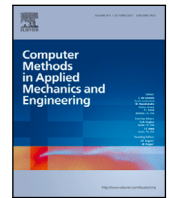




Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

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

The subdivision-based IGA-EIEQ numerical scheme for the Navier–Stokes equations coupled with Cahn–Hilliard phase-field model of two-phase incompressible flow on complex curved surfaces

Qing Pan^{a,b}, Yunqing Huang^a, Timon Rabczuk^c, Xiaofeng Yang^{d,*}

^a National Center for Applied Mathematics in Hunan, Xiangtan University, 411105, China

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

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

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

ARTICLE INFO

Keywords:

IGA-EIEQ

Fully-decoupled

Unconditional energy stability

Cahn–Hilliard

Navier–Stokes

ABSTRACT

We develop an accurate and robust numerical scheme for solving the incompressible hydrodynamically coupled Cahn–Hilliard system of the two-phase fluid flow system on complex surfaces. Our algorithm leverages a number of efficient techniques, including the subdivision-based isogeometric analysis (IGA) method for spatial discretization, the explicit Invariant Energy Quadratization (EIEQ) method for linearizing nonlinear potentials, the Zero-Energy-Contribution (ZEC) method for decoupling, and the projection method for the Navier–Stokes equation to facilitate fully decoupled type implementations. The integration of these methodologies results in a fully discrete scheme with desired properties such as linearity, second-order temporal accuracy, full decoupling, and unconditional energy stability. The implementation of the scheme is straightforward, requiring the solution of a few elliptic equations with constant coefficients at each time step. The rigorous stability proof of unconditional energy stability and the implementation procedure are given in detail. Numerous numerical simulations on complex curved surfaces are carried out to verify the effectiveness of the proposed numerical scheme.

1. Introduction

In this study, we investigate numerical approximations of the hydrodynamically coupled Cahn–Hilliard phase-field model [1], pertaining to the incompressible two-phase fluid flow system, on complex curved surfaces. The Cahn–Hilliard equation, recognized as the fundamental model within the phase-field framework, is generally derived through an energy variational approach applied to the total free energy, encompassing both linear and nonlinear potentials. We note that extensive research has been dedicated to the theoretical analysis and computational techniques associated with phase-field equations, along with their adaptable applications in various systems, such as coupling with hydrodynamics to form multiphase flow systems, coupling with electromagnetic equations to form multiphase magnetic fluids or ferromagnetic fluids, and coupling with heat equations to form dendritic crystal growth systems, etc. However, it is also evident that the advancement of numerical methods and simulations for phase-field-related systems has predominantly focused on traditional rectangular or box-shaped regions, with comparatively limited exploration in the context of regions featuring complex closed manifolds or curved surfaces.

* Corresponding author.

E-mail addresses: panqing@lsec.cc.ac.cn (Q. Pan), huangyq@xtu.edu.cn (Y. Huang), timon.rabczuk@uni-weimar.de (T. Rabczuk), xfyang@math.sc.edu (X. Yang).

<https://doi.org/10.1016/j.cma.2024.116901>

Received 19 December 2023; Received in revised form 2 February 2024; Accepted 1 March 2024

0045-7825/© 2024 Elsevier B.V. All rights reserved.

It is worth emphasizing, however, that numerical simulations of hydrodynamic systems within surface geometries are important for understanding the mechanical properties of key physical systems, including, but not limited to, soap bubble membranes [2–5], lipid bilayers [6–10], and interface-embedded colloidal systems [11–15], etc. Numerical simulations of hydrodynamical systems on curved surfaces have been relatively scarce thus far. The majority of efforts have focused on the development of spatial operator discretization methods, rather than the pursuit of achieving a numerical algorithm characterized by high spatio-temporal accuracy and unconditional energy stability. As a result, in the case of multiphase fluid flow systems coupled by the Cahn–Hilliard and hydrodynamic equations on curved surfaces – systems that are notably more complex and nonlinear – there is a significant deficiency in research efforts dedicated to developing numerical schemes and conducting computational simulations. It is important to note that the model studied in this paper serves as the foundation and source for numerous phase-field type models. Therefore, investigating the performance of this model on curved surfaces holds substantial guiding significance. In fact, we have identified only one recently introduced numerical scheme [16], specifically designed for a hydrodynamically coupled Cahn–Hilliard system on a curved surface. However, it is imperative to underscore that the numerical scheme detailed in that study suffers from a lack of energy stability and its inherent nonlinear nature, potentially incurring substantial computational expenses.

Therefore, the goal of this article is to propose an efficient numerical algorithm to solve the hydrodynamically coupled Cahn–Hilliard phase-field model, and then carry out the numerical simulation of drop dynamics under the gravity force, on the complex curved surfaces. For this purpose, we consider the discretization in the spatial and temporal directions separately. Recognizing the non-trivial nature of tackling computational challenges associated with surfaces featuring complex topologies, our objective is to devise an efficient and accurate approximation method for discretizing these surfaces, thereby effectively reducing geometric errors inherent in parametric approximations of such complexity.

The methodology of isogeometric analysis (IGA) has been witnessed to integrate the geometric simulation and the finite element solution [17,18]. It employs the same spline-type basis functions, such as non-uniform rational B-splines (NURBS) or B-splines, etc., for both the geometric representation of computer-aided design (CAD) models and the solution functions of the finite element method (FEM). This uniform representation brings us some superiority over the classical finite element basis functions. The geometric exactness can be maintained because it removes the geometric errors caused by mesh generation. It possesses high-order smoothness for basis functions, providing higher accuracy per degree of freedom through its basic knot inserting and/or order elevating method. This allows us to enhance the accuracy, efficiency, and quality of the solution method in computer-aided engineering (CAE). Subdivision is a mature technique for generating smooth surfaces/volumes in computer graphics. It can support more flexible geometry with any topology. It is well-known that subdivision surfaces can generate smooth surfaces with arbitrary topological control meshes using the quartering refinement technique [19–21]. Additionally, subdivision surfaces possess B-spline refinement capabilities, making them a superior choice for finite element analysis with higher-order properties in computer-aided engineering (CAE) [22–26].

When considering the time discretization for the system, one of the widely recognized numerical difficulties lies in the inherent stiffness of the Cahn–Hilliard equation, which is introduced by the nonlinear double-well potential incorporating the interfacial width parameter. The choice between fully implicit and explicit discretizations constitutes a well-acknowledged dilemma. A simple fully implicit scheme entails significant computational costs and potential instability, often necessitating certain time-step constraints [27]. On the other hand, a fully explicit scheme offers convenience in implementation but tends to result in computational instability [28]. Hence, a common strategy for discretizing nonlinear terms is to combine explicit and implicit methods, effectively mitigating or removing the time step's stability constraint. Prominent approaches in this context encompass nonlinear convex splitting methods [29–31], linear Invariant Energy Quadratization (IEQ) methods [32–35] or its variant version of Scalar Auxiliary Variable (SAV) method [36], etc. Frequently, when developing linear numerical discrete solution methods for the Cahn–Hilliard equation with a focus on achieving unconditional energy stability, the choice of the latter two methods often provides a convenient linear numerical approach. Importantly, they also ensure at least second-order time accuracy. Furthermore, as the Cahn–Hilliard equations are extended to integrate with the hydrodynamic system, giving rise to a coupled multiphase fluid flow system, more intricate numerical challenges come into play. These challenges include addressing the discretization of the nonlinear coupling terms arising from the integration of velocity and phase field variables to ensure the preservation of linearity, stability, and second-order accuracy. To tackle this numerical challenge, we can employ the so-called ZEC decoupling method which is quite effective in handling the nonlinear couplings, see [37–40]. This method entails introducing a nonlocal auxiliary variable and devising a new ordinary differential equation (ODE) for it. The ODE is used to transform the original system into an equivalent one that is more amenable to discretization, making it “algorithm-friendly”. In this way, the coupling terms can be simply discretized using the explicit type methods to achieve the desired numerical scheme.

Therefore, through the integration of the explicit-IEQ (EIEQ) method [32,33,41–43] for linearizing the nonlinear potential, the ZEC method [37–40] for managing coupled nonlinear terms to achieve decoupled implementations, the projection method [44] for separating the computation of the velocity field from the pressure, and the IGA method for the spatial discretization, we establish a fully-discrete numerical scheme tailored for solving the hydrodynamically coupled Cahn–Hilliard system on complex curved surfaces. The numerical scheme is highly efficient, as it only necessitates solving a small number of decoupled elliptic equations with constant coefficients at each time step. We also provide a rigorous demonstration of the scheme's unconditional energy stability and solvability, and carry out several compelling numerical examples of drop dynamics on complex curved surfaces to verify the robustness and accuracy of the scheme. The developed algorithm additionally serves as a framework for a geometrically accurate decoupling scheme to solve a wide range of flow-coupled phase-field systems on complicated surfaces, while maintaining second-order time accuracy, unconditional energy stability, decoupling structure, and linearity.

The rest of this paper is organized as follows: In Section 2, we introduce the phase-field Cahn–Hilliard equation coupled with the Navier–Stokes (CH-NS) system on complex curved surfaces, and achieve the law of energy dissipation. In Section 3, we describe the fully discrete time–space scheme, which combines the subdivision-based IGA method with temporal marching techniques including EIEQ, ZEC, and projection methods. We also provide rigorous proofs of the unconditional stability of energy and the solvability of the final linear system achieved through the utilization of the nonlocal variable splitting technique. In Section 4, we present various engaging numerical examples. We give the conclusion in Section 5.

2. Problem description

In this section, we provide a brief introduction to the CH-NS model on a complex curved surface of a closed manifold. This highly nonlinear system comprises the Cahn–Hilliard equation and the Navier–Stokes equations, commonly employed to represent the dynamics of a two-phase incompressible fluid flow system. To lay the groundwork, we initially introduce some fundamental concepts related to surfaces in the next subsection.

