

The subdivision-based IGA-EIEQ numerical scheme for the binary surfactant Cahn–Hilliard phase-field model on complex curved surfaces

Qing Pan^a, Chong Chen^b, Timon Rabczuk^c, Jin Zhang^a, Xiaofeng Yang^{d,*}

^a School of Computer and Communication Engineering, Changsha University of Science & Technology, Changsha, 410114, China

^b LSEC, ICMSEC, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, 100190, China

^c Institute of Structural Mechanics, Bauhaus Universität-Weimar, Weimar, 99423, Germany

^d Department of Mathematics, University of South Carolina, Columbia, SC 29208, USA

Received 18 November 2022; received in revised form 28 December 2022; accepted 14 January 2023

Available online xxxx

Abstract

In this paper, we consider numerical approximations of the binary surfactant phase-field model on complex surfaces. Consisting of two nonlinearly coupled Cahn–Hilliard type equations, the system is solved by a fully discrete numerical scheme with the properties of linearity, decoupling, unconditional energy stability, and second-order time accuracy. The IGA approach based on Loop subdivision is used for the spatial discretizations, where the basis functions consist of the quartic box-splines corresponding to the hierarchic subdivided surface control meshes. The time discretization is based on the so-called explicit-IEQ method, which enables one to solve a few decoupled elliptic constant-coefficient equations at each time step. We then provide a detailed proof of unconditional energy stability along with implementation details, and successfully demonstrate the advantages of this hybrid strategy by implementing various numerical experiments on complex surfaces.

© 2023 Elsevier B.V. All rights reserved.

Keywords: Loop subdivision; IGA-EIEQ; Decoupled; Unconditional energy stability; Binary fluid-surfactant

1. Introduction

Surfactants refer to organic compounds that are added to a multiphase solution (e.g., an oil–water mixture) to alter or reduce the surface tension of the solution. Therefore, when considering how to numerically simulate surfactants, the natural idea is to consider a three-phase coupled model that can describe not only the two-phase fluid interface dynamics, but also the surfactant concentration. The phase-field approach, due to its flexibility in the energetic modeling, is one of the best options for describing such scenarios. More precisely, in the framework of the phase-field method, the motion of the binary fluid can be described by one phase-field variable, and then another phase-field variable is set to represent the surfactant concentration. Then by adding some of corresponding energy potentials of the latter to the total free energy of the former, using the variation approach in some metric space, the so-called

* Corresponding author.

E-mail addresses: panqing@lsec.cc.ac.cn (Q. Pan), chench@lsec.cc.ac.cn (C. Chen), timon.rabczuk@uni-weimar.de (T. Rabczuk), jzhang@csust.edu.cn (J. Zhang), xyang@math.sc.edu (X. Yang).

<https://doi.org/10.1016/j.cma.2023.115905>

0045-7825/© 2023 Elsevier B.V. All rights reserved.

phase-field surfactant model can be derived. Surfactant models obtained using this type modeling approaches can be traced back to the seminal work of Laradji et al. about three decades ago, see [1,2], as well as some extended modeling/numerical work in [3–7], in which the main idea is also based on the above framework although some various energy functionals may be employed. Formally, the commonality of these phase-field surfactant models is a high degree of coupling (e.g., coupling between phase-field variables and/or between the fluid velocity and phase-field variables) and nonlinearity, making it rather challenging to develop efficient numerical methods for them. Moreover, we recall that some special phase separation dynamics of the multiphase flow may occur on static or dynamic surfaces, see [8–10], thus, the numerical study of the behavior of interfacial changes of the phase-field model on complex surfaces also has considerable significance and practical applications. Therefore, the binary surfactant model, as one of the important applications of the phase-field model, is also very necessary to design an effective fully-discrete numerical scheme to study its dynamical behaviors on the curved surfaces, which will be considered in this article.

It is well known that the design of a fully discretized scheme of a PDE system requires different discretization methods in two different directions, the space and the time, at the same time. Considering the spatial discretization methods on the curved surfaces, we recall that there are many successful numerical methods, such as the Spectral method [11,12], the finite difference method [13], and the finite element method (FEM) [14], etc. The first two methods have made considerable progress in recent years, however, most of the problems they deal with are still limited to simple domains, such as rectangular, circular, or spherical regions. Compared with them, the FEM can more effectively deal with various irregular surfaces that appear in the practical engineering, see [15–17] for theoretical/numerical studies of the phase transition-related models on surfaces. However, a subsequent disadvantage of discretizing surfaces using the FEM involves the approximation of surface differential geometry operators, which may cause additional errors caused by the numerical approximation schemes. Taking the Laplacian operator on a surface as an example, the derivative of the surface itself needs to be integrated, while the traditional finite element method requires the manual intervention, which is costly, time-consuming and labor-intensive, and the refining process is difficult, especially for complex geometric surfaces. Compared with these methods for dealing with surface PDEs, the so-called subdivision-based isogeometric analysis (IGA) method [18,19] has considerable advantages and is adopted in this paper.

The framework of IGA [20–25] was proposed to develop a seamless integration between FEM and computer-aided design (CAD). It has the higher numerical accuracy than FEM, and can also easily implement p -refinement, h -refinement and even k -refinement through the knot insertion and/or order elevation techniques. As a result, the accuracy of the numerical simulations can be improved without destroying the original geometry, thereby eliminating the interactive communication with the CAD system. Surface subdivision can construct smooth surfaces from arbitrary topological meshes by designing a set of simple and efficient refinement schemes [26–28]. It is not only compatible with NURBS, but also has the refinement capability of B-spline technology. Subdivision technology not only conveniently handles complicated geometric forms but also maintains original characteristics near boundaries through straightforward extensions, such as concave/convex angles and sharp/smooth creases. Both Loop subdivision [29–32] and Catmull–Clark subdivision [33–35] have been utilized in IGA. Local refinement and convergence rate [36,37] were investigated in Catmull–Clark subdivision-based IGA as well. In short, the subdivision-based IGA method can be viewed as the natural choice for higher-order FEM in engineering practice, cf. [31,32,38,39].

After we employ the efficient subdivision-based IGA method for the spatial discretization, we further consider what time-advance method to use to discretize the binary surfactant phase-field model in the time direction. The challenge of the time advancement for this particular model involves how to deal with the nonlinear and coupled terms, so that not only the energy dissipation laws of the PDE system can be extended to the discrete level, but also the high-order time accuracy and the computational efficiency can be as convenient and practical as possible. In a rather extensive numerical study of different versions of the surfactant phase-field model, we notice that there have been many successful attempts at the time discretization, for instance, the nonlinear convex-splitting approach applied in [40], the nonlinear implicit approach given in [41], the IEQ (Invariant Energy Quadratization) approach (or its various version the so-called Scalar Auxiliary Variable approach) developed in [42–46], etc. However, these known methods are either fully discrete methods based on regular regions or only semi-discrete methods in time. Note that the aim of this article is to construct an efficiently fully discretized scheme for the surfactant phase-field model on a curved surface that is both easy to implement and has the high temporal accuracy. Hence, in addition to

using the subdivision-based IGA method described above for the spatial discretization, for the time discretization we adopt the IEQ method because it can automatically generate an unconditionally stable linear algorithm. We expect the combination of these two approaches will give us a satisfactory fully discrete numerical scheme on the complex curved surface.

However, the IEQ method still has some shortcomings, especially in the practical implementation. It generally requires solving a linear system with variable coefficients at each time step, see [47–50]. It is a well-known fact that solving a variable coefficient system not only requires more complex steps, such as finding an iterative solver or constructing some preconditioner, but also requires more computational time than a linear system with only constant coefficients. Therefore, to solve this defect existing inside the original IEQ method, this paper modifies the IEQ method into a slightly newer version, the so-called explicit IEQ (referred to as EIEQ) method, in which by introducing a nonlocal variable and designing for a special but trivial ODE for it, the decoupling structure is achieved, see also [51] for its application in a different crystal growth model on the regular geometric domain. The benefit brought about by the new EIEQ method is that the scheme involved ultimately only requires the solution of some linear constant-coefficient equations with the fully decoupled structure at each time step, and is therefore highly efficient in practice. We not only rigorously demonstrate the property of our scheme to have the unconditional energy stability but also present the flexibility and the robustness of the proposed scheme by implementing several complex surface-based examples.

We organize the rest of this paper in the following way. In Section 2, we give a brief preliminary notation for surfaces. In Section 3, the binary surfactant phase-field model on the surface is presented. In Section 4, the subdivision-based IGA method is introduced and a fully discrete scheme based on its combination with the EIEQ time advancing strategy is developed. We further provide a rigorous proof of the unconditional energy stability, and the implementation method to achieve the full decoupling of all variables. In Section 5, we present various numerical examples to illustrate the efficiency and the accuracy of the proposed method. Section 6 gives some concluding remarks.

2. Preliminary: Surface differential geometry

We first briefly introduce a mathematical framework of the surface which includes the parameterization and some differential geometric operators.

We denote $\mathcal{S} = \{\mathbf{x}(u^1, u^2) \in \mathbb{R}^3 : (u^1, u^2) \in \mathcal{D} \subset \mathbb{R}^2\}$ as a sufficiently smooth and orientable surface, which is parameterized as follows. We let $g_{\alpha\beta} = \langle \mathbf{x}_{u^\alpha}, \mathbf{x}_{u^\beta} \rangle$ and $b_{\alpha\beta} = \langle \mathbf{n}, \mathbf{x}_{u^\alpha u^\beta} \rangle$ be the coefficients of the first and the second fundamental forms of $\mathcal{S}$ with

$$\begin{aligned} \mathbf{x}_{u^\alpha} &= \frac{\partial \mathbf{x}}{\partial u^\alpha}, \quad \mathbf{x}_{u^\alpha u^\beta} = \frac{\partial^2 \mathbf{x}}{\partial u^\alpha \partial u^\beta}, \quad \alpha, \beta = 1, 2, \\ \mathbf{n} &= (\mathbf{x}_{u^1} \times \mathbf{x}_{u^2}) / \|\mathbf{x}_{u^1} \times \mathbf{x}_{u^2}\|, \end{aligned}$$

where $\langle \cdot, \cdot \rangle$, $\|\cdot\|$ and $\cdot \times \cdot$ stand for the usual inner product, Euclidean norm and cross product in $\mathbb{R}^3$ respectively. Then we introduce some notations as

$$[g^{\alpha\beta}] = [g_{\alpha\beta}]^{-1}, \quad g = \det[g_{\alpha\beta}], \quad [b^{\alpha\beta}] = [b_{\alpha\beta}]^{-1}, \quad b = \det[b_{\alpha\beta}].$$

The matrix form of the Weingarten map is denoted as

$$\mathbf{S} = [b_{\alpha\beta}][g^{\alpha\beta}] = \frac{1}{g} \begin{bmatrix} b_{11}g_{22} - b_{12}g_{12} & b_{12}g_{11} - b_{11}g_{12} \\ b_{12}g_{22} - b_{22}g_{12} & b_{22}g_{11} - b_{12}g_{12} \end{bmatrix},$$

which is a self-adjoint linear map on the tangent space $T_x\mathcal{S} := \text{span}\{\mathbf{x}_{u^1}, \mathbf{x}_{u^2}\}$. Then the eigenvalues k_1 and k_2 of $\mathbf{S}$ are the *principal curvatures* of $\mathcal{S}$, whose arithmetic average and product are the *mean curvature* H and the *Gaussian curvature* K , namely,

$$H = \frac{k_1 + k_2}{2} = \frac{\text{tr}(\mathbf{S})}{2} = \frac{b_{11}g_{22} - 2b_{12}g_{12} + b_{22}g_{11}}{2g}, \quad K = k_1k_2 = \det(\mathbf{S}) = \frac{b_{11}b_{22} - b_{12}^2}{g},$$

respectively. The mean curvature normal is referred to $\mathbf{H} = H\mathbf{n}$.

Some differential geometric operators of surface need to be introduced as follows.

