagram for microphase separation of diblock copolymers: an approach via a nonlocal cahn–hilliard functional. SIAM J Appl Math 2009;69:1712–38.