2.1. Preliminaries

Some basic notations for surfaces are provided here. Assuming the domain of interest is a surface, we proceed with surface parameterization. For a sufficiently smooth and orientable surface $S := \{\mathbf{x}(u^1, u^2) \in \mathbb{R}^3 : (u^1, u^2) \in \mathcal{D} \subset \mathbb{R}^2\}$, with the help of its parameterization (u^1, u^2) , we denote the following derivative formula

$$\mathbf{x}_{u^\alpha} = \frac{\partial \mathbf{x}}{\partial u^\alpha}, \quad \mathbf{x}_{u^\alpha u^\beta} = \frac{\partial^2 \mathbf{x}}{\partial u^\alpha \partial u^\beta}, \quad \alpha, \beta = 1, 2. \quad (2.1)$$

We have the coefficients of the first fundamental form on a surface, $g_{\alpha\beta} = (\mathbf{x}_{u^\alpha}, \mathbf{x}_{u^\beta})$, $\alpha, \beta = 1, 2$, and the important surface normal

$$\mathbf{n} = \frac{\mathbf{x}_{u^1} \times \mathbf{x}_{u^2}}{\|\mathbf{x}_{u^1} \times \mathbf{x}_{u^2}\|}, \quad (2.2)$$

where $(\cdot, \cdot)$, $\cdot \times \cdot$ and $\|\cdot\|$ stand for the usual inner product, cross product and Euclidean norm in $\mathbb{R}^3$ respectively. We need denote $[g^{\alpha\beta}] = [g_{\alpha\beta}]^{-1}$ and $g = \det[g_{\alpha\beta}]$ before introducing the following classical differential geometric operators.

2.1.1. Tangential gradient operator

On the function space $C^1(S)$, for any $f \in C^1(S)$, the tangential gradient operator ∇_s is defined as

$$\nabla_s f = [\mathbf{x}_{u^1}, \mathbf{x}_{u^2}][g^{\alpha\beta}][f_{u^1}, f_{u^2}]^T \in \mathbb{R}^3. \quad (2.3)$$

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

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

2.1.2. Divergence operator

On the smooth vector field $\mathbf{v} \in [C^1(S)]^3$, the divergence operator div_s is defined as

$$\text{div}_s(\mathbf{v}) = \frac{1}{\sqrt{g}} \left[\frac{\partial}{\partial u^1} \left[\sqrt{g} [g^{\alpha\beta}] [\mathbf{x}_{u^1}, \mathbf{x}_{u^2}]^T \mathbf{v} \right] \right]. \quad (2.4)$$

2.1.3. Laplace–Beltrami operator

For any $f \in C^2(S)$, the Laplace–Beltrami operator (LBO) Δ_s is defined as

$$\Delta_s f = \text{div}_s(\nabla_s f). \quad (2.5)$$

2.1.4. Sobolev space on surface

For a given constant k and a function $f \in C^\infty(S)$, we denote $\nabla^k f$ the k th order covariant derivative of function f , with the convention $\nabla^0 f = f$. Let

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

and S be a compact surface with at least k th order smoothness Sobolev space $H^k(S)$, which is the completion of $C_k(S)$ in the sense of norm

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

For any functions $\xi, \rho \in H^1(S)$, two classical inner products on the surface S are described as

$$(\xi, \rho) = \int_S \xi \rho d\mathbf{x}, \quad \text{and} \quad (\nabla_s \xi, \nabla_s \rho) = \int_S \nabla_s \xi \cdot \nabla_s \rho d\mathbf{x}.$$

2.2. CH-NS system

Suppose S be a sufficiently smooth and orientable surface. We use $\phi(\mathbf{x}, t)$ as a phase-field variable to mark the volume fraction of the two different fluid components in the fluid mixture, i.e.,

$$\phi(\mathbf{x}, t) = \begin{cases} -1 & \text{fluid component 1,} \\ 1 & \text{fluid component 2.} \end{cases} \quad (2.7)$$

These two distinct regions of $\phi(\mathbf{x}, t)$ are smoothly attached by a thin spanned region with a width of $O(\epsilon)$, $\epsilon \leq 1$. To simplify the system, we make the assumption that the two fluids share the same density. Introducing the velocity field $\mathbf{u}$ in the fluid mixture, we assume the total free energy to be

$$E(\mathbf{u}, \phi) = \int_s \left(\frac{1}{2} |\mathbf{u}|^2 + \lambda \left(\frac{1}{2} |\nabla_s \phi|^2 + F(\phi) \right) \right) d\mathbf{x}, \quad (2.8)$$

which is a summation of the gradient potential $|\nabla_s \phi|^2$ and Ginzburg–Landau type double-well functional $F(\phi) = \frac{1}{4\epsilon^2} (\phi^2 - 1)^2$. The model parameter λ depends on the surface tension, which is defined as $\frac{2\sqrt{2}}{3} \frac{\lambda}{\epsilon}$ [45]. The gradient entropy represents the hydrophilic type of interaction or the tendency to mix, and the other double-well potential describes the hydrophobic type of interaction or the tendency to separation.

The CH-NS system is characterized by adopting the H^{-1} gradient flow approach for the phase-field variable. Simultaneously, the system assumes that the fluid is incompressible, adhering to the generalized Fick's law, which dictates that the mass flux is proportional to the chemical potential gradient. This combination of conditions results in the following formulation for the CH-NS system:

$$\mathbf{u}_t + (\mathbf{u} \cdot \nabla_s) \mathbf{u} - \nu \Delta_s \mathbf{u} + \nabla_s p + \phi \nabla_s \omega = \mathbf{0}, \quad (2.9)$$

$$\operatorname{div}_s \mathbf{u} = 0, \quad (2.10)$$

$$\phi_t + \nabla_s \cdot (\mathbf{u} \phi) = M \Delta_s \omega, \quad (2.11)$$

$$\omega = \lambda (-\Delta_s \phi + f(\phi)), \quad (2.12)$$

where M is the mobility parameters, ω is the chemical potential, $\phi \nabla_s \omega$ is the surface tension term, $\nabla_s (\mathbf{u} \phi)$ is the advection, ν is the fluid viscosity, p is the pressure, and $f(\phi) = F'(\phi) = \frac{1}{\epsilon} (\phi^3 - \phi)$ and the variational derivative $\omega = \frac{\delta E}{\delta \phi}$. In this article we only treat closed surface domains, and remove all boundary conditions.

Next, we will show the character of the energy dissipation for the CH-NS dynamical system (2.9)–(2.12), as follows.

Lemma 2.1. *The CH-NS dynamical system (2.9)–(2.12) satisfies*

$$\frac{d}{dt} E(\phi, \mathbf{u}) = -M \|\nabla_s \omega\|^2 - \nu \|\nabla_s \mathbf{u}\|^2 \leq 0, \quad (2.13)$$

where $E(\mathbf{u}, \phi)$ is defined as (2.8).

Proof. By taking the L^2 inner products of (2.9) with $\mathbf{u}$, considering (2.10) and using the integration by parts, we get

$$\frac{1}{2} \frac{d}{dt} \|\mathbf{u}\|^2 + \nu \|\nabla_s \mathbf{u}\|^2 + (\phi \nabla_s \omega, \mathbf{u}) + ((\mathbf{u} \cdot \nabla_s) \mathbf{u}, \mathbf{u}) = 0, \quad (2.14)$$

where we apply $(\mathbf{u}_t, \mathbf{u}) = \frac{1}{2} \frac{d}{dt} \|\mathbf{u}\|^2$.

By taking the L^2 inner products of (2.11) with $-\omega$ and integration by parts, we get

$$M \|\nabla_s \omega\|^2 + (\nabla_s \cdot (\mathbf{u} \phi), \omega) = -(\phi_t, \omega). \quad (2.15)$$

By taking the L^2 inner product of (2.12) with ϕ_t and integration by parts, we obtain

$$\lambda \frac{d}{dt} \left(\frac{1}{2} \|\nabla_s \phi\|^2 \right) + \lambda \frac{d}{dt} \int_s F(\phi) d\mathbf{x} = (\omega, \phi_t). \quad (2.16)$$

By summing up the aforementioned three equalities and consolidating all three terms associated with $\frac{d}{dt}$ on the left-hand side, we derive (2.13). $\square$

In the preceding proof, the cancellation of inner products involving nonlinear terms is attributed to the properties stated as

$$(\nabla_s \cdot (\mathbf{u} \phi), \omega) + (\phi \nabla_s \omega, \mathbf{u}) = 0 \quad \text{and} \quad ((\mathbf{u} \cdot \nabla_s) \mathbf{u}, \mathbf{u}) = 0. \quad (2.17)$$

These crucial zero equalities, referred to as the ‘zero-energy-contribution’ (ZEC) property (cf. [46,47]), arise from integration by parts, causing these nonlinear terms to vanish and not contribute to the total energy. This property serves as a key element guiding our approach to decoupling strategies.

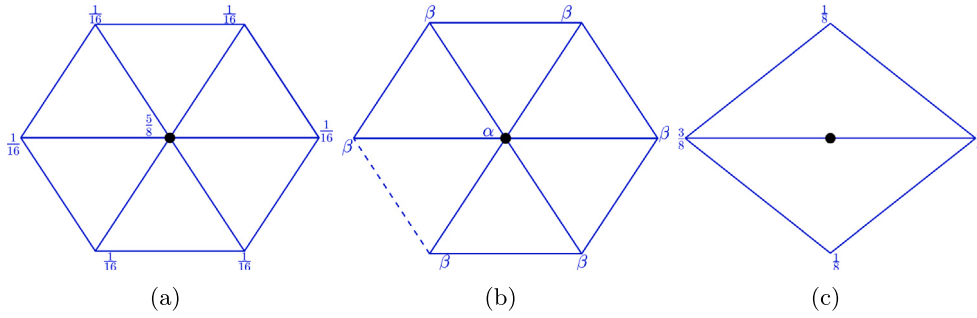


Fig. 3.1. (a): Vertex mask for a regular vertex, which has 6 adjacent vertices, and (b): Vertex mask for an irregular vertex, where the number of its adjacent points is not equal to 6, and the weight coefficient is $\alpha = 1 - n\beta$, $\beta = \frac{1}{n} \left[\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} \cos \frac{2\pi}{n} \right)^2 \right]$ and n is its valence. (c): Edge mask.

3. IGA-EIEQ scheme on the surface

In this section, we present the spatiotemporally fully discrete scheme for the model system (2.9)–(2.12). The IGA paradigm unifies the shape parameterization methods adopted in CAD and the discretization techniques used in FEM. We consider the domains of geometrically and topologically complex surface. Subdivision surfaces originate from CAD (see [19–21]) and are widely applied in computer animation applications. The subdivision technique can generate smooth surfaces on unstructured meshes, which is similar to B-splines or NURBS on structured meshes. In the FEM method, subdivision surfaces have superior conditions for square integrability with second-order derivatives. We adopt the recently developed subdivision-based IGA framework for spatial discretization and introduce the EIEQ method, merging the ZEC concept and the projection approach for time marching [41–43]. We prove that the discrete scheme is unconditionally energy stable and present the corresponding decoupling implementation approach to obtain high computational efficiency.