Tangential gradient operator. Denote $C^1(\mathcal{S})$ be a function space composing of C^1 smooth functions of $\mathcal{S}$. Considering $f \in C^1(\mathcal{S})$, the tangential gradient operator ∇_s acting on f is defined as

$$\nabla_s f = [\mathbf{x}_{u^1}, \mathbf{x}_{u^2}][g^{\alpha\beta}][f_{u^1}, f_{u^2}]^T \in \mathbb{R}^3. \quad (2.1)$$

For a vector-valued function $\mathbf{f} = [f_1, \dots, f_k]^T \in C^1(\mathcal{S})^k$, the gradient ∇_s acting on $\mathbf{f}$ is defined as

$$\nabla_s \mathbf{f} = [\nabla_s f_1, \dots, \nabla_s f_k] \in \mathbb{R}^{3 \times k}.$$

Divergence operator. Let $\mathbf{v} \in [C^1(\mathcal{S})]^3$ be a smooth vector field on $\mathcal{S}$. Then the divergence operator div_s acting on $\mathbf{v}$ is defined as

$$\text{div}_s(\mathbf{v}) = \frac{1}{\sqrt{g}} \left[\frac{\partial}{\partial u^1}, \frac{\partial}{\partial u^2} \right] [\sqrt{g}[g^{\alpha\beta}][\mathbf{x}_{u^1}, \mathbf{x}_{u^2}]^T \mathbf{v}]. \quad (2.2)$$

Laplace–Beltrami operator. Let $f \in C^2(\mathcal{S})$. Then the Laplace–Beltrami operator (LBO) Δ_s acting on f is defined as

$$\Delta_s f = \text{div}_s(\nabla_s f). \quad (2.3)$$

With the definitions of Δ_s and div_s , we derive

$$\Delta_s f = \frac{1}{g}(g_{22}f_{11} + g_{11}f_{22} - 2g_{12}f_{12}), \quad (2.4)$$

where $f_{\alpha\beta} = f_{u^{\alpha}u^{\beta}} - (\nabla_s f)^T \mathbf{x}_{u^{\alpha}u^{\beta}}$, $\alpha, \beta = 1, 2$. The second-order differential operator Δ_s relates to the mean curvature vector, i.e., $\Delta_s \mathbf{x} = 2\mathbf{H}$. Two inner products on surface need to be introduced as

$$(u, v) = \int_{\mathcal{S}} uv \, d\mathbf{x}, \quad \text{and} \quad (\nabla_s u, \nabla_s v) = \int_{\mathcal{S}} \nabla_s u \cdot \nabla_s v \, d\mathbf{x}.$$

Sobolev space $H^k(\mathcal{S})$. Assuming that $\mathcal{S}$ is a sufficiently smooth surface, for a given constant k and a function $f \in C^\infty(\mathcal{S})$, we denote $\nabla^k f$ the k th order covariant derivative of f , with the convention $\nabla^0 f = f$. Let

$$C_k(\mathcal{S}) = \left\{ f \in C^\infty(\mathcal{S}) : \int_{\mathcal{S}} |\nabla^j f|^2 d\mathbf{x} \leq \infty \text{ for } j = 0, \dots, k \right\}. \quad (2.5)$$

It is obvious that $C_k(\mathcal{S}) = C^\infty(\mathcal{S}) \subset C^k(\mathcal{S}) \subset H^k(\mathcal{S})$, where the Sobolev space $H^k(\mathcal{S})$ on the surface $\mathcal{S}$ is denoted as [Definition 2.1](#).

Definition 2.1. Let $\mathcal{S}$ be a compact surface with at least k th order smoothness. Sobolev space $H^k(\mathcal{S})$ is the completion of $C_k(\mathcal{S})$ in the sense of norm

$$\|f\|_{H^k(\mathcal{S})} := \left(\sum_{j=0}^k \int_{\mathcal{S}} |\nabla^j f|^2 d\mathbf{x} \right)^{\frac{1}{2}}. \quad (2.6)$$

3. Binary surfactant model on the surface

The framework of the binary surfactant model is to use two phase-field variables ($\phi(\mathbf{x}, t)$ and $\rho(\mathbf{x}, t)$) to simulate the dynamics of the microphase separation in the microemulsion system. The total energy functional $E(\phi, \rho)$ (cf. [1,2,4–7]) of this system is expressed as follows,

$$E(\phi, \rho) = E_1(\phi) + E_2(\phi, \rho), \quad (3.1)$$

in which, the phase-field variable $\phi(\mathbf{x}, t)$ is the labeling function to mark the two fluid components as

$$\phi(\mathbf{x}, t) = \begin{cases} -1 & \text{fluid component I,} \\ 1 & \text{fluid component II,} \end{cases} \quad (3.2)$$

and the other phase-field variable $\rho(\mathbf{x}, t)$ is used to denote the local concentration of the surfactant. To remove the discontinuity, a thin smooth transition layer of width $O(\epsilon)$ is assumed to connect the two distinct values of ϕ . In

such a framework, the interface of the binary mixture can be represented by the zero level set $\Gamma_t = \{\mathbf{x} : \phi(\mathbf{x}, t) = 0\}$ automatically.

We introduce each of the two terms in (3.1) as follows. $E_1(\phi)$ is the commonly-used energy potential in the phase-field model for the binary fluid mixture. It reads as

$$E_1(\phi) = \int_S \left(\frac{\epsilon}{2} |\nabla_s \phi|^2 + \frac{1}{\epsilon} F(\phi) \right) d\mathbf{x}, \quad (3.3)$$

where $F(\phi) = \frac{1}{4}(\phi^2 - 1)^2$ is the Ginzburg–Landau double-well potential. The first term in (3.3) represents the hydrophilic (mixing) type potential, and the second term in (3.3) is the hydrophobic (de-mixing) type potential. The equilibrium configuration of the diffusive interface is the consequence of the competition between these two types of potentials.

$E_2(\phi, \rho)$ represents the interaction between the surfactant and the binary fluid interface. It reads as

$$E_2(\phi, \rho) = \int_S \left(\frac{\alpha}{2} W(\phi, \rho) + \beta \mathcal{G}(\rho) \right) d\mathbf{x}, \quad (3.4)$$

which is the penalty term that enables the concentration of the surfactant to accumulate near the interface with a relatively high value. In the first term of (3.4), $W(\phi, \rho) = (\rho - |\nabla_s \phi|)^2$ which is a local nonlinear coupling entropy term between ϕ and ρ , and α is a positive parameter. In the second term of (3.4), $\mathcal{G}(\rho) = \rho \ln \rho + (1 - \rho) \ln(1 - \rho)$ which is the logarithmic Flory–Huggins type energy potential, and β is a positive parameter. $\mathcal{G}(\rho)$ restricts the value of ρ to be inside the domain of $(0, 1)$, and ρ will reach its upper bound if the interface is fully saturated with the surfactant.

Based on the free energy functional (3.1), the Cahn–Hilliard type surfactant phase-field system can be derived using the variational gradient flow approach in the H^{-1} space, that reads as

$$\phi_t = M_1 \Delta_s \mu, \quad (3.5)$$

$$\mu = \frac{\delta E}{\delta \phi} = -\epsilon \Delta_s \phi + \frac{1}{\epsilon} f(\phi) + \alpha W_\phi(\phi, \rho), \quad (3.6)$$

$$\rho_t = M_2 \Delta_s \omega, \quad (3.7)$$

$$\omega = \frac{\delta E}{\delta \rho} = \beta \mathcal{G}'(\rho) + \alpha W_\rho(\phi, \rho), \quad (3.8)$$

where

$$\begin{cases} f(\phi) = F'(\phi) = \phi(\phi^2 - 1), \\ W_\phi(\phi, \rho) = \frac{\delta W(\phi, \rho)}{\delta \phi} = \nabla_s \cdot \left((\rho - |\nabla_s \phi|) \frac{\nabla_s \phi}{|\nabla_s \phi|} \right), \\ W_\rho(\phi, \rho) = \frac{\delta W(\phi, \rho)}{\delta \rho} = \rho - |\nabla_s \phi|, \end{cases} \quad (3.9)$$

and M_1, M_2 are the mobility parameters. In this article we only treat closed surface domains, and remove all boundary conditions.

By taking the L^2 inner products of (3.5) with μ , of (3.6) with $-\phi_t$, of (3.7) with ω , and of (3.8) with $-\rho_t$, we obtain the energy dissipation law as follows,

$$\frac{d}{dt} E(\phi, \rho) = -M_1 \|\nabla_s \mu\|^2 - M_2 \|\nabla_s \omega\|^2 \leq 0. \quad (3.10)$$

4. IGA-EIEQ scheme on the surface

In this section, we first design a second-order discrete scheme using the EIEQ method for the time marching. Second, we introduce the recently developed IGA approach based on the Loop subdivision [31,32] for the spatial discretization. The Subdivision-based IGA strategy provides an elegant opportunity to integrate the geometric modeling with the concurrent finite element analysis for complex and flexible surface topology structures. Finally, we prove that the discrete scheme is unconditional energy stable, and present the corresponding decoupling implementation approach to obtain high computational efficiency.

Before constructing the numerical scheme, the logarithmic Flory–Huggins potential $\mathcal{G}(\rho)$ should be normalized from its original domain $(0, 1)$ to $(-\infty, \infty)$. If we do not do this, numerically we must strictly ensure that the value

of the calculated solution is within the domain of $(0, 1)$, otherwise, the calculation will easily overflow. Therefore it is the common practice to modify the logarithmic potential to an extended form defined on $(-\infty, \infty)$, which can be realized by using the following C^2 continuous, convex, and piecewisely defined function $G(\rho)$. That is, for any $\varepsilon > 0$, the term $\mathcal{G}(\rho)$ is replaced as

$$G(\rho) = \begin{cases} \rho \ln \rho + \frac{(1-\rho)^2}{2\varepsilon} + (1-\rho) \ln \varepsilon - \frac{\varepsilon}{2}, & \text{if } \rho \geq 1 - \varepsilon, \\ \rho \ln \rho + (1-\rho) \ln(1-\rho), & \text{if } \varepsilon \leq \rho \leq 1 - \varepsilon, \\ (1-\rho) \ln(1-\rho) + \frac{\rho^2}{2\varepsilon} + \rho \ln \varepsilon - \frac{\varepsilon}{2}, & \text{if } \rho \leq \varepsilon. \end{cases} \quad (4.1)$$

We can see that $G(\rho) \rightarrow \mathcal{G}(\rho)$ when $\varepsilon \rightarrow 0$. The error bound between the normalized term and the original term is controlled by ε up to a constant scaling for the surfactant phase-field system. In this paper, we simply use $G(\rho)$ to replace $\mathcal{G}(\rho)$, see also [44–46,52–54].

4.1. EIEQ method for time marching

The goal of the time marching strategy is to develop a linear, easy-to-implement, unconditionally energy stable scheme for the system (3.5)–(3.8). The basic framework of the EIEQ method includes two steps, where the first step is to transform or “quadrature” the nonlinear functional by applying a local auxiliary variable, and the second step is to design a local variable and its associated ODE so that the decoupling type calculation can be obtained. In these ways, a linear and second-order time-accurate scheme with the property of the unconditional energy stability is designed. It also has an extra advantage, that is, one only needs to solve a few independent, constant-coefficient, elliptic equations at each time step, thus achieving very efficient computation. The details read as follows.

First, we define an auxiliary variable U (local type) as

$$U(\phi, \rho) = \sqrt{N(\phi, \rho) + B}, \quad (4.2)$$

with

$$N(\phi, \rho) = \frac{1}{\varepsilon} F(\phi) + \beta G(\rho) + \frac{\alpha}{2} W(\phi, \rho) - \frac{\eta_1}{2} \phi^2 - \frac{\eta_2}{2} \rho^2, \quad (4.3)$$

where $\eta_1, \eta_2 > 0$ and $B > 0$ are three predetermined constants. Note that $N(\phi, \rho)$ is always bounded from below, since the two negative quadratic terms related to η_1 and η_2 can be always bounded by $F(\phi)$ and $G(\rho)$. That is, we can always find a constant B to make the term $N(\phi, \rho) + B$ always positive.

Using the new variable U , we rewrite the system (3.5)–(3.8) into the following equivalent form:

$$\phi_t = M_1 \Delta_s \mu, \quad (4.4)$$

$$\mu = -\varepsilon \Delta_s \phi + \eta_1 \phi + HU, \quad (4.5)$$

$$\rho_t = M_2 \Delta_s \omega, \quad (4.6)$$

$$\omega = \eta_2 \rho + RU, \quad (4.7)$$

$$U_t = \frac{1}{2} H \phi_t + \frac{1}{2} R \rho_t, \quad (4.8)$$

where

$$\begin{cases} H(\phi, \rho) = 2 \frac{\delta U(\phi, \rho)}{\delta \phi} = \frac{\frac{1}{\varepsilon} f(\phi) - \eta_1 \phi + \alpha W_\phi(\phi, \rho)}{\sqrt{N(\phi, \rho) + B}}, \\ R(\phi, \rho) = 2 \frac{\delta U(\phi, \rho)}{\delta \rho} = \frac{\beta G'(\rho) - \eta_2 \rho + \alpha W_\rho(\phi, \rho)}{\sqrt{N(\phi, \rho) + B}}. \end{cases} \quad (4.9)$$

Second, we introduce another nonlocal variable $Q(t)$ and an ODE system related to it that reads as

$$\begin{cases} Q_t = (HU, \phi_t) - (H\phi_t, U) + (RU, \rho_t) - (R\rho_t, U), \\ Q|_{t=0} = 1. \end{cases} \quad (4.10)$$

Obviously $Q_t = 0$ and $Q|_{t=0} = 1$ implies $Q(t) \equiv 1$.