3.1. Subdivision-based IGA method for space discretization

The well-known IGA methodology adopts the NURBS/T-splines [48–51], which uses the same set of basis functions to describe the solution domain and implement the numerical computation of PDEs. The framework of IGA not only develops the seamless integration between FEM and CAD, but also has higher numerical exactness than FEM through p -refinement, h -refinement, and even k -refinement with the knot insertion and/or order elevation. Therefore we can improve the accuracy of the numerical simulation without destroying the original geometric properties, which removes the interactive communication with the CAD system.

Subdivision technology was proposed in computer graphics area, which can construct smooth surfaces from arbitrary topological meshes by designing a set of simple and efficient refinement schemes [19,20], and handle complicated geometric models while maintaining original characteristics near boundaries through straightforward extensions, such as concave/convex angles and sharp/smooth creases. Subdivision is compatible with NURBS, which has the capability of the refineability of B-spline techniques. Both Loop subdivision [24,26] and Catmull–Clark subdivision [25] have been utilized in IGA. The subdivision-based IGA methodology can be viewed as the natural choice for higher-order FEM in engineering practice, see [22–24,26] as well.

In this work, we apply loop subdivision surfaces based on triangle meshes, or more exactly, the essential subdivision shape functions, for the discretization method of the weak form. The surface subdivision technique possesses superior conditions of high-order smoothness and suitability for any topological structure, which is widely used in high-order model such as shell model [22,23]. We can generate spline-class surfaces using repeated refinement on a given initial mesh, which is essentially a discretization method. In addition to the flexible expression of geometric shapes, the high-order smoothness of subdivision surfaces also makes them ideal for solving high-order PDEs.

3.1.1. Loop subdivision

We adopt Loop subdivision approach [20], which creates quartic box-splines defined on three-directional triangular meshes with arbitrary topological structure. We know that the quartic box-splines are defined on a translationally invariant three-dimensional mesh, which means that the mesh is only composed of triangular patches, and each control vertex is related to adjacent triangular patches. As different from quartic box-splines, for Loop scheme, each control vertex can have any number of triangular patches, and the resulting refiner surfaces can be at least C^1 continuous. In each refinement step, the refined meshes are achieved by repeating quadrilateral segmentation of the previous mesh, with each refinement step subdividing the triangle into four smaller triangles. The position of each control vertex is recalculated as a weighted average of itself and its neighboring control vertices. The weight coefficients are represented in the form of subdivisions stencils. Vertex stencils are introduced to recalculate the coordinates of existing control vertices of the coarse mesh (see Fig. 3.1(a) and (b)). Edge stencils are introduced to compute the coordinates of control vertices generated on the edge during each quartering (see Fig. 3.1(c)). It should be noted that the weight coefficients only depend on the connect relationship of the mesh and not on the actual positions of the control vertices.

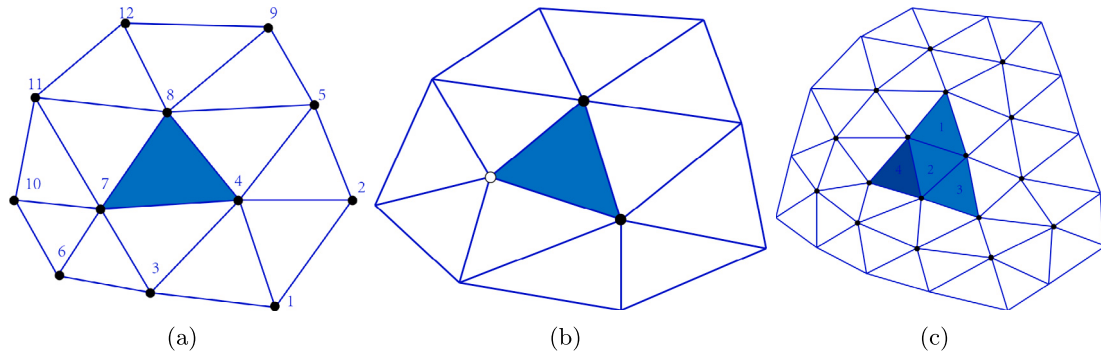


Fig. 3.2. (a): We can evaluate a regular Loop subdivision surface patch with its neighboring 12 control points. (b): We can evaluate an irregular loop subdivision surface patch whose neighboring control points are not equal to 12. The point marked in the hollow ring is an irregular point whose valence is five not six. (c): Quartering this patch can get four sub-patches. Three sub-patches marked in 1, 2, 3 are computable, and one sub-patch marked in 4 is uncomputable, which should be repeatedly quartered to obtain more sub-patches.

We need to compute quantities, such as the tangential vector and curvature, at any location on the parameterized subdivision surface. Loop subdivision can generate box-splines on patches supported only by regular control vertices, i.e.,

$$\mathbf{x}(u^1, u^2) = \sum_{i=1}^{12} B_i(u^1, u^2) \mathbf{x}_i^k. \quad (3.1)$$

We use $(u^1, u^2, 1 - u^1 - u^2)$ denote the barycentric coordinates of the unit parametric triangle and $\mathbf{x}_i^k$ denote its 2-ring surrounding neighbor control vertices of the control triangle (see Fig. 3.2(a)), and use B_i as the quartic box-splines (see the details in [21]). A patch containing irregular control vertices whose valence is different from 6 can also be parameterized with box spline shape functions. It is crucial to recognize that in the process of refinement, only regular vertices are generated through the original coarse triangular mesh divided by quartered separating, which means that the valence of all new vertices is 6 (see Fig. 3.2(b) and (c)). Obviously, the total number of irregular control vertices in the mesh remains unchanged, and their influence domain finally closes to the limit of zero. Therefore, in the process of repeated subdivision and refinement, we can obtain more and more regular patches. We should note that Stam proposed a fast parameterization method for subdivision surfaces, which coincides with the spline surface on the regular patch [21].

3.1.2. Loop subdivision finite element

We next analyze the smoothness and convergence properties of loop subdivision schemes. For the convergence properties of subdivision schemes at each control vertex, we should consider their one-ring adjacent control vertices. We denote an assembly matrix

$$\mathbf{X}^0 = (\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)^T, \quad (3.2)$$

where $\mathbf{X}^0$ is a $(n+1) \times 3$ matrix, and valence n denotes the number of its connected edges. The superscript 0 indicates the subdivision level. We can compute the positions of $\mathbf{X}^k$, where the initial positions $\mathbf{X}^0$ is multiplied k times with the subdivision matrix $\mathbf{S}$

$$\mathbf{X}^k = \mathbf{S}^k \mathbf{X}^0. \quad (3.3)$$

We can refer the fast algorithm about the subdivision matrix $\mathbf{S}^k$ as [21] because it depends only on the valence of the control vertex and unrelated to the control vertex positions. A limit surface can be achieved as the subdivision level $k \rightarrow \infty$, whose every control vertex has the explicit position, as described in the following Lemma 3.1. The limit surface of Loop subdivision is C^2 everywhere except at the extraordinary vertices where it is C^1 .

Lemma 3.1. Let $\mathbf{x}_0^k$ be a control vertex of valence n on the mesh Ω_n^k , and $\mathbf{x}_j^k, j = 1, \dots, n$, be its 1-ring neighbor-controlled vertices. All these vertices converge to a single position

$$\hat{\mathbf{x}}_0 = (1 - nl)\mathbf{x}_0^k + l \sum_{j=1}^n \mathbf{x}_j^k, \quad l = \frac{1}{n + 3/(8\beta)}, \quad (3.4)$$

as the subdivision time $k \rightarrow \infty$ (see [20] for the proof).

A quartic box spline shape function contains two rings of neighboring elements as its support, which consists of piecewise polynomial surface patches that run smoothly across the surrounding patch boundaries. The finite element solution procedure is implemented on the limit surface. Loop basis functions naturally satisfy the H^2 -smoothness requirement of the fourth-order PDEs and exactly describe arbitrary topology, which provides a uniform mathematical description for representing both the geometry

and the physics character of the complex geometry. We perform actual calculations at the Gaussian integral points on the limit surface (3.4), so we need to establish a mapping from the unit parameter triangle to the control surface patch, where we can refer it to [24–26].

3.2. Spatiotemporal fully discrete formation

3.2.1. Reformulation

Firstly, according to the initial version of IEQ strategy, we only need to define a local-type auxiliary variable U as

$$U(\phi) = \sqrt{F(\phi) - \frac{S}{2\epsilon^2}\phi^2 + B}. \quad (3.5)$$

Here, the negative term $-\frac{S}{2\epsilon^2}\phi^2$ can be bounded for any S , and the constant B needs to be positive to make sure the summation of all terms in the square root is always positive. The goal of designing the variable U is to guarantee the original energy (2.8) to be quadratic. Using the new variable U , the new CH-NS system can be rewritten as

$$\mathbf{u}_t + (\mathbf{u} \cdot \nabla_s) \mathbf{u} - \nu \Delta_s \mathbf{u} + \nabla_s p + \phi \nabla_s \omega = \mathbf{0}, \quad (3.6)$$

$$\operatorname{div}_s \mathbf{u} = 0, \quad (3.7)$$

$$\phi_t + \nabla_s \cdot (\mathbf{u} \phi) = M \Delta_s \omega, \quad (3.8)$$

$$\omega = \lambda(-\Delta_s \phi + \frac{S}{\epsilon^2} \phi + HU), \quad (3.9)$$

$$U_t = \frac{1}{2} H \phi_t, \quad (3.10)$$

where the new term

$$H(\phi) = \frac{f(\phi) - \frac{S}{\epsilon^2} \phi}{\sqrt{F(\phi) - \frac{S}{2\epsilon^2} \phi^2 + B}}. \quad (3.11)$$

Secondly, we modify the IEQ method to the explicit-IEQ (EIEQ) method combined with the ZEC method by introducing an auxiliary nonlocal variable and design a special type but trivial ODE for it. We introduce another nonlocal variable $Q(t)$ and an ODE system related to it that reads as

$$\begin{cases} Q_t = (\nabla_s \cdot (\mathbf{u} \phi), \omega) + ((\mathbf{u} \cdot \nabla_s) \mathbf{u}, \mathbf{u}) + (\phi \nabla_s \omega, \mathbf{u}) + \lambda(HU, \phi_t) - \lambda(H\phi_t, U), \\ Q|_{t=0} = 1. \end{cases} \quad (3.12)$$

Obviously, from (2.17), we get $Q_t = 0$. Using the initial condition $Q|_{t=0} = 1$, we get the exact solution of (3.12) is $Q(t) \equiv 1$. The last two terms in (3.12) correspond to the EIEQ technique [32,33,41–43], while the first three terms in (3.12) belong to the ZEC decoupling method [37–40]. Note that the objective of designing Q in this manner is to achieve the decoupled type scheme while ensuring the energy stability, as detailed in Theorem 3.2 and Section 3.2.4.