Finally, by combining the two new variables U and Q , we rewrite the system (4.4)–(4.8) into the following equivalent form:

$$\phi_t = M_1 \Delta_s \mu, \quad (4.11)$$

$$\mu = -\epsilon \Delta_s \phi + \eta_1 \phi + QHU, \quad (4.12)$$

$$\rho_t = M_2 \Delta_s \omega, \quad (4.13)$$

$$\omega = \eta_2 \rho + QRU, \quad (4.14)$$

$$U_t = \frac{1}{2} QH\phi_t + \frac{1}{2} QR\rho_t, \quad (4.15)$$

$$Q_t = (HU, \phi_t) - (H\phi_t, U) + (RU, \rho_t) - (R\rho_t, U), \quad (4.16)$$

with the initial conditions that read as

$$\begin{cases} \phi|_{t=0} = \phi_0, \quad \rho|_{t=0} = \rho_0, \quad U|_{t=0} = \sqrt{N(\phi_0, \rho_0) + B}, \quad Q|_{t=0} = 1, \\ \mu|_{t=0} = -\epsilon \Delta_s \phi_0 + \frac{1}{\epsilon} f(\phi_0) + \alpha W_\phi(\phi_0, \rho_0), \quad \omega|_{t=0} = \beta G'(\rho_0) + \alpha W_\rho(\phi_0, \rho_0). \end{cases} \quad (4.17)$$

Remark 4.1. Note that the above two obtained systems after the transformation are equivalent, that is, the system after each step of the transformation is equivalent to the previous system. For example, if we integrate (4.8) over time, and apply the initial conditions given in (4.17), (4.2) will be recovered, and the system (4.4)–(4.8) changes back to the original system (3.5)–(3.8). Moreover, the equivalence between the new system (4.11)–(4.16) and the system (4.4)–(4.8) is obvious as well, that is because $Q \equiv 1$.

The new transformed system (4.11)–(4.16) also retains the law of the energy dissipation, which is described as the following Theorem 4.1.

Theorem 4.1. *The transformed equivalent system (4.11)–(4.16) holds the law of the energy dissipation as*

$$\frac{d}{dt} \tilde{E}(\phi, \rho, Q, U) = -M_1 \|\nabla_s \mu\|^2 - M_2 \|\nabla_s \omega\|^2 \leq 0, \quad (4.18)$$

where

$$\tilde{E}(\phi, \rho, Q, U) = \int_S \left(\frac{\epsilon}{2} |\nabla_s \phi|^2 + \frac{\eta_1}{2} |\phi|^2 + \frac{\eta_2}{2} |\rho|^2 + |U|^2 - B \right) dx + \frac{1}{2} |Q|^2 - \frac{1}{2}. \quad (4.19)$$

Proof. By taking the L^2 inner product of (4.11) with $-\mu$, we get

$$-(\phi_t, \mu) = M_1 \|\nabla_s \mu\|^2. \quad (4.20)$$

By taking the L^2 inner product of (4.12) with ϕ_t in L^2 space, and using the integration by parts, we obtain

$$(\mu, \phi_t) = \epsilon (\nabla_s \phi, \nabla_s \phi_t) + \eta_1 (\phi, \phi_t) + Q(HU, \phi_t). \quad (4.21)$$

By taking the L^2 inner product of (4.13) with $-\omega$, we obtain

$$-(\rho_t, \omega) = M_2 \|\nabla_s \omega\|^2. \quad (4.22)$$

By taking the L^2 inner product of (4.14) with ρ_t in L^2 space, we get

$$(\omega, \rho_t) = \eta_2 (\rho, \rho_t) + Q(RU, \rho_t). \quad (4.23)$$

By taking the L^2 inner product of (4.15) with $2U$, we obtain

$$2(U_t, U) = \frac{d}{dt} \|U\|^2 = Q(H\phi_t, U) + Q(R\rho_t, U). \quad (4.24)$$

By multiplying (4.16) with Q , we obtain

$$\frac{d}{dt} \left(\frac{1}{2} |Q|^2 \right) = Q(HU, \phi_t) - Q(H\phi_t, U) + Q(RU, \rho_t) - Q(R\rho_t, U). \quad (4.25)$$

After taking the combination of (4.20)–(4.25), we obtain the law of the energy dissipation (4.18). $\square$

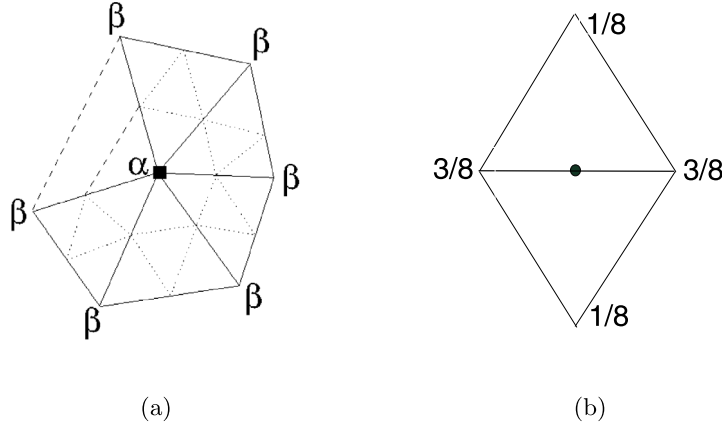


Fig. 4.1. (a) Point refinement rule with the weight $\alpha = 1 - n\beta$, where $\beta = \frac{1}{n} \left[\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} \cos \frac{2\pi}{n} \right)^2 \right]$ and n is the valence of the control vertex; and (b) Edge refinement rule.

Remark 4.2. We emphasize that the energy law (4.18) for the transformed system is exactly the same as the energy law (3.10) for the original system in the time continuous case. In the following contents, we will develop the fully discrete scheme for the new transformed system (4.11)–(4.16) according to the new energy dissipation law (4.18).

4.2. Subdivision-based IGA method

Subdivision-based IGA method can precisely represent the surface piecewisely by the limited form of the subdivision mesh with the elegant isogeometric concept. Moreover, it has the flexibility of treating complex surface models with any topology structure and the exactly expressible property for geometries which can be remained unchanged under the parameterization. The refinement of the subdivision process is equivalent to the h -refinement of NURBS. Different from traditional spline descriptions, subdivision representations do not need complex cross-patch continuity constraints and can model free-form surfaces with the structure of arbitrary topologies.

It is well known that the Loop subdivision is a simple and explicit algorithm. From the beginning of an initial coarse mesh which is called as the control mesh, the resulting smooth free-form surface can be exactly represented in a fixed and standard formation of bézier pieces of high order. We can create a refiner mesh of level $l + 1$ from level l by the subdivision scheme through

$$\mathbf{p}^{l+1} = S^l \mathbf{p}^l, \quad (4.26)$$

where P^l and P^{l+1} denote the control points of the meshes of level l and $l + 1$, and S^l is the sparse matrix representing the global subdivision operation. The entries of S^l are defined by the subdivision schemes and the topology of the mesh. The Loop subdivision schemes are divided into the rules of the old point recomputation and the edge point generation. The new position of the old control point with n valence is updated by the sum of the old position with the weight $\alpha = 1 - n\beta$ and all 1-ring surrounding control points with the weight $\beta = \frac{1}{n} \left[\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} \cos \frac{2\pi}{n} \right)^2 \right]$ (see Fig. 4.1(a)). A new point on an edge is generated as the average of the four adjacent points, where the averaging masks are given in Fig. 4.1(b). Repeated refinement leads to the hierarchical and increasingly refined surface models which finally generate the limit surface with C^1 smoothness.

Denote an initial control mesh Ω_h^0 , and we can obtain the hierarchic control meshes by use of the Loop subdivision algorithm, named as Ω_h^k , $k = 0, \dots, \infty$. A limit surface S can be achieved by an infinite subdivision procedure as $k \rightarrow \infty$. The patches of the hierarchic Loop subdivision surfaces are glued together parametrically and agree with its limit surface except around the immediate neighborhood at the extraordinary points of the control mesh; in such a neighborhood they join with the tangent continuity and interpolate the limit position of the extraordinary points of the control mesh. An explicit form of the limit position of each control point is stated as the following Lemma 4.1.

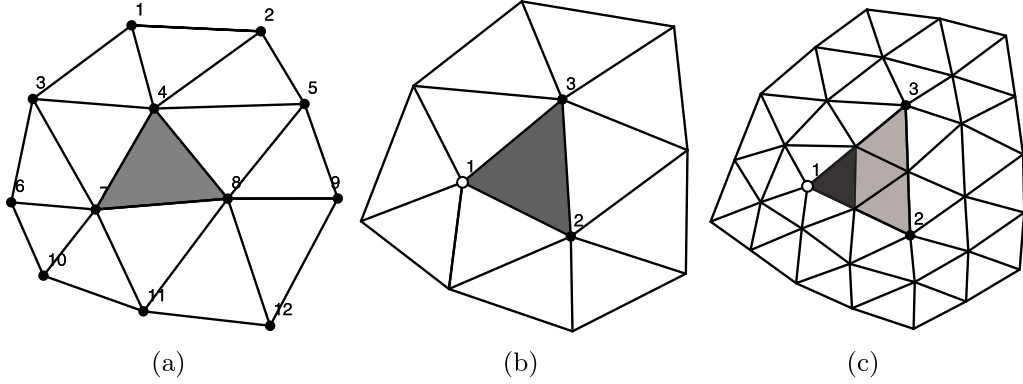


Fig. 4.2. (a): Computation of a regular Loop subdivision patch. The shaded patch is the computable Loop subdivision patch with its surrounding 12 control points, and (b): Computation of an irregular Loop subdivision patch. The control vertex denoted as “1” is an extraordinary control point. Subdividing this patch once can generate three computable sub-patches and one uncomputable sub-patch. This uncomputable patch can be repeatedly subdivided to generate more computable sub-patches.

Lemma 4.1. Let $\mathbf{x}_0^k$ be a control point of valence n on the mesh Ω_h^k , and $\mathbf{x}_j^k$, $j = 1, \dots, n$, be its 1-ring neighbor control points. All these points converge to a single position

$$\hat{\mathbf{x}}_0 = (1 - nl)\mathbf{x}_0^k + l \sum_{j=1}^n \mathbf{x}_j^k, \quad l = \frac{1}{n + 3/(8\beta)}, \quad (4.27)$$

as the subdivision step $k \rightarrow \infty$ (see [27] for the proof).

The hierarchic patches of the Loop subdivision surface are glued together C^2 almost everywhere and are at least tangent continuous near the extraordinary point of the control mesh, where the transitions between patches are almost all parametric. Any points lying within a regular patch may be computed directly by use of the box spline basis functions without any more subdivision, i.e., the regular patch is the polynomial of order 4 (quartic box-spline)

$$\mathcal{S}(u, v) = \sum_{i=1}^{12} B_i(u, v) \mathbf{x}_i^k, \quad (4.28)$$

where we use $(u, v, 1 - u - v)$ as the barycentric coordinates of the parametric patch, $\mathbf{x}_i^k$ are the corresponding 2-ring surrounding neighbor control points of the control mesh (See Fig. 4.2(a)), and B_i are the quartic box-splines. For the case that the valence of at least one control point for the patch is not six, there has no explicitly computable formula for this triangular loop surface patch with an irregular topology structure. Fortunately, we can still use the quartic box-spline form (4.28) to calculate the resulting subdivision surface patch through a fast algorithm proposed by Stam [55]. The core idea is to subdivide repeatedly this patch until the position you need to calculate is included in a quartic box-spline patch (See Fig. 4.2(b) and (c)).