Therefore, by combining the two new variables U and Q , we rewrite the system (3.6)–(3.10) into the following equivalent form:

$$\mathbf{u}_t + Q(\mathbf{u} \cdot \nabla_s) \mathbf{u} - \nu \Delta_s \mathbf{u} + \nabla_s p + Q\phi \nabla_s \omega = \mathbf{0}, \quad (3.13)$$

$$\operatorname{div}_s \mathbf{u} = 0, \quad (3.14)$$

$$\phi_t + Q \nabla_s \cdot (\mathbf{u} \phi) = M \Delta_s \omega, \quad (3.15)$$

$$\omega = \lambda(-\Delta_s \phi + \frac{S}{\epsilon^2} \phi + QHU), \quad (3.16)$$

$$U_t = \frac{1}{2} Q H \phi_t, \quad (3.17)$$

$$Q_t = (\nabla_s \cdot (\mathbf{u} \phi), \omega) + (\phi \nabla_s \omega, \mathbf{u}) + ((\mathbf{u} \cdot \nabla_s) \mathbf{u}, \mathbf{u}) + \lambda(HU, \phi_t) - \lambda(H\phi_t, U). \quad (3.18)$$

The initial conditions read as

$$\begin{cases} \phi|_{t=0} = \phi_0, \omega|_{t=0} = \lambda(-\Delta_s \phi_0 + f(\phi_0)), p|_{t=0} = p_0, \mathbf{u}|_{t=0} = \mathbf{u}_0, \\ Q|_{t=0} = 1, U(\phi) = \sqrt{F(\phi_0) - \frac{S}{2\epsilon^2} \phi_0^2 + B}. \end{cases} \quad (3.19)$$

Note that we multiply the surface tension term $\phi \nabla_s \omega$ and the advection term $\nabla_s \cdot (\mathbf{u} \phi)$ by the nonlocal variable Q . However, it is important to emphasize that (3.13)–(3.18) remains unaffected as $Q(t) \equiv 1$. In addition, integrating (3.17) with respect to time t and incorporating the initial condition (3.19) enables us to readily obtain the original system (2.12).

The inherent characteristic of the hydrodynamically coupled phase-field system is its adherence to the energy dissipation law. Consequently, the newly transformed system (3.13)–(3.18) also preserves the structure of the energy dissipation law, as stated in the following Theorem 3.1.

Theorem 3.1. The transformed equivalent system (3.13)–(3.18) holds the law of the energy dissipation as

$$\frac{d}{dt} E(\phi, Q, U, \mathbf{u}) = -M \|\nabla_s \omega\|^2 - \nu \|\nabla_s \mathbf{u}\|^2 \leq 0, \quad (3.20)$$

where

$$E(\phi, Q, U, \mathbf{u}) = \int_S \left(\frac{1}{2} |\mathbf{u}|^2 + \frac{\lambda}{2} |\nabla_s \phi|^2 + \frac{\lambda S}{2\epsilon^2} |\phi|^2 + \lambda |U|^2 - \lambda B \right) dx + \frac{1}{2} |Q|^2 - \frac{1}{2}. \quad (3.21)$$

Proof. By taking the L^2 inner product of (3.13) with $\mathbf{u}$, then applying (3.14), we obtain

$$\frac{1}{2} \frac{d}{dt} \|\mathbf{u}\|^2 + \nu \|\nabla_s \mathbf{u}\|^2 = -Q((\mathbf{u} \cdot \nabla_s) \mathbf{u}, \mathbf{u}) - Q(\phi \nabla_s \omega, \mathbf{u}). \quad (3.22)$$

By taking the L^2 inner product of (3.15) with ω , we get

$$(\phi_t, \omega) = -M \|\nabla_s \omega\|^2 - Q(\nabla_s \cdot (\mathbf{u} \phi), \omega). \quad (3.23)$$

By taking the L^2 inner product of (3.16) with $-\phi_t$ in L^2 space, and using the integration by parts to get

$$-(\omega, \phi_t) = -\frac{1}{2} \lambda \frac{d}{dt} \|\nabla_s \phi\|^2 - \frac{\lambda S}{2\epsilon^2} \frac{d}{dt} \|\phi\|^2 - \lambda Q(HU, \phi_t). \quad (3.24)$$

By taking the L^2 inner product of (3.17) with $2\lambda U$, we obtain

$$\lambda \frac{d}{dt} \|U\|^2 = \lambda Q(H\phi_t, U), \quad (3.25)$$

where $2\lambda(U_t, U) = \lambda \frac{d}{dt} \|U\|^2$.

By multiplying (3.18) with Q , we obtain

$$\begin{aligned} \frac{d}{dt} \left(\frac{|Q|^2}{2} \right) = & -Q(\mathbf{u} \phi, \nabla_s \omega) + Q(\phi \nabla_s \omega, \mathbf{u}) + Q((\mathbf{u} \cdot \nabla_s) \mathbf{u}, \mathbf{u}) \\ & + \lambda Q(HU, \phi_t) - \lambda Q(H\phi_t, U). \end{aligned} \quad (3.26)$$

After combining (3.22)–(3.26) and observing that all terms with the factor Q cancel out, we derive the law of energy dissipation (3.21). $\square$

3.2.2. Numerical scheme

In this subsection, we describe the fully discrete scheme for solving the transformed dynamical system (3.13)–(3.18). In what follows, the time step size is denoted as $\delta t > 0$ and $t^n = n\delta t$. The test functions are chosen as $\xi_h, \theta_h, \vartheta_h, \varsigma_h, \zeta_h \in H^2(S_h)$, which is the finite element space supported by Loop subdivision elements described in Section 3.1.

Assuming that $\tilde{\mathbf{u}}_h^k, \phi_h^k, \omega_h^k, U_h^k, Q^k, \mathbf{u}_h^k$ for $k = n, n-1, n-2$, and $n \geq 2$ are known, we proceed with the update:

$$\begin{cases} \phi_h^* = 2\phi_h^n - \phi_h^{n-1}, & \omega_h^* = 2\omega_h^n - \omega_h^{n-1}, & \mathbf{u}_h^* = 2\mathbf{u}_h^n - \mathbf{u}_h^{n-1}, \\ U_h^* = U(\phi_h^*), & H_h^* = H(\phi_h^*), & \phi_{ht}^* = \frac{5\phi_h^n - 8\phi_h^{n-1} + 3\phi_h^{n-2}}{2\delta t}. \end{cases} \quad (3.27)$$

Furthermore, we calculate $\tilde{\mathbf{u}}_h^{n+1}, \phi_h^{n+1}, \omega_h^{n+1}, U_h^{n+1}, Q^{n+1}, \mathbf{u}_h^{n+1}$ by the following second-order difference formula (BDF2):

$$\left(\frac{3\tilde{\mathbf{u}}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}}{2\delta t}, \xi_h \right) + Q^{n+1}((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \xi_h) + \nu (\nabla_s \tilde{\mathbf{u}}_h^{n+1}, \nabla_s \xi_h) \quad (3.28)$$

$$+ (\nabla_s p_h^n, \xi_h) + Q^{n+1}(\phi_h^* \nabla_s \omega_h^*, \xi_h) = 0,$$

$$\left(\frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}, \theta_h \right) - Q^{n+1}(\mathbf{u}_h^* \phi_h^*, \nabla_s \theta_h) = -M(\nabla_s \omega_h^{n+1}, \nabla_s \theta_h), \quad (3.29)$$

$$(\omega_h^{n+1}, \vartheta_h) = \lambda (\nabla_s \phi_h^{n+1}, \nabla_s \vartheta_h) + \frac{\lambda S}{\epsilon^2} (\phi_h^{n+1}, \vartheta_h) + \lambda Q^{n+1}(H_h^* U_h^*, \vartheta_h), \quad (3.30)$$

$$\left(\frac{3U_h^{n+1} - 4U_h^n + U_h^{n-1}}{2\delta t}, \varsigma_h \right) = \frac{1}{2} Q^{n+1}(H_h^* \phi_{ht}^*, \varsigma_h), \quad (3.31)$$

$$\frac{3Q^{n+1} - 4Q^n + Q^{n-1}}{2\delta t} = -(\mathbf{u}_h^* \phi_h^*, \nabla_s \omega_h^{n+1}) + (\phi_h^* \nabla_s \omega_h^*, \tilde{\mathbf{u}}_h^{n+1}) + ((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \tilde{\mathbf{u}}_h^{n+1}) \quad (3.32)$$

$$+ \lambda (H_h^* U_h^*, \frac{3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}) - \lambda (H_h^* \phi_{ht}^*, U_h^{n+1}), \quad (3.33)$$

and

$$(\nabla_s (p_h^{n+1} - p_h^n), \nabla_s \zeta_h) = -\frac{3}{2\delta t} (\nabla_s \cdot \tilde{\mathbf{u}}_h^{n+1}, \zeta_h), \quad (3.34)$$

$$\mathbf{u}_h^{n+1} = \tilde{\mathbf{u}}_h^{n+1} - \frac{2\delta t}{3} (\nabla_s p_h^{n+1} - \nabla_s p_h^n). \quad (3.35)$$

Remark 3.1. The time derivative is discretized using the BDF2 method. However, achieving a high-order formula like BDF3/4 poses significant challenges, primarily due to stability issues. Additionally, a direct discretization method is used in the above scheme. Specifically, linear components are implicitly treated, while nonlinear components involving a nonlocal variable Q are treated through the explicit-implicit schemes. It means that we implicitly process the term Q , and explicitly process all other terms. The decoupling method of the above scheme is obviously coupled and must fully utilize the nonlocal characteristics of the Q variable. Next, we will demonstrate the energy stability of the developed scheme, followed by the implementation details of decoupling.

Remark 3.2. We adopt a projection-based approach for the incompressible fluid dynamic system. This algorithm can decouple the calculation of pressure and velocity fields. The results show that the velocity field and pressure field can obtain second-order and first-order accuracy in time, respectively, and strict error estimates are provided [52]. The reason for the first-order accuracy loss of pressure is the application of artificial homogeneous Neumann boundary conditions to the pressure [53].