The subdivision shape functions are the quartic box-splines, which possess the property of C^1 continuity (and H^2 integrability). Therefore they can be naturally used to represent the solution of high-order PDE systems governing the dynamical behaviors of complex geometries with any topological structure. The subdivision model provides the rather elegant property of representing both the geometry and the physics of the complex geometry through the same mathematical description. Back to the difficulty of obtaining the reliable and the accurate evaluation of the Loop limit surface during the process of the finite element integration, it is feasible to compute the Loop basis functions and their derivatives at arbitrary parameter positions on the triangular patch. For the case of the regular patch which means the valence of all control points of the triangular patch is six, the basis functions are the quartic box-splines. For the case of the irregular patch which means the valence of at least one point of the triangular patch is not six, the patch consists of an infinite sequence of ever-smaller nested sub-patches resulting from the Loop subdivision, most of which are underlying quartic box-spline patches, therefore we can calculate the value at any point in the region except at its extraordinary control points.

The Loop basis functions have the 2-ring of triangles surrounding the given control points as its natural support set. The evaluation of the limit surface can be performed at some fixed parameter points (Gauss points) within every limit patch. In the context of the finite element simulations, we only need to ensure that the quadratures are computed to arrive at the necessary precision. Therefore, we only need to perform very limited subdivision steps to make all of the hierarchical limit patches become computable quartic box-splines patches. To reduce the computation cost, we develop a simplified and generalized approach for adaptive discretization subdivision, whose basic idea is to replace the refinement of mesh patches with the refinement of basis functions (see [32]).

The standard way of solving the governing equations is to discretize the variables by representing them as the linear combinations of the finite element basis functions with compact supports. We demonstrate the unity of employing the Loop subdivision basis functions to represent the surface geometry and perform the finite element simulation of the phase-field movement on the surface. Loop subdivision patches can exactly represent geometries in the same way which is consistent with the concept of the isogeometric strategy. We denote the set of basis functions $\phi_1, \dots, \phi_n$ for the finite element function space $H^2(\mathcal{S}_h)$ where the control mesh $\mathcal{S}_h$ has n control points $\mathbf{x}_1, \dots, \mathbf{x}_n$. As described above, our basis functions result from the limit process based on the Loop subdivision approach. The solution surface $\mathcal{S}$ is exactly represented as the limit form through the Loop subdivision scheme,

$$\mathcal{S}(\mathbf{x}(u, v)) = \mathcal{S}(x(u, v), y(u, v), z(u, v)) = \sum_{j=1}^n \phi_j(u, v) \mathbf{x}_j. \quad (4.29)$$

This kind of basis functions also have the superior properties of the classical finite elements. First, they are nonnegative everywhere and positive around their corresponding control point because the weight coefficients of the subdivision schemes are positive. Second, because the limit value at each control point $\mathbf{x}_j$ is a linear summation of itself and its one-ring neighboring control points, the support of each basis function has two-ring neighborhoods. Finally, the weight coefficients of all subdivision schemes are summed to one, therefore we have $\sum_{j=0}^n \phi_j = 1$.

4.3. Fully discrete scheme

In this subsection, we design the fully discrete scheme for solving the transformed system (4.11)–(4.16). In what follows, denote the time step size as $\delta t > 0$ and $t^n = n\delta t$ for $0 \leq n \leq N = [T/\delta t]$. Take the test functions $\theta_h, \vartheta_h, \xi_h, \zeta_h$ and $\varsigma_h \in H^2(\mathcal{S}_h)$, which is our IGA finite element space induced by the limit form of Loop subdivision as described in Section 4.2.

Suppose $\phi_h^n, \mu_h^n, \rho_h^n, \omega_h^n, U_h^n, Q^n$ and $\phi_h^{n-1}, \mu_h^{n-1}, \rho_h^{n-1}, \omega_h^{n-1}, U_h^{n-1}, Q^{n-1}$ for $n \geq 1$ are known, we compute $\phi_h^{n+1}, \mu_h^{n+1}, \rho_h^{n+1}, \omega_h^{n+1}, U_h^{n+1}, Q^{n+1}$ by the following second-order backward difference formula:

$$\left(\frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}, \theta_h \right) = -M_1(\nabla_s \mu_h^{n+1}, \nabla_s \theta_h), \quad (4.30)$$

$$(\mu_h^{n+1}, \vartheta_h) = \epsilon(\nabla_s \phi_h^{n+1}, \nabla_s \vartheta_h) + \eta_1(\phi_h^{n+1}, \vartheta_h) + Q^{n+1}(H_h^* U_h^*, \vartheta_h), \quad (4.31)$$

$$\left(\frac{3\rho_h^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}, \xi_h \right) = -M_2(\nabla_s \omega_h^{n+1}, \nabla_s \xi_h), \quad (4.32)$$

$$(\omega_h^{n+1}, \zeta_h) = \eta_2(\rho_h^{n+1}, \zeta_h) + Q^{n+1}(R_h^* U_h^*, \zeta_h), \quad (4.33)$$

$$\left(\frac{3U_h^{n+1} - 4U_h^n + U_h^{n-1}}{2\delta t}, \varsigma_h \right) = \frac{1}{2} Q^{n+1}(H_h^* \phi_{ht}^*, \varsigma_h) + \frac{1}{2} Q^{n+1}(R_h^* \rho_{ht}^*, \varsigma_h), \quad (4.34)$$

$$\begin{aligned} \frac{3Q^{n+1} - 4Q^n + Q^{n-1}}{2\delta t} &= (H_h^* U_h^*, \frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}) - (H_h^* \phi_{ht}^*, U_h^{n+1}) \\ &\quad + (R_h^* U_h^*, \frac{3\rho_h^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}) - (R_h^* \rho_{ht}^*, U_h^{n+1}), \end{aligned} \quad (4.35)$$

where

$$\begin{cases} \phi_h^* = 2\phi_h^n - \phi_h^{n-1}, & \rho_h^* = 2\rho_h^n - \rho_h^{n-1}, & U_h^* = 2U_h^n - U_h^{n-1}, \\ H_h^* = H(\phi_h^*, \rho_h^*), & R_h^* = R(\phi_h^*, \rho_h^*), \\ \phi_{ht}^* = \frac{5\phi_h^n - 8\phi_h^{n-1} + 3\phi_h^{n-2}}{2\delta t}, & \rho_{ht}^* = \frac{5\rho_h^n - 8\rho_h^{n-1} + 3\rho_h^{n-2}}{2\delta t}. \end{cases} \quad (4.36)$$

In the above design, the discretization approach we adopted is very simple where the linear part is treated implicitly, and the nonlinear part including the nonlocal variable Q uses the combination of explicit and implicit methods, that is, in addition to the implicit processing of Q , all others are processed explicitly. The above scheme seems to be a coupling type, however, we can apply the nonlinear property of the variable Q to construct the following decoupling approach, shown in Section 4.3.1.

4.3.1. Decoupled type implementation

In this section, we introduce a nonlocal splitting method to obtain the decoupling implementation process for the scheme (4.30)–(4.35).

Step 1: we use Q^{n+1} to split ϕ_h^{n+1} , μ_h^{n+1} , ρ_h^{n+1} , ω_h^{n+1} and U_h^{n+1} into the linear combination forms that read as

$$\begin{cases} \phi_h^{n+1} = \phi_{1h}^{n+1} + Q^{n+1}\phi_{2h}^{n+1}, & \mu_h^{n+1} = \mu_{1h}^{n+1} + Q^{n+1}\mu_{2h}^{n+1}, \\ \rho_h^{n+1} = \rho_{1h}^{n+1} + Q^{n+1}\rho_{2h}^{n+1}, & \omega_h^{n+1} = \omega_{1h}^{n+1} + Q^{n+1}\omega_{2h}^{n+1}, \\ U_h^{n+1} = U_{1h}^{n+1} + Q^{n+1}U_{2h}^{n+1}. \end{cases} \quad (4.37)$$

By applying the linear form given in (4.37), we can split the scheme (4.30)–(4.34) as follows,

$$\begin{cases} \left(\frac{3\phi_{1h}^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}, \theta_h \right) = -M_1(\nabla_s \mu_{1h}^{n+1}, \nabla_s \theta_h), \\ (\mu_{1h}^{n+1}, \vartheta_h) = \epsilon(\nabla_s \phi_{1h}^{n+1}, \nabla_s \vartheta_h) + \eta_1(\phi_{1h}^{n+1}, \vartheta_h), \\ \left(\frac{3\rho_{1h}^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}, \xi_h \right) = -M_2(\nabla_s \omega_{1h}^{n+1}, \nabla_s \xi_h), \\ (\omega_{1h}^{n+1}, \zeta_h) = \eta_2(\rho_{1h}^{n+1}, \zeta_h), \end{cases} \quad (4.38)$$

$$\begin{cases} \left(\frac{3\phi_{2h}^{n+1}}{2\delta t}, \theta_h \right) = -M_1(\nabla_s \mu_{2h}^{n+1}, \nabla_s \theta_h), \\ (\mu_{2h}^{n+1}, \vartheta_h) = \epsilon(\nabla_s \phi_{2h}^{n+1}, \nabla_s \vartheta_h) + \eta_1(\phi_{2h}^{n+1}, \vartheta_h) + (H_h^* U_h^*, \vartheta_h), \\ \left(\frac{3\rho_{2h}^{n+1}}{2\delta t}, \xi_h \right) = -M_2(\nabla_s \omega_{2h}^{n+1}, \nabla_s \xi_h), \\ (\omega_{2h}^{n+1}, \zeta_h) = \eta_2(\rho_{2h}^{n+1}, \zeta_h) + (R_h^* U_h^*, \zeta_h), \end{cases} \quad (4.39)$$

$$\left(\frac{3U_{1h}^{n+1}}{2\delta t}, s_h \right) = \left(\frac{4U_h^n - U_h^{n-1}}{2\delta t}, s_h \right), \quad (4.40)$$

$$\left(\frac{3U_{2h}^{n+1}}{2\delta t}, s_h \right) = \frac{1}{2} (H_h^* \phi_{ht}^*, s_h) + \frac{1}{2} (R_h^* \rho_{ht}^*, s_h). \quad (4.41)$$

The system (4.38) and (4.39) is easy to be solved because there are linear elliptic with constant coefficients. (4.40) and (4.41) are also very easy to be solved.

Step 2: By applying the values ϕ_h^{n+1} , ρ_h^{n+1} and U_h^{n+1} obtained from (4.38)–(4.41), we update Q^{n+1} in (4.35) through

$$\left(\frac{3}{2\delta t} - \gamma_2 \right) Q^{n+1} = \frac{1}{2\delta t} (4Q^n - Q^{n-1}) + \gamma_1, \quad (4.42)$$

where γ_1 and γ_2 are given as

$$\begin{cases} \gamma_1 = (H_h^* U_h^*, \frac{3\phi_{1h}^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}) - (H_h^* \phi_{ht}^*, U_{1h}^{n+1}) + (R_h^* U_h^*, \frac{3\rho_{1h}^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}) - (R_h^* \rho_{ht}^*, U_{1h}^{n+1}), \\ \gamma_2 = (H_h^* U_h^*, \frac{3\phi_{2h}^{n+1}}{2\delta t}) - (H_h^* \phi_{ht}^*, U_{2h}^{n+1}) + (R_h^* U_h^*, \frac{3\rho_{2h}^{n+1}}{2\delta t}) - (R_h^* \rho_{ht}^*, U_{2h}^{n+1}). \end{cases} \quad (4.43)$$

We need prove the solvability of (4.42) by showing $\frac{3}{2\delta t} - \gamma_2 \neq 0$. First, by taking $\theta_h = \mu_{2h}^{n+1}$, $\vartheta_h = \frac{3}{2\delta t} \phi_{2h}^{n+1}$ and $\xi_h = w_{2h}^{n+1}$, $\zeta_h = \frac{3}{2\delta t} \rho_{2h}^{n+1}$ in (4.39), we get

$$\begin{aligned} M_1 \|\nabla_s \mu_{2h}^{n+1}\|^2 + \frac{3\epsilon}{2\delta t} \|\nabla_s \phi_{2h}^{n+1}\|^2 + \frac{3\eta_1}{2\delta t} \|\phi_{2h}^{n+1}\|^2 + M_2 \|\nabla_s w_{2h}^{n+1}\|^2 + \frac{3\eta_2}{2\delta t} \|\rho_{2h}^{n+1}\|^2 \\ = -(H_h^* U_h^*, \frac{3\phi_{2h}^{n+1}}{2\delta t}) - (R_h^* U_h^*, \frac{3\rho_{2h}^{n+1}}{2\delta t}). \end{aligned} \quad (4.44)$$

Then we set $\varsigma_h = 2U_{2h}^{n+1}$ in (4.41) to achieve

$$\left\| \frac{3U_{2h}^{n+1}}{\delta t} \right\|^2 = (H_h^* \phi_{ht}^*, U_{2h}^{n+1}) + (R_h^* \rho_{ht}^*, U_{2h}^{n+1}). \quad (4.45)$$

Thus, we deduce $-\gamma_2 \geq 0$ by combining (4.44) and (4.45), which implies (4.42) is always solvable.