The final velocity field is denoted as $\mathbf{u}_h^n$, and the intermediate velocity field is denoted as $\tilde{\mathbf{u}}_h^n$, which satisfy the divergence-free condition in a discrete sense. For $\nabla v_h, v_h \in O_h$, we take the L^2 inner product of (3.35) to obtain

$$(\mathbf{u}_h^{n+1}, \nabla_s v_h) = -(\nabla_s \cdot \tilde{\mathbf{u}}_h^{n+1}, v_h) - \frac{2\delta t}{3} (\nabla_s (p_h^{n+1} - p_h^n), \nabla_s v_h). \quad (3.36)$$

Therefore, from (3.34), we derive

$$(\mathbf{u}_h^{n+1}, \nabla_s v_h) = 0. \quad (3.37)$$

3.2.3. Energy stability

We aim to demonstrate that the proposed discrete scheme (3.28)–(3.35) unconditionally satisfies the energy dissipation law, akin to the continuous system (3.13)–(3.18). Prior to presenting the proof in Theorem 3.2, we introduce two identities that will be repeatedly used:

$$2(a - b, a) = |a|^2 - |b|^2 + |a - b|^2, \quad (3.38)$$

$$2(3a - 4b + c)a = |a|^2 - |b|^2 + |2a - b|^2 - |2b - c|^2 + |a - 2b + c|^2. \quad (3.39)$$

Theorem 3.2. The discrete scheme (3.28)–(3.35) is second-order convergent and unconditionally energy stable, i.e., satisfies the following discrete energy dissipation law:

$$\frac{E^{n+1} - E^n}{\delta t} \leq -M \|\nabla_s \omega_h^{n+1}\|^2 - \nu \|\nabla_s \tilde{\mathbf{u}}_h^{n+1}\|^2 \leq 0, \quad (3.40)$$

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

$$\begin{aligned} E^k &= \frac{1}{2} (\|\mathbf{u}_h^k\|^2 + \|2\mathbf{u}_h^k - \mathbf{u}_h^{k-1}\|^2) + \frac{2\delta t^2}{3} \|\nabla_s p^k\|^2 \\ &\quad + \frac{\lambda}{2} (\|\nabla_s \phi_h^k\|^2 + \|2\nabla_s \phi_h^k - \nabla_s \phi_h^{k-1}\|^2) + \frac{\lambda S}{2\epsilon^2} (|\phi_h^k|^2 + |2\phi_h^k - \phi_h^{k-1}|^2) \\ &\quad + \lambda (|U^k|^2 + |2U^k - U^{k-1}|^2) + \frac{1}{2} (|Q^k|^2 + |2Q^k - Q^{k-1}|^2) - \lambda B|S| - \frac{1}{2}. \end{aligned} \quad (3.41)$$

Proof. By taking the test function $\xi_h = 2\delta t \tilde{\mathbf{u}}_h^{n+1}$ in (3.28), we get

$$\begin{aligned} (3\tilde{\mathbf{u}}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1}) + 2\delta t Q^{n+1} ((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \tilde{\mathbf{u}}_h^{n+1}) + 2\nu \delta t \|\nabla_s \tilde{\mathbf{u}}_h^{n+1}\|^2 \\ + 2\delta t (\nabla_s p_h^n, \tilde{\mathbf{u}}_h^{n+1}) + 2\delta t Q^{n+1} (\phi_h^* \nabla_s \omega_h^*, \tilde{\mathbf{u}}_h^{n+1}) = 0. \end{aligned} \quad (3.42)$$

We rewrite (3.35) as

$$\tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^{n+1} = \frac{2\delta t}{3} \nabla_s (p_h^{n+1} - p_h^n). \quad (3.43)$$

Taking the L^2 inner product of the above equality with $\mathbf{u}_h^k$ with $k = n+1, n, n-1$, we derive

$$(\tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^{n+1}, \mathbf{u}_h^{n+1}) = \frac{2\delta t}{3} (\nabla_s (p_h^{n+1} - p_h^n), \mathbf{u}_h^{n+1}) = 0, \quad (3.44)$$

and

$$\begin{aligned} (3\tilde{\mathbf{u}}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^{n+1}) \\ = (3\tilde{\mathbf{u}}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \frac{2\delta t}{3} \nabla_s (p_h^{n+1} - p_h^n)) = 0, \end{aligned} \quad (3.45)$$

where (3.37) is used.

We reformulate the first term in (3.42) as

$$\begin{aligned} (3\tilde{\mathbf{u}}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1}) \\ = (3\tilde{\mathbf{u}}_h^{n+1} - 3\mathbf{u}_h^{n+1} + 3\mathbf{u}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1}) \\ = (3\tilde{\mathbf{u}}_h^{n+1} - 3\mathbf{u}_h^{n+1}, \tilde{\mathbf{u}}_h^{n+1}) + (3\mathbf{u}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1}). \end{aligned} \quad (3.46)$$

where we add the underlined term $-3\mathbf{u}_h^{n+1} + 3\mathbf{u}_h^{n+1}$, which is zero.

For the first split term $(3\tilde{\mathbf{u}}_h^{n+1} - 3\mathbf{u}_h^{n+1}, \tilde{\mathbf{u}}_h^{n+1})$ in (3.46), we deduce

$$\begin{aligned} (3\tilde{\mathbf{u}}_h^{n+1} - 3\mathbf{u}_h^{n+1}, \tilde{\mathbf{u}}_h^{n+1}) &= (3\tilde{\mathbf{u}}_h^{n+1} - 3\mathbf{u}_h^{n+1}, \tilde{\mathbf{u}}_h^{n+1}) + \underline{(3\tilde{\mathbf{u}}_h^{n+1} - 3\mathbf{u}_h^{n+1}, \mathbf{u}_h^{n+1})} \\ &= (3\tilde{\mathbf{u}}_h^{n+1} - 3\mathbf{u}_h^{n+1}, \tilde{\mathbf{u}}_h^{n+1} + \mathbf{u}_h^{n+1}) \\ &= 3\|\tilde{\mathbf{u}}_h^{n+1}\|^2 - 3\|\mathbf{u}_h^{n+1}\|^2, \end{aligned} \quad (3.47)$$

where the added underlined term is actually zero, as can be seen from (3.44).

For the second split term $(3\mathbf{u}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1})$ in (3.46), we deduce

$$\begin{aligned} (3\mathbf{u}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1}) &= (3\mathbf{u}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^{n+1} + \mathbf{u}_h^{n+1}) \\ &= \underline{(3\mathbf{u}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^{n+1})} + (3\mathbf{u}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \mathbf{u}_h^{n+1}) \\ &= (3\mathbf{u}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \mathbf{u}_h^{n+1}) \\ &= \frac{1}{2}(\|\mathbf{u}_h^{n+1}\|^2 - \|\mathbf{u}_h^n\|^2 + \|2\mathbf{u}_h^{n+1} - \mathbf{u}_h^n\|^2 - \|2\mathbf{u}_h^n - \mathbf{u}_h^{n-1}\|^2 \\ &\quad + \|\mathbf{u}_h^{n+1} - 2\mathbf{u}_h^n + \mathbf{u}_h^{n-1}\|^2), \end{aligned} \quad (3.48)$$

where the underlined term is zero due to (3.45), and the last equality is derived using (3.39).

Thus, the first term in (3.42) can be written as

$$\begin{aligned} (3\tilde{\mathbf{u}}_h^{n+1} - 4\mathbf{u}_h^n + \mathbf{u}_h^{n-1}, \tilde{\mathbf{u}}_h^{n+1}) &= \frac{1}{2}(\|\mathbf{u}_h^{n+1}\|^2 - \|\mathbf{u}_h^n\|^2 + \|2\mathbf{u}_h^{n+1} - \mathbf{u}_h^n\|^2 - \|2\mathbf{u}_h^n - \mathbf{u}_h^{n-1}\|^2 \\ &\quad + \|\mathbf{u}_h^{n+1} - 2\mathbf{u}_h^n + \mathbf{u}_h^{n-1}\|^2) + 3\|\tilde{\mathbf{u}}_h^{n+1}\|^2 - 3\|\mathbf{u}_h^{n+1}\|^2. \end{aligned} \quad (3.49)$$

We rewrite (3.35) as

$$\mathbf{u}_h^{n+1} + \frac{2\delta t}{3}\nabla_s p_h^{n+1} = \tilde{\mathbf{u}}_h^{n+1} + \frac{2\delta t}{3}\nabla_s p_h^n. \quad (3.50)$$

Taking the L^2 inner product of the above equation with itself and multiply the result with $\frac{3}{2}$, we derive

$$\begin{aligned} \frac{3}{2}\|\mathbf{u}_h^{n+1}\|^2 + \frac{2\delta t^2}{3}\|\nabla_s p_h^{n+1}\|^2 + 2\delta t(\mathbf{u}_h^{n+1}, \nabla_s p_h^{n+1}) \\ = \frac{3}{2}\|\tilde{\mathbf{u}}_h^{n+1}\|^2 + \frac{2\delta t^2}{3}\|\nabla_s p_h^n\|^2 + 2\delta t(\tilde{\mathbf{u}}_h^{n+1}, \nabla_s p_h^n). \end{aligned} \quad (3.51)$$

From (3.37), we get $2\delta t(\mathbf{u}_h^{n+1}, \nabla_s p_h^{n+1}) = 0$, thus (3.51) can be rewritten as

$$2\delta t(\tilde{\mathbf{u}}_h^{n+1}, \nabla_s p_h^n) = \frac{3}{2}\|\mathbf{u}_h^{n+1}\|^2 - \frac{3}{2}\|\tilde{\mathbf{u}}_h^{n+1}\|^2 + \frac{2\delta t^2}{3}\|\nabla_s p_h^{n+1}\|^2 - \frac{2\delta t^2}{3}\|\nabla_s p_h^n\|^2. \quad (3.52)$$

We rewrite (3.35) as

$$\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1} = -\frac{2\delta t}{3}\nabla_s p_h^{n+1} + \frac{2\delta t}{3}\nabla_s p_h^n. \quad (3.53)$$

By taking the L^2 inner product of the above equation with $3\mathbf{u}_h^{n+1}$, and using (3.37), we obtain

$$3(\mathbf{u}_h^{n+1}, \mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}) = 3(\mathbf{u}_h^{n+1}, -\frac{2\delta t}{3}\nabla_s p_h^{n+1} + \frac{2\delta t}{3}\nabla_s p_h^n) = 0. \quad (3.54)$$

Then we get by (3.54)

$$\begin{aligned} \frac{3}{2}\|\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}\|^2 &= \frac{3}{2}\|\mathbf{u}_h^{n+1}\|^2 + \frac{3}{2}\|\tilde{\mathbf{u}}_h^{n+1}\|^2 - 3(\mathbf{u}_h^{n+1}, \tilde{\mathbf{u}}_h^{n+1}) \\ &= \frac{3}{2}\|\mathbf{u}_h^{n+1}\|^2 + \frac{3}{2}\|\tilde{\mathbf{u}}_h^{n+1}\|^2 + 3(\mathbf{u}_h^{n+1}, \mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}) - 3(\mathbf{u}_h^{n+1}, \mathbf{u}_h^{n+1}) \\ &= \frac{3}{2}\|\tilde{\mathbf{u}}_h^{n+1}\|^2 - \frac{3}{2}\|\mathbf{u}_h^{n+1}\|^2. \end{aligned} \quad (3.55)$$

By combining (3.42), (3.46), (3.52), and (3.55), we get

$$\begin{aligned} \frac{1}{2}(\|\mathbf{u}_h^{n+1}\|^2 - \|\mathbf{u}_h^n\|^2 + \|2\mathbf{u}_h^{n+1} - \mathbf{u}_h^n\|^2 - \|2\mathbf{u}_h^n - \mathbf{u}_h^{n-1}\|^2 + \|\mathbf{u}_h^{n+1} - 2\mathbf{u}_h^n + \mathbf{u}_h^{n-1}\|^2) \\ + \frac{3}{2}\|\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}\|^2 + \frac{2\delta t^2}{3}(\|\nabla_s p_h^{n+1}\|^2 - \|\nabla_s p_h^n\|^2) + 2v\delta t\|\nabla_s \tilde{\mathbf{u}}_h^{n+1}\|^2 \\ + 2\delta tQ^{n+1}((\mathbf{u}_h^* \cdot \nabla_s)\mathbf{u}_h^*, \tilde{\mathbf{u}}_h^{n+1}) + 2\delta tQ^{n+1}(\phi_h^* \nabla_s \omega_h^*, \tilde{\mathbf{u}}_h^{n+1}) = 0. \end{aligned} \quad (3.56)$$

By taking $\theta_h = 2\delta t\omega_h^{n+1}$ in (3.29), we get

$$(3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}, \omega_h^{n+1}) - 2\delta tQ^{n+1}(\mathbf{u}_h^* \phi_h^*, \nabla_s \omega_h^{n+1}) + 2\delta tM\|\nabla_s \omega_h^{n+1}\|^2 = 0. \quad (3.57)$$

By taking $\vartheta_h = -(3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1})$ in (3.30), we derive

$$\begin{aligned} -(\omega_h^{n+1}, 3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}) &= -\lambda(\nabla_s \phi_h^{n+1}, 3\nabla_s \phi_h^{n+1} - 4\nabla_s \phi_h^n + \nabla_s \phi_h^{n-1}) \\ &\quad - \frac{\lambda S}{\varepsilon^2}(\phi_h^{n+1}, 3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}) \\ &\quad - \lambda Q^{n+1}(H_h^* U_h^*, 3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}). \end{aligned} \quad (3.58)$$

By taking $\varsigma_h = 4\lambda\delta t U_h^{n+1}$ in (3.31), we get

$$\begin{aligned} \lambda(|U_h^{n+1}|^2 - |U_h^n|^2 + |2U_h^{n+1} - U_h^n|^2 - |2U_h^n - U_h^{n-1}|^2 \\ + |U_h^{n+1} - 2U_h^n + U_h^{n-1}|^2) = 2\lambda\delta t Q^{n+1}(H_h^* \phi_{ht}^*, U_h^{n+1}), \end{aligned} \quad (3.59)$$

where (3.39) is used.

By multiplying (3.32) with $2\delta t Q^{n+1}$ and using (3.39), we obtain

$$\begin{aligned} \frac{1}{2}(|Q^{n+1}|^2 - |Q^n|^2 + |2Q^{n+1} - Q^n|^2 - |2Q^n - Q^{n-1}|^2 + |Q^{n+1} - 2Q^n + Q^{n-1}|^2) \\ = -2\delta t Q^{n+1}(\mathbf{u}_h^* \phi_h^*, \nabla_s \omega_h^{n+1}) + 2\delta t Q^{n+1}(\phi_h^* \nabla_s \omega_h^*, \tilde{\mathbf{u}}_h^{n+1}) \\ + \lambda Q^{n+1}(H_h^* U_h^*, 3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}) \\ + 2\delta t Q^{n+1}((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \tilde{\mathbf{u}}_h^{n+1}) - 2\lambda\delta t Q^{n+1}(H_h^* \phi_{ht}^*, U_h^{n+1}). \end{aligned} \quad (3.60)$$

We combine (3.57)–(3.60) to get the following

$$\begin{aligned} \lambda(\nabla_s \phi_h^{n+1}, 3\nabla_s \phi_h^{n+1} - 4\nabla_s \phi_h^n + \nabla_s \phi_h^{n-1}) + \frac{\lambda S}{\varepsilon^2}(\phi_h^{n+1}, 3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}) \\ + \lambda(|U_h^{n+1}|^2 - |U_h^n|^2 + |2U_h^{n+1} - U_h^n|^2 - |2U_h^n - U_h^{n-1}|^2 + |U_h^{n+1} - 2U_h^n + U_h^{n-1}|^2) \\ + \frac{1}{2}(|Q^{n+1}|^2 - |Q^n|^2 + |2Q^{n+1} - Q^n|^2 - |2Q^n - Q^{n-1}|^2 + |Q^{n+1} - 2Q^n + Q^{n-1}|^2) \\ = -2\delta t M \|\nabla_s \omega_h^{n+1}\|^2 + 2\delta t Q^{n+1}(\phi_h^* \nabla_s \omega_h^*, \tilde{\mathbf{u}}_h^{n+1}) + 2\delta t Q^{n+1}((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \tilde{\mathbf{u}}_h^{n+1}). \end{aligned} \quad (3.61)$$

For the first two terms of (3.61), by using (3.39), we have

$$\begin{aligned} \lambda(\nabla_s \phi_h^{n+1}, 3\nabla_s \phi_h^{n+1} - 4\nabla_s \phi_h^n + \nabla_s \phi_h^{n-1}) + \frac{\lambda S}{\varepsilon^2}(\phi_h^{n+1}, 3\phi_h^{n+1} - 4\phi_h^n + \phi_h^{n-1}) \\ = \frac{\lambda}{2}(\|\nabla_s \phi_h^{n+1}\|^2 - \|\nabla_s \phi_h^n\|^2 + \|2\nabla_s \phi_h^{n+1} - \nabla_s \phi_h^n\|^2 - \|2\nabla_s \phi_h^n - \nabla_s \phi_h^{n-1}\|^2) \\ + \frac{\lambda}{2}\|\nabla_s \phi_h^{n+1} - 2\nabla_s \phi_h^n + \nabla_s \phi_h^{n-1}\|^2 \\ + \frac{\lambda S}{2\varepsilon^2}(|\phi_h^{n+1}|^2 - |\phi_h^n|^2 + |2\phi_h^{n+1} - \phi_h^n|^2 - |2\phi_h^n - \phi_h^{n-1}|^2) \\ + \frac{\lambda S}{2\varepsilon^2}|\phi_h^{n+1} - 2\phi_h^n + \phi_h^{n-1}|^2. \end{aligned} \quad (3.62)$$

By combining (3.56), (3.61) and (3.62), we obtain

$$\begin{aligned} \frac{1}{2}(\|\mathbf{u}_h^{n+1}\|^2 - \|\mathbf{u}_h^n\|^2 + \|2\mathbf{u}_h^{n+1} - \mathbf{u}_h^n\|^2 - \|2\mathbf{u}_h^n - \mathbf{u}_h^{n-1}\|^2) \\ + \frac{2\delta t^2}{3}(\|\nabla_s p_h^{n+1}\|^2 - \|\nabla_s p_h^n\|^2) \\ + \frac{\lambda}{2}(\|\nabla_s \phi_h^{n+1}\|^2 - \|\nabla_s \phi_h^n\|^2 + \|2\nabla_s \phi_h^{n+1} - \nabla_s \phi_h^n\|^2 - \|2\nabla_s \phi_h^n - \nabla_s \phi_h^{n-1}\|^2) \\ + \frac{\lambda S}{2\varepsilon^2}(|\phi_h^{n+1}|^2 - |\phi_h^n|^2 + |2\phi_h^{n+1} - \phi_h^n|^2 - |2\phi_h^n - \phi_h^{n-1}|^2) \\ + \lambda(|U_h^{n+1}|^2 - |U_h^n|^2 + |2U_h^{n+1} - U_h^n|^2 - |2U_h^n - U_h^{n-1}|^2) \\ + \frac{1}{2}(|Q^{n+1}|^2 - |Q^n|^2 + |2Q^{n+1} - Q^n|^2 - |2Q^n - Q^{n-1}|^2) \\ + \left\{ \frac{1}{2}\|\mathbf{u}_h^{n+1} - 2\mathbf{u}_h^n + \mathbf{u}_h^{n-1}\|^2 + \frac{3}{2}\|\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}\|^2 \right. \\ + \lambda|U^{n+1} - 2U^n + U^{n-1}|^2 + \frac{1}{2}|Q^{n+1} - 2Q^n + Q^{n-1}|^2 \\ \left. + \frac{\lambda}{2}\|\nabla_s \phi_h^{n+1} - 2\nabla_s \phi_h^n + \nabla_s \phi_h^{n-1}\|^2 + \frac{\lambda S}{2\varepsilon^2}|\phi_h^{n+1} - 2\phi_h^n + \phi_h^{n-1}|^2 \right\} \\ = -2\delta t M \|\nabla_s \omega_h^{n+1}\|^2 - 2\delta t \nu \|\nabla_s \tilde{\mathbf{u}}_h^{n+1}\|^2. \end{aligned} \quad (3.63)$$

Finally, we obtain the desired result (3.40) after ignoring positive terms in $\{ \}$ in (3.63). $\square$

3.2.4. Decoupled type implementation

In this section, we introduce our decoupling computational steps by using a splitting technique for treating the systems (3.28)–(3.32).