It can be seen that the computation of the above scheme is completely decoupled, and the nonlinear term does not introduce any unnecessary iterations since one just needs to solve some elliptic equations with constant coefficients at each time step.

4.3.2. Energy stability

We show the unconditional energy stability of the discrete scheme (4.30)–(4.35) in the following Theorem 4.2.

Theorem 4.2. *The second-order discrete scheme (4.30)–(4.35) is unconditionally energy stable, i.e., satisfies the following discrete energy dissipation law:*

$$\frac{\tilde{E}^{n+1} - \tilde{E}^n}{\delta t} \leq -M_1 \|\nabla_s \mu_h^{n+1}\|^2 - M_2 \|\nabla_s \omega_h^{n+1}\|^2, \quad (4.46)$$

where, for an integer $k \geq 0$, the discrete energy $\tilde{E}^k$ is defined as

$$\begin{aligned} \tilde{E}^k &= \frac{\epsilon}{4} (\|\nabla_s \phi_h^k\|^2 + \|2\nabla_s \phi_h^k - \nabla_s \phi_h^{k-1}\|^2) + \frac{\eta_1}{4} (\|\phi_h^k\|^2 + \|2\phi_h^k - \phi_h^{k-1}\|^2) \\ &+ \frac{\eta_2}{4} (\|\rho_h^k\|^2 + \|2\rho_h^k - \rho_h^{k-1}\|^2) + \frac{1}{2} (\|U_h^k\|^2 + \|2U_h^k - U_h^{k-1}\|^2) \\ &+ \frac{1}{4} (|Q^k|^2 + |2Q^k - Q^{k-1}|^2) - \frac{1}{2}. \end{aligned} \quad (4.47)$$

Proof. By taking $\theta_h = -\mu_h^{n+1}$ in (4.30), we get

$$-\left(\frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}, \mu_h^{n+1}\right) = M_1 \|\nabla_s \mu_h^{n+1}\|^2. \quad (4.48)$$

By taking $\vartheta_h = \frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}$ in (4.31) and then performing the integration by parts, we derive

$$\begin{aligned} \left(\mu_h^{n+1}, \frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}\right) &= \frac{\epsilon}{4\delta t} (\|\nabla_s \phi_h^{n+1}\|^2 - \|\nabla_s \phi_h^n\|^2 + \|2\nabla_s \phi_h^{n+1} - \nabla_s \phi_h^n\|^2 \\ &- \|2\nabla_s \phi_h^n - \nabla_s \phi_h^{n-1}\|^2 + \|\nabla_s \phi_h^{n+1} - 2\nabla_s \phi_h^n + \nabla_s \phi_h^{n-1}\|^2) \\ &+ \frac{\eta_1}{4\delta t} (\|\phi_h^{n+1}\|^2 - \|\phi_h^n\|^2 + \|2\phi_h^{n+1} - \phi_h^n\|^2 \\ &- \|2\phi_h^n - \phi_h^{n-1}\|^2 + \|\phi_h^{n+1} - 2\phi_h^n + \phi_h^{n-1}\|^2) \\ &+ Q^{n+1} (H_h^* U_h^*, \frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}), \end{aligned} \quad (4.49)$$

where the following identity is used

$$2(3a - 4b + c, a) = a^2 - b^2 + (2a - b)^2 - (2b - c)^2 + (a - 2b + c)^2. \quad (4.50)$$

By taking $\xi_h = -\omega_h^{n+1}$ in (4.32), we get

$$-\left(\frac{3\rho_h^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}, \omega_h^{n+1}\right) = M_2 \|\nabla_s \omega_h^{n+1}\|^2. \quad (4.51)$$

By taking $\zeta_h = \frac{3\rho_h^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}$ in (4.33) and using the identity (4.50), we achieve

$$\begin{aligned} (\omega_h^{n+1}, \frac{3\rho_h^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}) &= \frac{\eta_2}{4\delta t} (\|\rho_h^{n+1}\|^2 - \|\rho_h^n\|^2 + \|2\rho_h^{n+1} - \rho_h^n\|^2 \\ &\quad - \|2\rho_h^n - \rho_h^{n-1}\|^2 + \|\rho_h^{n+1} - 2\rho_h^n + \rho_h^{n-1}\|^2) \\ &\quad + Q^{n+1}(R_h^* U_h^*, \frac{3\rho_h^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}). \end{aligned} \quad (4.52)$$

By taking $\varsigma_h = 2U_h^{n+1}$ in (4.34) and using (4.50), we have

$$\begin{aligned} \frac{1}{2\delta t} (\|U_h^{n+1}\|^2 - \|U_h^n\|^2 + \|2U_h^{n+1} - U_h^n\|^2 - \|2U_h^n - U_h^{n-1}\|^2 \\ + \|U_h^{n+1} - 2U_h^n + U_h^{n-1}\|^2) &= Q^{n+1}(H_h^* \phi_{ht}^*, U_h^{n+1}) + Q^{n+1}(R_h^* \rho_{ht}^*, U_h^{n+1}). \end{aligned} \quad (4.53)$$

By multiplying (4.35) with Q^{n+1} and using (4.50), we obtain

$$\begin{aligned} \frac{1}{4\delta t} (|Q^{n+1}|^2 - |Q^n|^2 + |2Q^{n+1} - Q^n|^2 - |2Q^n - Q^{n-1}|^2 + |Q^{n+1} - 2Q^n + Q^{n-1}|^2) \\ = (H_h^* U_h^*, \frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}) - (H_h^* \phi_{ht}^*, U_h^{n+1}) \\ + (R_h^* U_h^*, \frac{3\rho_h^{n+1} - 4\rho_h^n + \rho_h^{n-1}}{2\delta t}) - (R_h^* \rho_{ht}^*, U_h^{n+1}). \end{aligned} \quad (4.54)$$

By combining (4.48)–(4.54), we get

$$\begin{aligned} \frac{\epsilon}{4\delta t} (\|\nabla_s \phi_h^{n+1}\|^2 - \|\nabla_s \phi_h^n\|^2 + \|2\nabla_s \phi_h^{n+1} - \nabla_s \phi_h^n\|^2 - \|2\nabla_s \phi_h^n - \nabla_s \phi_h^{n-1}\|^2 \\ + \|\nabla_s \phi_h^{n+1} - 2\nabla_s \phi_h^n + \nabla_s \phi_h^{n-1}\|^2) \\ + \frac{\eta_1}{4\delta t} (\|\phi_h^{n+1}\|^2 - \|\phi_h^n\|^2 + \|2\phi_h^{n+1} - \phi_h^n\|^2 - \|2\phi_h^n - \phi_h^{n-1}\|^2 + \|\phi_h^{n+1} - 2\phi_h^n + \phi_h^{n-1}\|^2) \\ + \frac{\eta_2}{4\delta t} (\|\rho_h^{n+1}\|^2 - \|\rho_h^n\|^2 + \|2\rho_h^{n+1} - \rho_h^n\|^2 - \|2\rho_h^n - \rho_h^{n-1}\|^2 + \|\rho_h^{n+1} - 2\rho_h^n + \rho_h^{n-1}\|^2) \\ + \frac{1}{2\delta t} (\|U_h^{n+1}\|^2 - \|U_h^n\|^2 + \|2U_h^{n+1} - U_h^n\|^2 - \|2U_h^n - U_h^{n-1}\|^2 + \|U_h^{n+1} - 2U_h^n + U_h^{n-1}\|^2) \\ + \frac{1}{4\delta t} (|Q^{n+1}|^2 - |Q^n|^2 + |2Q^{n+1} - Q^n|^2 - |2Q^n - Q^{n-1}|^2 + |Q^{n+1} - 2Q^n + Q^{n-1}|^2) \\ = -M_1 \|\nabla_s \mu_h^{n+1}\|^2 - M_2 \|\nabla_s \omega_h^{n+1}\|^2. \end{aligned} \quad (4.55)$$

Finally, we obtain the desired result (4.46) after dropping some positive terms of (4.55). $\square$

5. Numerical examples

In this section, we carry out some numerical simulations to validate the accuracy, the efficiency, and the energy stability of the proposed scheme. The surface of the performed numerical experiments is generated by the limit process of the proposed Loop subdivision. Similar to the classical finite element method, the Gaussian integral is calculated for each patch of the triangular discretization of the limit surface. The linear system of the fully discretized scheme for the equation is highly sparse, and we need a robust iterative method to solve them. In this paper, we use the GMRES solver, and we set the tolerance small enough to obtain the proper convergence of the proposed scheme.

5.1. Convergence test

We first verify the convergence rate of the developed numerical scheme. The computational domain is set to be a spherical surface with radius 1, i.e.,

$$S = \{(x, y, z) : \sqrt{x^2 + y^2 + z^2} = 1\}. \quad (5.1)$$

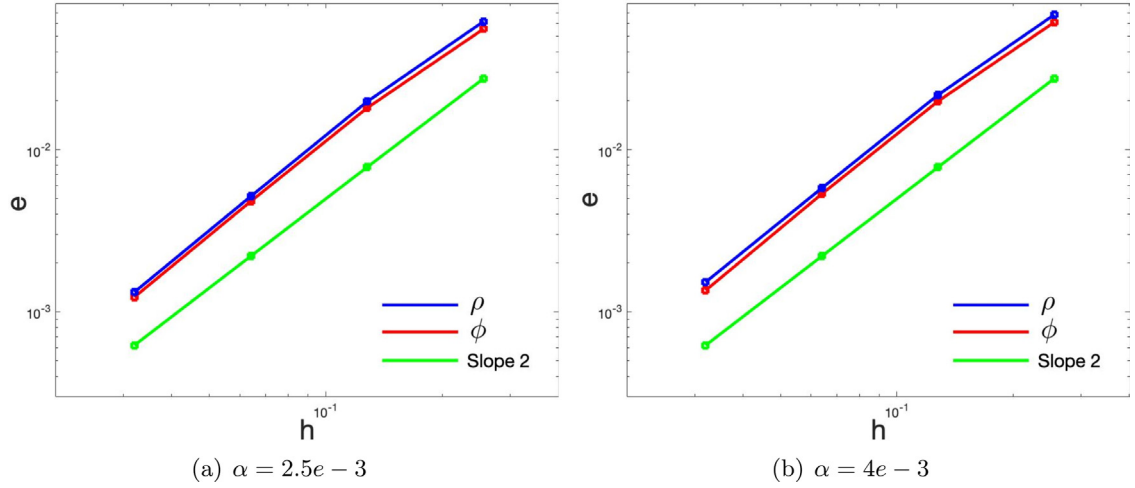


Fig. 5.1. Convergence rate of the surfactant system on the sphere. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

We set the model parameters as

$$\begin{cases} M_1 = M_2 = 2.5e - 3, \quad \varepsilon = 1e - 4, \quad \epsilon = 1.2e - 2, \\ \beta = 1e - 1, \quad \eta_1 = 4, \quad \eta_2 = 4, \quad B = 1e5, \end{cases} \quad (5.2)$$

and the initial condition as

$$\begin{cases} \phi_0 = 0.1 \cos(2x) + 0.2 \cos(y), \\ \rho_0 = 0.4 \sin(2x) + 0.3 \cos(y). \end{cases} \quad (5.3)$$

To obtain the convergence rate, a series of uniform meshes $\mathcal{S}_{h_i}$ from coarse to refined are performed, where the values of the vertex valence are in $[4, 6]$, and h_i is the mesh size. Since the subdivision-based IGA method is also second-order accurate for the spatial grid size (cf. [31]), we set the time step size $\delta t_i = Ch_i$ for the i th level surface mesh $\mathcal{S}_{h_i}$, where C is a given constant between 0 and 1. Since the exact solutions are not known, we choose the solutions obtained with a very fine mesh size computed as the benchmark solutions, which are treated approximately as the exact solutions to get the numerical errors. The L^2 errors of the two variables ϕ and ρ are plotted in Fig. 5.1 when $t = 2.56 \times 10^{-1}$, where the total numbers of patches/points for these surface models are 512/258, 2048/1026, 8192/4098, 32768/16386, and the corresponding mesh sizes $h_i = 0.2543, 0.1169, 0.0622, 0.0326$, respectively. In Fig. 5.1, we set the model parameter $\alpha = 2.5e - 3$ and $4e - 3$. We observe that the scheme is second-order accurate for space.

5.2. Spinodal decomposition

In this section, we carry out the phase separation (spinodal decomposition) simulations for the surfactant model on three different surface models. We set the initial conditions as

$$\begin{cases} \phi_0 = \bar{\phi}_0 + 0.001 \text{rand}(x, y, z), \\ \rho_0 = 0.2 + 0.001 \text{rand}(x, y, z), \end{cases} \quad (5.4)$$

where the term $\text{rand}(x, y, z)$ is the random number in $[-1, 1]$ which has zero mean, and $\bar{\phi}_0$ will be adjusted.

The first surface domain is set as a closed sphere with a unit radius, which is defined as

$$\mathcal{S}_1 = \{(x, y, z) : x^2 + y^2 + z^2 = 1\} \quad (5.5)$$

with 131 072 Loop limit subdivision patches by use of 65 538 control points. The span of the vertex valence is 4 to 6. We carry out two simulations with different initial values of $\bar{\phi}_0 = 0.0$ and $\bar{\phi}_0 = 0.3$, respectively. The model

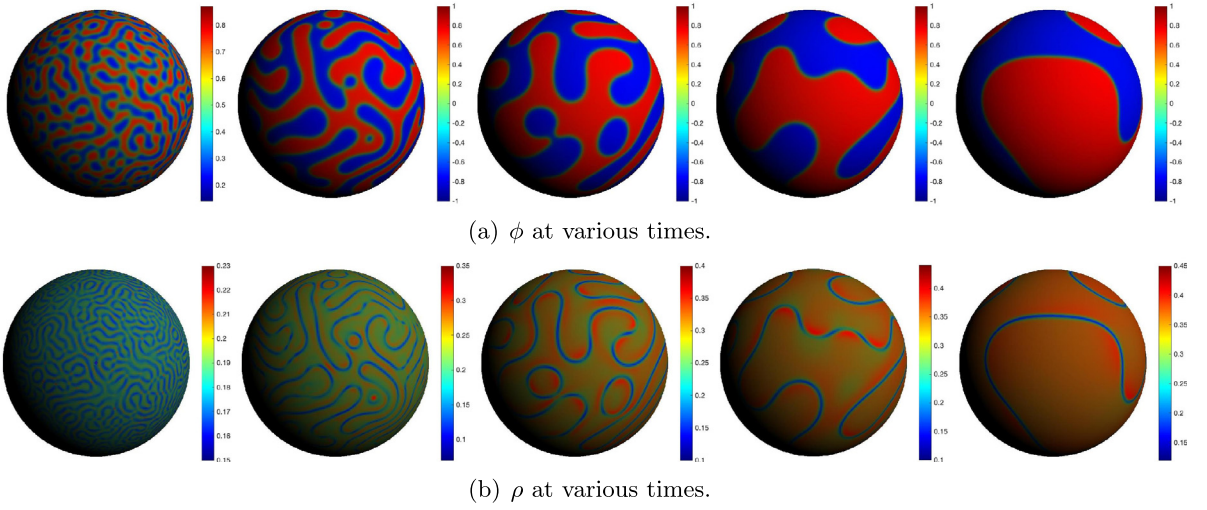


Fig. 5.2. Snapshots of the phase-field variables ϕ and ρ on the spherical surface S_1 for the example of the initial value $\bar{\phi}_0 = 0$, where (a) is ϕ and (b) is ρ , respectively. For each panel, snapshots are taken at $t = 0.5, 1.5, 4, 16, 55$ from left to right.

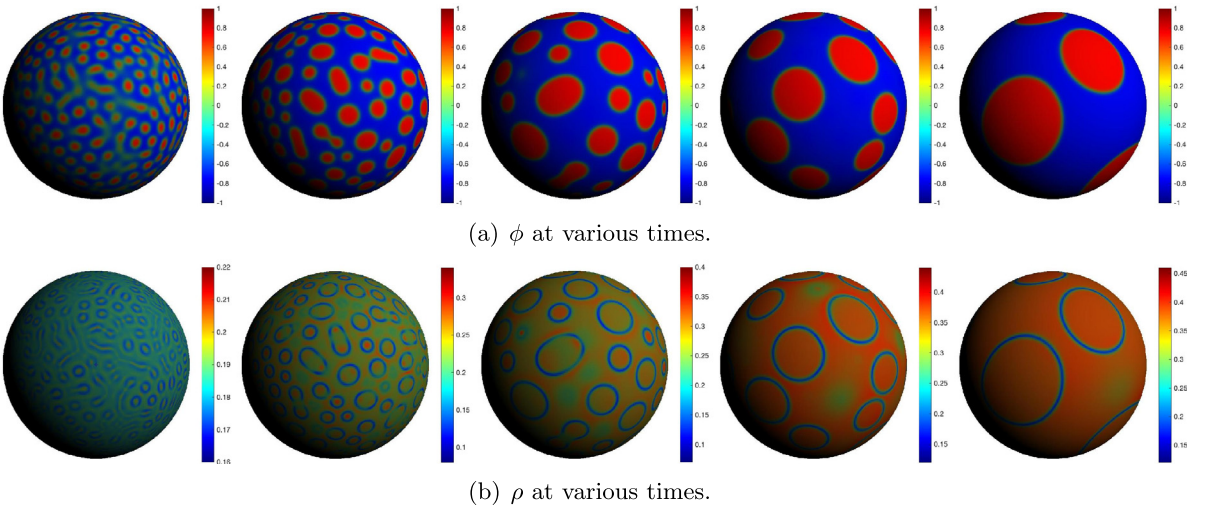


Fig. 5.3. Snapshots of the phase-field variables ϕ and ρ on the spherical surface S_1 for the example of the initial value $\bar{\phi}_0 = 0.3$, where (a) is ϕ and (b) is ρ , respectively. For each panel, snapshots are taken at $t = 0.5, 1.5, 4, 16, 55$ from left to right.

parameters are

$$\begin{cases} M_1 = M_2 = 2.5e - 3, & \varepsilon = 1e - 4, & \epsilon = 1.2e - 2, \\ \alpha = 2.5e - 3, & \beta = 1e - 1, & \eta_1 = 4, & \eta_2 = 4, & B = 1e5, \end{cases} \quad (5.6)$$

and $\delta t = 1 \times 10^{-3}$. In Fig. 5.2, we show the snapshots of the coarsening dynamics with $\bar{\phi}_0 = 0.0$ that implies the volume of the two fluid components is the same. Initially, the two fluid components are uniformly mixed, and over time, the two phases begin to separate, eventually forming two distinctly separated regions. From the evolution of ρ over time, we can observe that the higher concentration of the surfactant is located at the interface of the two fluids. Similarly, the dynamic evolution process of ϕ and ρ for $\bar{\phi}_0 = 0.3$ is presented in Fig. 5.3. Since the volume of the

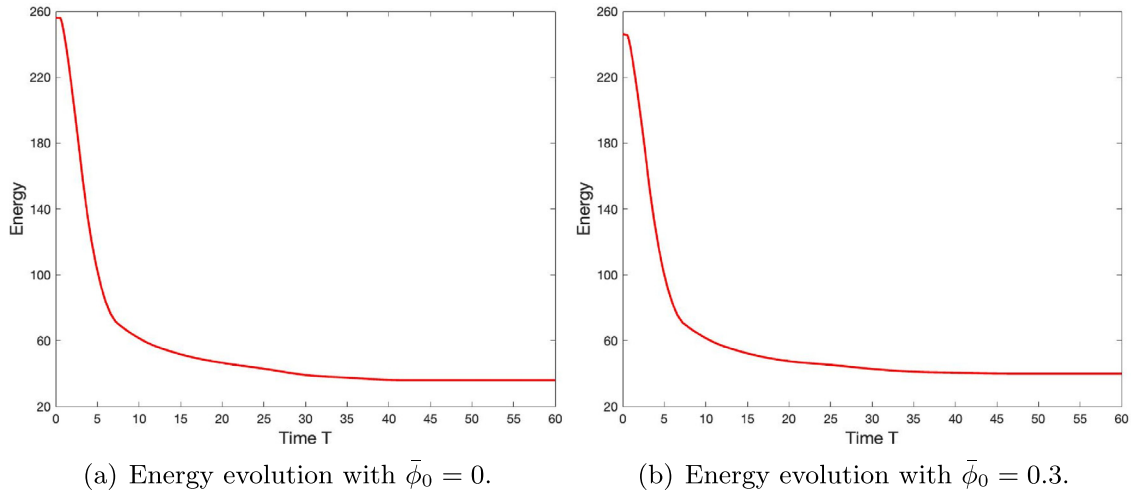


Fig. 5.4. Time evolution of the total free energy on the spherical surface $\mathcal{S}_1$ given in (5.5) with (a) $\bar{\phi}_0 = 0.0$ and (b) $\bar{\phi}_0 = 0.3$.

two fluid components is significantly different, it can be seen that the fluid with the smaller volume will aggregate into smaller droplets and eventually combine into larger droplets. For the above two simulations, we plot the time evolution of the free energy in Fig. 5.4, and we can see that the energy in both cases decays monotonically with time.

The second numerical example is a ring within the domain

$$\mathcal{S}_2 = \{(x, y, z) : x \in [-0.54, 0.65], y \in [-0.61, 0.61], z \in [-0.25, 0.25]\}. \quad (5.7)$$

The surface $\mathcal{S}_2$ is discretized by 102 400 Loop limit subdivision patches with 51 200 control points, and the span of the vertex valence is 4 to 8. The parameters are given as

$$\begin{cases} M_1 = M_2 = 3 \times 10^{-3}, \quad \varepsilon = 1 \times 10^{-4}, \quad \epsilon = 1 \times 10^{-2}, \\ \alpha = 3 \times 10^{-3}, \quad \beta = 1.5 \times 10^{-1}, \quad \eta_1 = 4, \quad \eta_2 = 4, \quad B = 9 \times 10^3. \end{cases} \quad (5.8)$$

We use the time step size $\delta t = 5 \times 10^{-4}$ to perform the simulations. The snapshots of the dynamical behaviors with the two values of the initial conditions $\bar{\phi}_0 = 0.0$ and $\bar{\phi}_0 = 0.3$ are shown in Figs. 5.5 and 5.6 respectively. We can get their final equilibrium state at around $t = 65$. For the evolution of the values ϕ and ρ , we observe similar phenomena to the first sphere example $\mathcal{S}_1$.

We perform the third simulation for a complex bunny surface model within the domain

$$\mathcal{S}_3 = \{(x, y, z) : x \in [-9.12, 6.27], y \in [-3.38, 18.72], z \in [-6.49, 6.08]\}, \quad (5.9)$$

which is discretized with 144 046 Loop limit subdivision patches by use of 72 047 control points, and the span of the vertex valence is [3,10]. We choose the time step size $\delta t = 5 \times 10^{-3}$ and the parameters are set as

$$\begin{cases} M_1 = M_2 = 3 \times 10^{-3}, \quad \varepsilon = 1 \times 10^{-4}, \quad \epsilon = 1 \times 10^{-2}, \\ \alpha = 3 \times 10^{-3}, \quad \beta = 1.5 \times 10^{-1}, \quad \eta_1 = 4, \quad \eta_2 = 4, \quad B = 1 \times 10^4. \end{cases} \quad (5.10)$$

Snapshots of the profile for the phase-field variable ϕ and the concentration variable ρ under the two values of the initial conditions $\bar{\phi}_0 = 0.0$ and $\bar{\phi}_0 = 0.3$ are shown in Figs. 5.7 and 5.8 respectively, where the final equilibrium is achieved at around $t = 75$.