Step 1: we use Q^{n+1} to split $\tilde{\mathbf{u}}_h^{n+1}$ into the linear combination form that reads as

$$\tilde{\mathbf{u}}_h^{n+1} = \tilde{\mathbf{u}}_{1h}^{n+1} + Q^{n+1} \tilde{\mathbf{u}}_{2h}^{n+1}. \quad (3.64)$$

By applying the linear form given in (3.64), we can split the scheme (3.28) as follows:

$$\left(\frac{3\tilde{\mathbf{u}}_{1h}^{n+1}}{2\delta t}, \xi_h \right) + \nu(\nabla_s \tilde{\mathbf{u}}_{1h}^{n+1}, \nabla_s \xi_h) = \left(\frac{4\mathbf{u}_h^n - \mathbf{u}_h^{n-1}}{2\delta t}, \xi_h \right) - (\nabla_s p_h^n, \xi_h), \quad (3.65)$$

$$\left(\frac{3\tilde{\mathbf{u}}_{2h}^{n+1}}{2\delta t}, \xi_h \right) + \nu(\nabla_s \tilde{\mathbf{u}}_{2h}^{n+1}, \nabla_s \xi_h) = -((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \xi_h) - (\phi_h^* \nabla_s \omega_h^*, \xi_h). \quad (3.66)$$

Hence, during this step, $\tilde{\mathbf{u}}_{ih}^{n+1} (i = 1, 2)$ can be computed from (3.65)–(3.66).

Step 2: we use Q^{n+1} to split ϕ_h^{n+1} , ω_h^{n+1} and U_h^{n+1} into the linear combination forms that read as

$$\begin{cases} \phi_h^{n+1} = \phi_{1h}^{n+1} + Q^{n+1} \phi_{2h}^{n+1}, \\ \omega_h^{n+1} = \omega_{1h}^{n+1} + Q^{n+1} \omega_{2h}^{n+1}, \\ U_h^{n+1} = U_{1h}^{n+1} + Q^{n+1} U_{2h}^{n+1}. \end{cases} \quad (3.67)$$

By applying the linear form given in (3.67), we can split the scheme (3.29)–(3.31) as follows:

$$\begin{cases} \left(\frac{3\phi_{1h}^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}, \theta_h \right) = -M(\nabla_s \omega_{1h}^{n+1}, \nabla_s \theta_h), \\ (\omega_{1h}^{n+1}, \vartheta_h) = \lambda(\nabla_s \phi_{1h}^{n+1}, \nabla_s \vartheta_h) + \frac{\lambda S}{\epsilon^2} (\phi_{1h}^{n+1}, \vartheta_h), \end{cases} \quad (3.68)$$

$$\begin{cases} \left(\frac{3\phi_{2h}^{n+1}}{2\delta t}, \theta_h \right) = -M(\nabla_s \omega_{2h}^{n+1}, \nabla_s \theta_h) + (\mathbf{u}_h^* \phi_h^*, \nabla_s \theta_h), \\ (\omega_{2h}^{n+1}, \vartheta_h) = \lambda(\nabla_s \phi_{2h}^{n+1}, \nabla_s \vartheta_h) + \frac{\lambda S}{\epsilon^2} (\phi_{2h}^{n+1}, \vartheta_h) + \lambda(H_h^* U_h^*, \vartheta_h), \end{cases} \quad (3.69)$$

$$\left(\frac{3U_{1h}^{n+1}}{2\delta t}, \varsigma_h \right) = \left(\frac{4U_h^n - U_h^{n-1}}{2\delta t}, \varsigma_h \right), \quad (3.70)$$

$$\left(\frac{3U_{2h}^{n+1}}{2\delta t}, \varsigma_h \right) = \frac{1}{2} (H_h^* \phi_{ht}^*, \varsigma_h). \quad (3.71)$$

Hence, during this step, ϕ_{ih}^{n+1} , ω_{ih}^{n+1} , $U_{ih}^{n+1} (i = 1, 2)$ can be computed from (3.68)–(3.71).

Step 3: By applying the values ϕ_{ih}^{n+1} , ω_{ih}^{n+1} , U_{ih}^{n+1} and $\tilde{\mathbf{u}}_{ih}^{n+1}$ with $i = 1, 2$, obtained from (3.65)–(3.71), we update Q^{n+1} in (3.32) through

$$\left(\frac{3}{2\delta t} - \gamma_2 \right) Q^{n+1} = \frac{1}{2\delta t} (4Q^n - Q^{n-1}) + \gamma_1, \quad (3.72)$$

where γ_1 and γ_2 are

$$\begin{cases} \gamma_1 = -(\mathbf{u}_h^* \phi_h^*, \nabla_s \omega_{1h}^{n+1}) + (\phi_h^* \nabla_s \omega_h^*, \tilde{\mathbf{u}}_{1h}^{n+1}) + ((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \tilde{\mathbf{u}}_{1h}^{n+1}) \\ \quad + \lambda(H_h^* U_h^*, \frac{3\phi_{1h}^{n+1} - 4\phi_h^n + \phi_h^{n-1}}{2\delta t}) - \lambda(H_h^* \phi_{ht}^*, U_{1h}^{n+1}), \\ \gamma_2 = -(\mathbf{u}_h^* \phi_h^*, \nabla_s \omega_{2h}^{n+1}) + (\phi_h^* \nabla_s \omega_h^*, \tilde{\mathbf{u}}_{2h}^{n+1}) + ((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \tilde{\mathbf{u}}_{2h}^{n+1}) \\ \quad + \lambda(H_h^* U_h^*, \frac{3\phi_{2h}^{n+1}}{2\delta t}) - \lambda(H_h^* \phi_{ht}^*, U_{2h}^{n+1}). \end{cases} \quad (3.73)$$

We need to prove the solvability of (3.72) by showing $\frac{3}{2\delta t} - \gamma_2 \neq 0$. First, by taking $\theta_h = \omega_{2h}^{n+1}$ and $\vartheta_h = -\frac{3}{2\delta t} \phi_{2h}^{n+1}$ in (3.69), we get

$$M \left\| \nabla_s \omega_{2h}^{n+1} \right\|^2 + \frac{3\lambda}{2\delta t} \left\| \nabla_s \phi_{2h}^{n+1} \right\|^2 + \frac{3\lambda S}{2\epsilon^2 \delta t} \left\| \phi_{2h}^{n+1} \right\|^2 = (\mathbf{u}_h^* \phi_h^*, \nabla_s \omega_{2h}^{n+1}) - \lambda(H_h^* U_h^*, \frac{3\phi_{2h}^{n+1}}{2\delta t}). \quad (3.74)$$

Then we choose $\varsigma_h = \lambda U_{2h}^{n+1}$ in (3.71) to achieve

$$\frac{3\lambda}{\delta t} \left\| U_{2h}^{n+1} \right\|^2 = \lambda(H_h^* \phi_{ht}^*, U_{2h}^{n+1}). \quad (3.75)$$

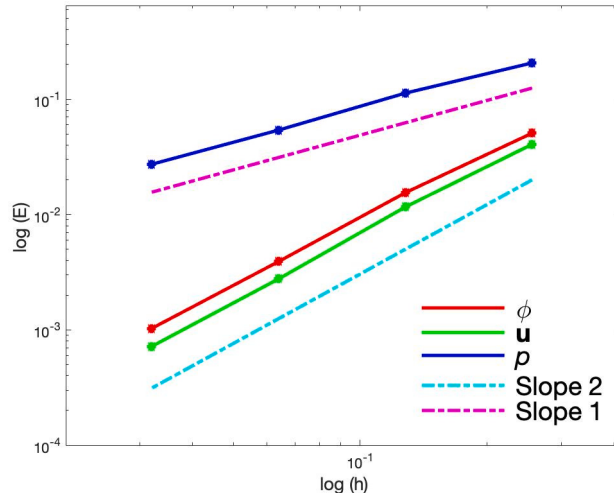


Fig. 4.1. Convergence rate of the CH-NS system on the sphere S_0 .

Finally, we choose $\xi_h = \tilde{\mathbf{u}}_{2h}^{n+1}$ in (3.66) to achieve

$$\frac{3}{2\delta t} \|\tilde{\mathbf{u}}_{2h}^{n+1}\|^2 + \nu \|\nabla_s \tilde{\mathbf{u}}_{2h}^{n+1}\|^2 = -((\mathbf{u}_h^* \cdot \nabla_s) \mathbf{u}_h^*, \tilde{\mathbf{u}}_{2h}^{n+1}) - (\phi_h^* \nabla_s \omega_h^*, \tilde{\mathbf{u}}_{2h}^{n+1}). \quad (3.76)$$

Thus, we deduce $-\gamma_2 \geq 0$ by combining (3.74), (3.75) and (3.76), which implies (3.72) is always solvable.

Step 4: $\mathbf{u}_h^{n+1}$ and p_h^{n+1} can be computed by use of (3.34) and (3.35).

4. Numerical examples

In this section, we demonstrate the effectiveness of the numerical scheme (3.28)–(3.35) developed through a series of numerical simulations. We validate its energy stability and convergence properties. The simulations cover scenarios including spinodal decomposition and droplet dripping dynamics on a complex curved surface, driven by gravitational forces. We perform the spatial discretization by use of the surface subdivision method in Section 3.1. A robust GMRES iterative solver is applied to solve the final systems derived from the variational format because these systems are highly sparse, where a small enough tolerance is set to obtain the proper convergence.

4.1. Convergence test

The considered surface is defined as a sphere with a radius of 1, as follows:

$$S_0 = \{(x, y, z) : \sqrt{x^2 + y^2 + z^2} = 1\}. \quad (4.1)$$

The initial conditions are given as

$$\begin{cases} \phi^0 = \cos(x) \cos(y) \cos(z), & p^0 = \sin(x) \sin(y) \sin(z), \\ \mathbf{u}^0 = (\cos(x) \sin(y) \cos(z), \sin(x) \cos(y) \cos(z), -\cos(x) \sin(y) \sin(z))^T. \end{cases} \quad (4.2)$$

The model parameters are set as

$$M = 1\text{e-}3, \lambda = 4\text{e-}4, S = 4, B = 1\text{e}5, \nu = 1, \epsilon = 1.2\text{e-}3. \quad (4.3)$$

We generate four refined meshes by Loop subdivision, where the total numbers of patches/vertices for these surface models are 512/258, 2048/1026, 8192/4098, 32768/16386 respectively. We denote them as S_{h_i} , where h_i represents the surface mesh size at the i th refinement level for $i = 0, 1, 2, 3$, and $h_0 = 0.2543, h_1 = 0.1189, h_2 = 0.0622, h_3 = 0.0326$. The subdivision-based IGA method has been shown to be second-order accurate in our previous work [24], hence we set the time step sizes as $\delta t_i = Ch_i$ with $C \in (0, 1)$. Since the exact solutions of this equation system are unavailable, we may set the solutions using a significantly small mesh size and consider them as reference solutions for computing numerical errors. For all unknown variables $\phi, \mathbf{u}$ and p , we plot their L^2 numerical errors at the time $t = 2.56\text{e-}1$ in Fig. 4.1, where the two variables ϕ and $\mathbf{u}$ have the second-order convergence rate and the variable p has the first-order convergence rate. These results align well with the theoretically anticipated convergence rate.