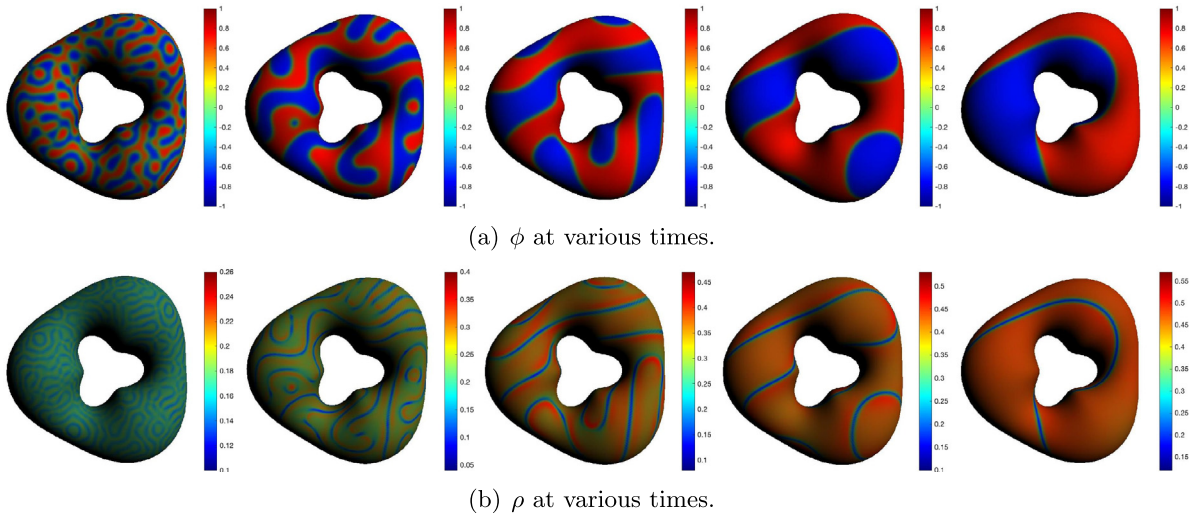


Fig. 5.5. Snapshots of the phase-field variables ϕ and ρ on the ring surface S_2 for the example of the initial value $\bar{\phi}_0 = 0$, where (a) is ϕ and (b) is ρ , respectively. For each panel, snapshots are taken at $t = 0.5, 1.5, 4, 16, 65$ from left to right.

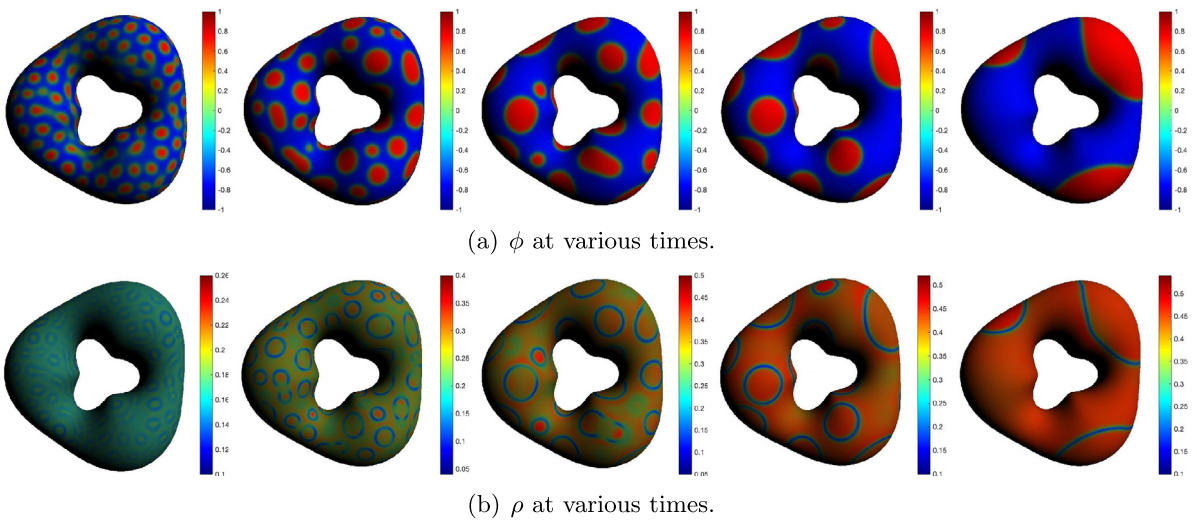


Fig. 5.6. Snapshots of the phase-field variables ϕ and ρ on the ring surface S_2 for the example of the initial value $\bar{\phi}_0 = 0.3$, where (a) is ϕ and (b) is ρ , respectively. For each panel, snapshots are taken at $t = 0.5, 1.5, 4, 16, 65$ from left to right.

6. Conclusions

In this paper, we construct a fully discrete strategy for the binary fluid-surfactant system on complex surfaces. The spatial discretization adopts the recently developed subdivision-based IGA method, where the advantages of the subdivision-based IGA method include the flexibility for complex surfaces with arbitrary topology structure and exactly expressible property for geometries which can be remained unchanged throughout the h -refinement process. The new EIEQ method for time discretization enables us to obtain a fully decoupled and energy stable linear scheme. The novel stability technique possesses the high efficiency through transforming the original nonlinear system into

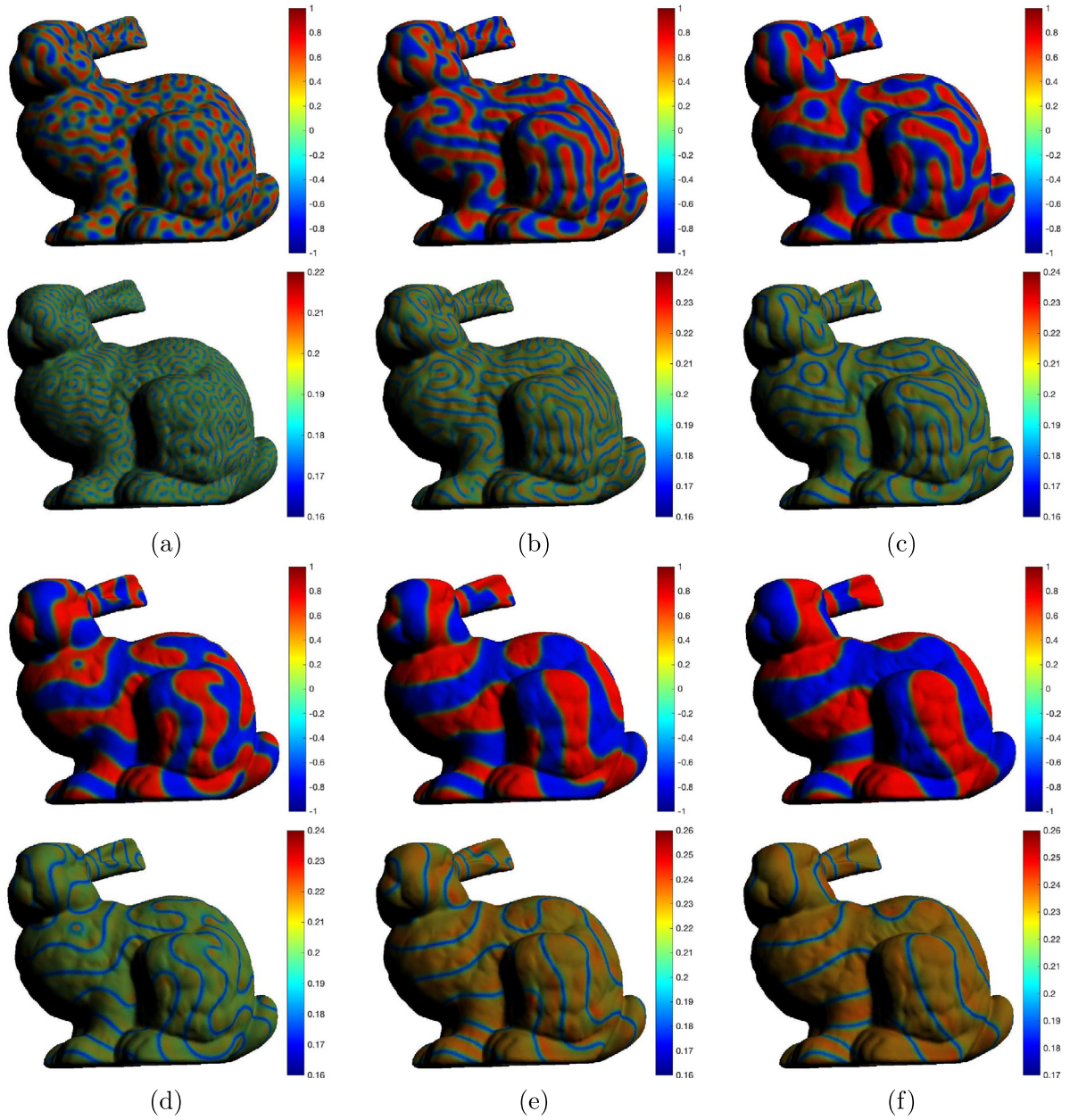


Fig. 5.7. Snapshots of the phase-field variables ϕ and ρ on the bunny surface S_3 for the example of the initial value $\bar{\phi}_0 = 0.0$. Subfigure (a) to (f) corresponds to the computation time at $t = 0.5, 2, 4, 18, 45, 75$ respectively, where the upper row is ϕ and the down row is ρ .

only some elliptic equations with constant-coefficient to be solved at each time step. We also prove the unconditional energy stability and carry out various numerical examples to demonstrate the energy stability and the accuracy of the developed numerical scheme. Moreover, it is remarkable that the developed spatiotemporal hybrid algorithm can be extended to various gradient flow problems with multiple variables and/or complex nonlinearities on complex surfaces.

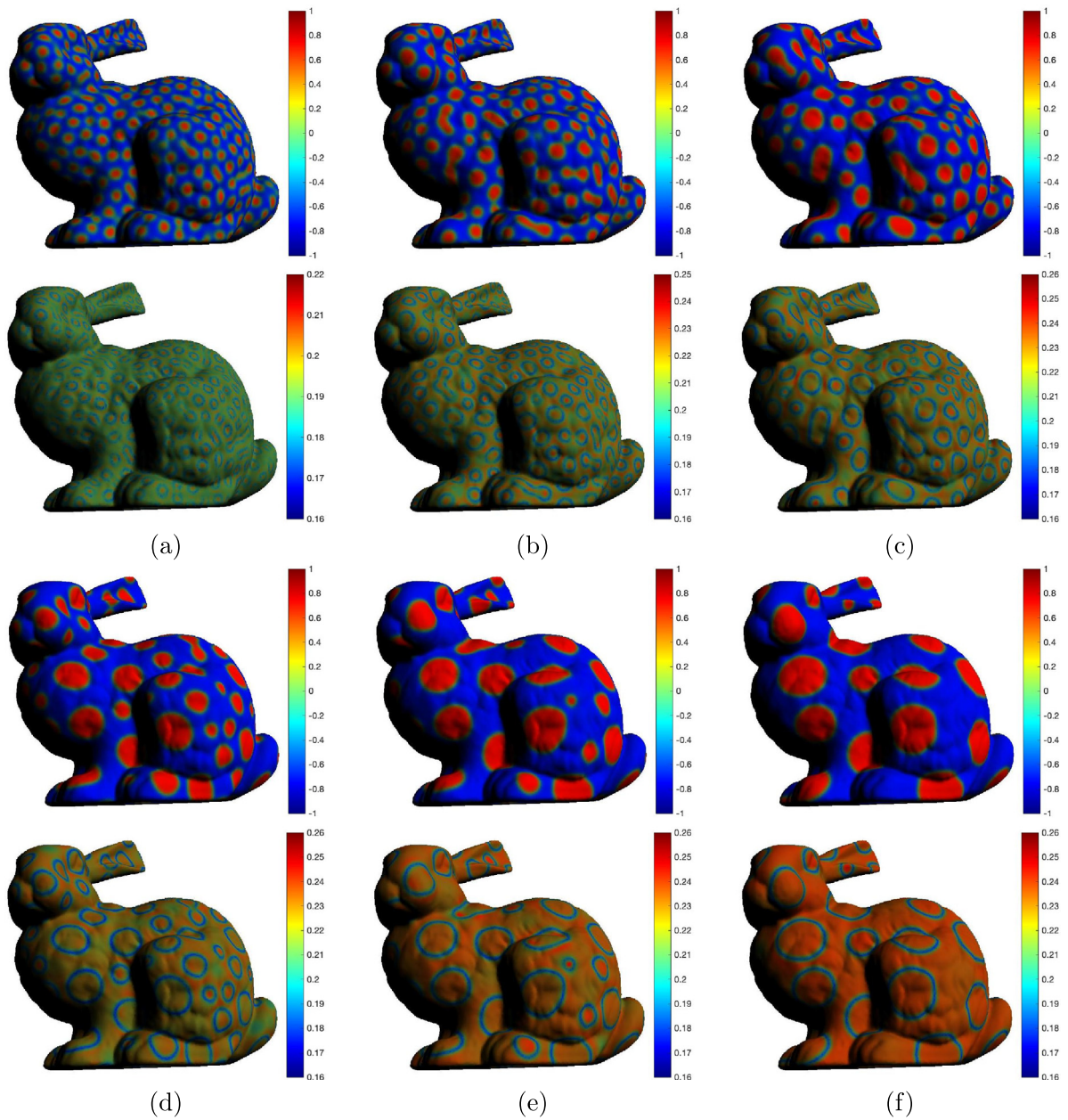


Fig. 5.8. Snapshots of the phase-field variables ϕ and ρ on the bunny surface S_3 for the example of the initial value $\bar{\phi}_0 = 0.3$. Subfigure (a) to (f) corresponds to the computation time at $t = 0.5, 2, 4, 18, 45, 75$ respectively, where the upper row is ϕ and the down row is ρ .

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Xiaofeng Yang reports financial support was provided by National Science Foundation.

Data availability

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

Acknowledgments

The work of Q. Pan was partially supported by National Natural Science Foundation of China with grant number 12171147. The work of C. Chen was partially funded by the key research project of Beijing Natural Science Foundation under Grant Z180002. The work of X. Yang was partially supported by National Science Foundation of USA with grant number DMS-2012490.

References

- [1] M. Laradji, H. Guo, M. Grant, M.J. Zuckermann, The effect of surfactants on the dynamics of phase separation, *J. Phys.: Condens. Matter* 4 (32) (1992) 6715–6728.
- [2] M. Laradji, O.G. Mouristen, S. Toxvaerd, M.J. Zuckermann, Molecular dynamics simulations of phase separation in the presence of surfactants, *Phys. Rev. E* 50 (1994) 1243–1252.
- [3] J. Bueno, H. Gomez, Liquid-vapor transformations with surfactants Phase-Field Model and Isogeometric Analysis, *J. Comput. Phys.* 321 (part A) (2016) 797–818.
- [4] R. Sman, S. Graaf, Diffuse interface model of surfactant adsorption onto flat and droplet interfaces, *Rheol. Acta* 46 (2006) 3–11.
- [5] S. Komura, H. Kodama, Two-order-parameter model for an oil-water-surfactant system, *Phys. Rev. E* 55 (1997) 1722–1727.
- [6] I. Fonseca, M. Morini, V. Slastikov, Surfactants in foam stability: A phase-field model, *Arch. Ration. Mech. Anal.* 183 (2007) 411–456.
- [7] C.H. Teng, I.L. Chern, M.C. Lai, Simulating binary fluid-surfactant dynamics by a phase-field model, *Discrete Contin. Dyn. Syst. Ser. B* 17 (2010) 1289–1307.
- [8] F.A. Heberle, G.W. Feigenson, Phase separation in lipid membranes, *Cold Spring Harbor Perspect. Biol.* 3 (4) (2011) a004630.
- [9] K. Yokota, T. Ogino, Phase separation in lipid bilayer membranes induced by intermixing at a boundary of two phases with different components, *Chem. Phys. Lipids* 191 (2015) 147–152.
- [10] R. Parthasarathy, C. Yu, J.T. Groves, Curvature-modulated phase separation in lipid bilayer membranes, *Langmuir* 22 (11) (2006) 5095–5099.
- [11] H.G. Lee, J.S. Lowengrub, J. Goodman, Modeling pinchoff and reconnection in a Hele-Shaw cell II. Analysis and Simulation in the Nonlinear Regime, *Phys. Fluids* 14 (2002) 514–545.
- [12] C. Liu, J. Shen, A phase-field model for the mixture of two incompressible fluids and its approximation by a Fourier-spectral method, *Physica D* 179 (2003) 211–228.
- [13] D. Furihata, A stable and conservative finite difference scheme for the Cahn-Hilliard equation, *Numer. Math.* 87 (2001) 675–699.
- [14] G. Dziuk, C.M. Elliot, Finite element methods for surface PDEs, *Acta Numer.* 22 (2013) 289–396.
- [15] S. Zhao, X. Xiao, X. Feng, An efficient time adaptively based on chemical potential for surface Cahn-Hilliard equation using finite element approximation, *Appl. Math. Comput.* 369 (2020) 124901.
- [16] Q. Du, L. Ju, L. Tian, Finite element approximation for the Cahn-Hilliard equation on surface, *Comput. Methods Appl. Mech. Engrg.* 200 (2011) 2458–2470.
- [17] D. Marenduzzo, E. Orlandini, Phase separation dynamics on curved surfaces, *Soft Matter* 9 (2013) 1178–1187.
- [18] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD. and elements, finite and NURBS, and geometry, exact. and mesh refinement, *Comput. Methods Appl. Mech. Engrg.* 194 (2005) 4135–4195.
- [19] Y. Bazilevs, L. Beirão da Veiga, J.A. Cottrell, T.J.R. Hughes, G. Sangalli, Isogeometric analysis: approximation stability and error estimates for H-refined meshes, *Math. Models Methods Appl. Sci.* 16 (7) (2006) 1031–1090.
- [20] L.A. Piegl, W. Tiller, The NURBS Book, second ed., in: (Monographs in Visual Communication, Springer-Verlag, New York, 1997.
- [21] T.W. Sederberg, G.T. Finnigan, X. Li, H. Lin, H. Ipson, Watertight trimmed NURBS, *ACM Trans. Graphics* 27 (3) (2008) 1–8.
- [22] Y. Zhang, Y. Bazilevs, S. Goswami, C. Bajaj, T.J.R. Hughes, Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow, *Comput. Methods Appl. Mech. Engrg.* 196 (29–30) (2007) 2943–2959.
- [23] T.W. Sederberg, J. Zheng, A. Bakenov, A. Nasri, T-splines and T-NURCCs, *ACM Trans. Graph.* 22 (3) (2003) 477–484.
- [24] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, T.W. Sederberg, Isogeometric analysis using T-splines, *Comput. Methods Appl. Mech. Engrg.* 5–8 (2010) 229–263.
- [25] K.D. Nguyen, C.L. Thanh, H. Nguyen-Xuan, M. Abdel-Wahab, A hybrid phase-field isogeometric analysis to crack propagation in porous functionally graded structures, *Eng. Comput.* 37 (4) (2021) 3017–3065.
- [26] E. Catmull, J. Clark, Recursively generated B-spline surfaces on arbitrary topological meshes, *Comput. Aided Des.* 10 (6) (1978) 350–355.
- [27] C. Loop, Smooth Subdivision Surfaces Based on Triangles (Master's thesis), Department of Mathematics, University of Utah, 1978.
- [28] J. Stam, Fast evaluation of Catmull-Clark subdivision surfaces at arbitrary parameter values, in: *SIGGRAPH '98 Proceedings*, 1998, pp. 395–404.
- [29] Q. Pan, G. Xu, G. Xu, Y. Zhang, Isogeometric analysis based on extended Loop's subdivision, *J. Comput. Phys.* 299 (15) (2015) 731–746.
- [30] Q. Pan, C. Chen, G. Xu, Isogeometric finite element approximation of minimal surfaces based on extended Loop subdivision, *J. Comput. Phys.* 343 (2017) 324–339.
- [31] Q. Pan, T. Rabczuk, G. Xu, C. Chen, Isogeometric analysis for surface PDEs with extended Loop subdivision, *J. Comput. Phys.* 398 (2019) 108892.
- [32] Q. Pan, T. Rabczuk, C. Chen, Subdivision based isogeometric analysis for geometric flows, *Internat. J. Numer. Methods Engrg.* 123 (2022) 610–633.

- [33] Q. Pan, G. Xu, G. Xu, Y. Zhang, Isogeometric analysis based on extended Catmull–Clark subdivision, *Comput. Math. Appl.* 71 (2016) 105–119.
- [34] Q. Pan, T. Rabczuk, G. Xu, C. Chen, Isogeometric analysis of minimal surfaces on the basis of extended Catmull–Clark subdivisions, *Comput. Methods Appl. Mech. Engrg.* 337 (2018) 128–149.
- [35] Q. Pan, T. Rabczuk, X. Yang, Subdivision-based isogeometric analysis for second order partial differential equations on surfaces, *Comput. Mech.* 68 (2021) 1205–1221.
- [36] X. Wei, Y. Zhang, T.J.R. Hughes, M.A. Scott, Truncated hierarchical Catmull–Clark subdivision with local refinement, *Comput. Methods Appl. Mech. Engrg.* 291 (2015) 1–20.
- [37] X. Wei, X. Li, Y.J. Zhang, T.J.R. Hughes, Tuned hybrid Non-Uniform subdivision surfaces with optimal convergence rates, *Int. J. Numer. Methods Eng.* 122 (9) (2021) 2117–2144.
- [38] F. Cirak, M. Ortiz, P. Schröder, Subdivision surfaces: a new paradigm for thin-shell finite-element analysis, *Internat. J. Numer. Methods Eng.* 47 (2000) 2039–2072.
- [39] F. Cirak, M.J. Scott, E.K. Antonsson, M. Ortiz, Schröder. P., Integrated modeling, finite-element analysis, and engineering design for thin-shell structures using subdivision, *Comput. Aided Des.* 34 (2) (2002) 137–148.
- [40] S. Gu, H. Zhang, Z. Zhang, An energy-stable finite-difference scheme for the binary fluid-surfactant system, *J. Comput. Phys.* 270 (2014) 3–11.
- [41] J. Kim, Phase-field models for multi-component fluid flows, *Commun. Comput. Phys.* 12 (2012) 613–661.
- [42] X. Yang, L. Ju, Linear and unconditionally energy stable schemes for the binary Fluid-Surfactant phase-field model, *Comput. Methods Appl. Mech. Engrg.* 318 (2017) 1005–1029.
- [43] X. Yang, Numerical approximations for the Cahn-Hilliard phase-field model of the binary fluid-surfactant system, *J. Sci. Comput.* 74 (2018) 1533–1553.
- [44] J. Zhang, C. Chen, J. Wang, X. Yang, Efficient, second order accurate, and unconditionally energy stable numerical scheme for a new hydrodynamics coupled binary phase-field surfactant system, *Comput. Phys. Comm.* 251 (2020) 107–122.
- [45] X. Yang, A novel fully-decoupled, second-order and energy stable numerical scheme of the conserved Allen-Cahn type flow-coupled binary surfactant model, *Comput. Methods Appl. Mech. Engrg.* 373 (2021) 113502.
- [46] X. Yang, Fully-discrete, decoupled, second-order time-accurate and energy stable finite element numerical scheme of the Cahn-Hilliard binary surfactant model confined in the Hele-Shaw cell, *ESAIM Math. Model. Numer. Anal.* 56 (2022) 651–678.
- [47] X. Yang, Linear, first and second-order, unconditionally energy stable numerical schemes for the phase field model of homopolymer blends, *J. Comput. Phys.* 327 (2016) 294–316.
- [48] X. Yang, H. Yu, Efficient second order unconditionally stable schemes for a phase-field moving contact line model using an Invariant Energy Quadraticization approach, *SIAM J. Sci. Comput.* 40 (2018) 889–914.
- [49] X. Yang, J. Zhao, Q. Wang, J. Shen, Numerical for a three components Cahn–Hilliard phase-field model based on the Invariant Energy Quadraticization method, *Math. Models Methods Appl. Sci.* 27 (11) (2017) 1993–2030.
- [50] J. Zhao, X. Yang, Y. Gong, Q. Wang, A Novel linear second order unconditionally energy stable scheme for a hydrodynamic Q-Tensor model of liquid crystals, *Comput. Methods Appl. Mech. Engrg.* 318 (2017) 803–825.
- [51] X. Yang, On a novel full decoupling, linear, second-order accurate, and unconditionally energy stable numerical scheme for the anisotropic phase-field dendritic crystal growth model, *Internat. J. Numer. Methods Engrg.* 122 (2021) 4129–4153.
- [52] X. Yang, On a novel fully-decoupled, second-order accurate energy stable numerical scheme for a binary fluid-surfactant phase-field model, *SIAM J. Sci. Comput.* 43 (2021) 479–507.
- [53] M.I.M. Copetti, C.M. Elliott, Numerical analysis of the Cahn-Hilliard equation with a logarithmic free energy, *Numer. Math.* 63 (4) (1992) 39–65.
- [54] C.M. Elliott, H. Garcke, On the cahn-hilliard equation with degenerate mobility, *SIAM J. Math. Anal.* 27 (1996) 404–423.
- [55] J. Stam, Fast evaluation of loop triangular subdivision surfaces at arbitrary parameter values, in: SIGGRAPH '98, 1998, CD-ROM supplement.