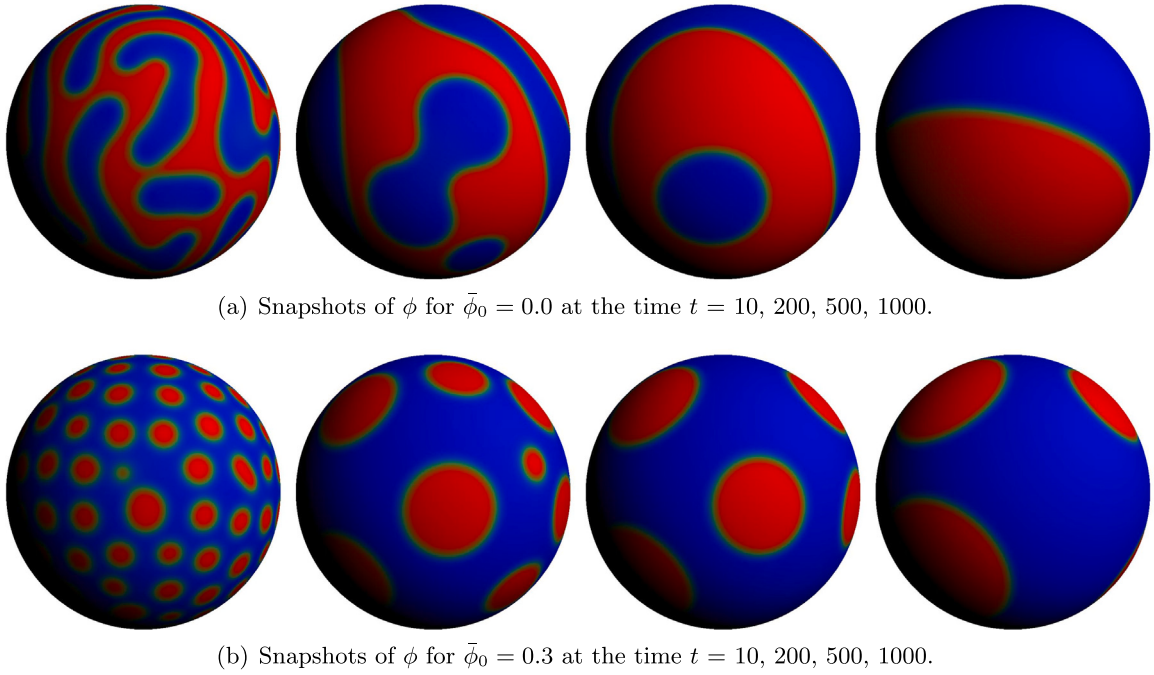


Fig. 4.2. Spinodal decomposition for the surface model S_1 , where (a) $\bar{\phi}_0 = 0.0$, and (b) $\bar{\phi}_0 = 0.3$.

4.2. Spinodal decomposition

This simulation focuses on the investigation of the dynamical modeling of the spinodal decomposition, which can show the effectiveness of our developed scheme. We set the initial state as a uniformly perturbed binary mixture within a certain range. The system goes through a transition from a homogeneous state to a stable situation with two distinct phases, driven by the essential growth of concentration fluctuations.

The interest domain is a closed sphere that reads as

$$S_1 = \{(x, y, z) : \sqrt{x^2 + y^2 + z^2} = 1\}. \quad (4.4)$$

The total numbers of patches/vertices for the surface model is 131 072/65 538, and the span of the vertex valence is 4 to 6. The initial conditions are given by

$$\phi^0 = \bar{\phi}_0 + 0.01 \text{rand}(x, y, z), \quad \mathbf{u}^0 = \mathbf{0}, \quad p^0 = 0, \quad (4.5)$$

where the term $\text{rand}(x, y, z)$ is the random number in $[-1, 1]$ with zero mean. The model parameters are set as

$$M = 1\text{e-}2, \quad \lambda = 4\text{e-}4, \quad S = 4, \quad B = 1\text{e}5, \quad \nu = 1, \quad \varepsilon = 2\text{e-}2, \quad \delta t = 1\text{e-}2. \quad (4.6)$$

In Fig. 4.2, two simulation results corresponding to initial values of $\bar{\phi}_0 = 0.0$ and $\bar{\phi}_0 = 0.3$ are presented, respectively. We plot snapshots of ϕ on the spherical surface over a period of time. Two final stable states correspond to a banded and a circular equilibrium state, respectively. Additionally, in Fig. 4.3, we plot the total free energy functionals over time for the two performed simulations, where we choose three different time steps $\delta t = 0.0025, 0.005, 0.01$ to test the energy decay situation. All of the evolving curves of the total energy monotonically decrease over time which confirm the unconditional stability of our method.

4.3. Drop dripping driven by gravitational force

In this simulation, we study the drop-dripping dynamics under the gravitational force on the complex surfaces. To mimic the gravitational force, we use the Boussinesq approximation by augmenting the fluid momentum equation (2.9) with an extra force that reads as (cf. [44,54])

$$\mathbf{u}_t + (\mathbf{u} \cdot \nabla_s) \mathbf{u} - \nu \nabla_s \nabla_s \mathbf{u} + \phi \nabla_s \omega = \phi(\rho_1 - \rho_2) \mathbf{g}, \quad \mathbf{g} = (0, 0, g), \quad (4.7)$$

where we assign g as a gravity constant, ρ_1 and ρ_2 are the density of the heavier drop and lighter medium, respectively. Note that the Boussinesq approximation is effective only for small density differences. When dealing with a large ratio in density variance

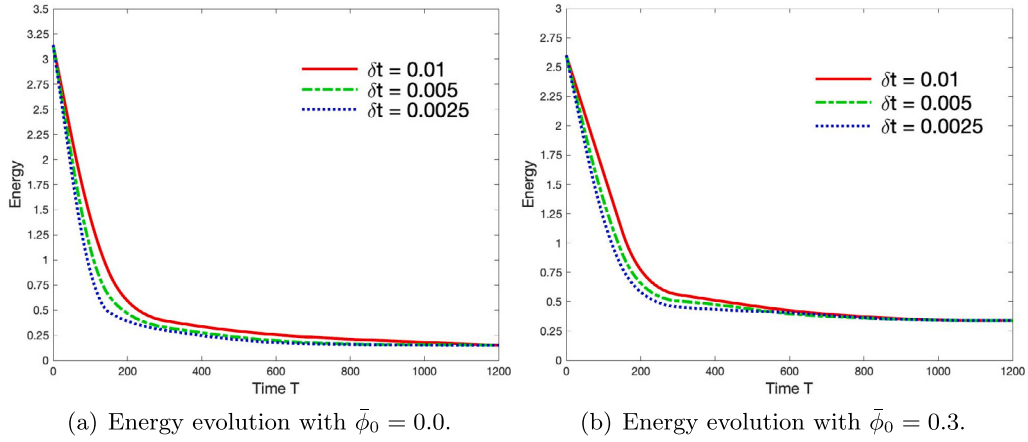


Fig. 4.3. Time evolution of the total free energy on the spherical surface S_1 with (a) $\bar{\phi}_0 = 0.0$ and (b) $\bar{\phi}_0 = 0.3$, where the chosen three time steps $\delta t = 0.0025, 0.005, 0.01$ corresponding to the blue, green and red curves, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

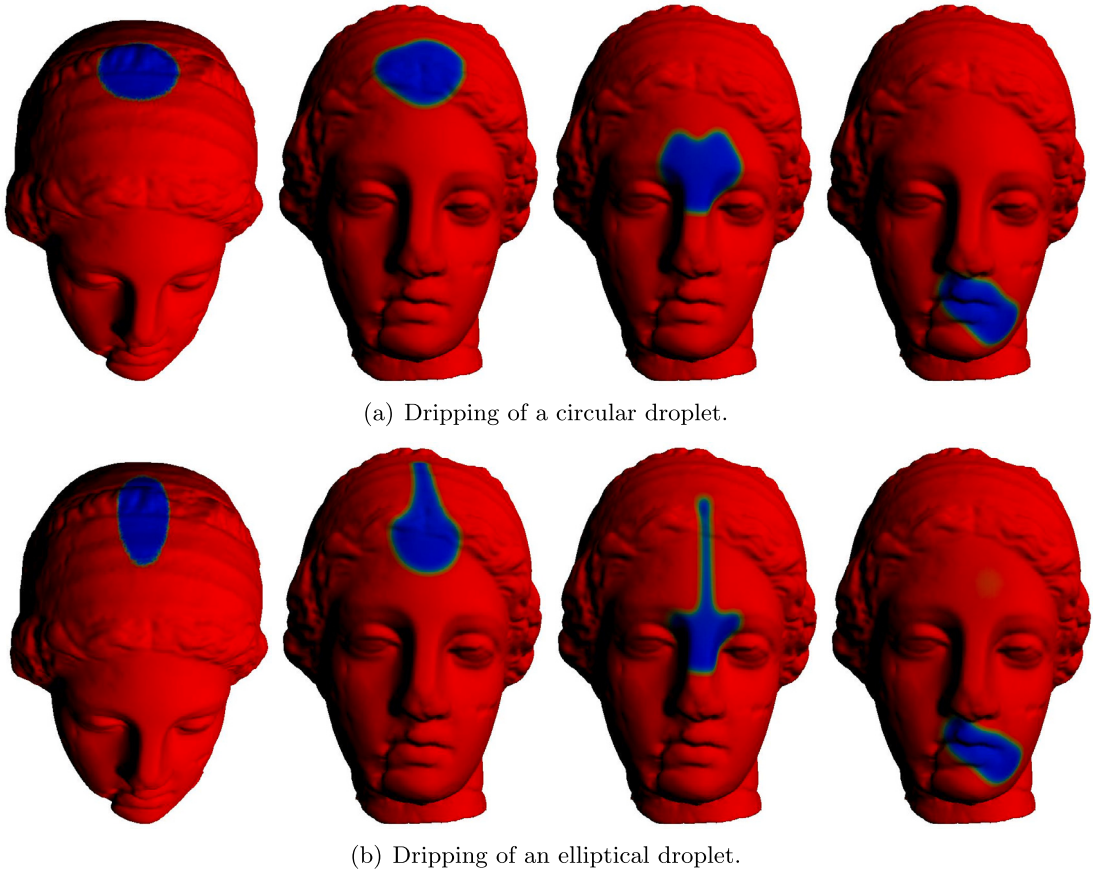


Fig. 4.4. The dynamic behaviors of the dripping drop under the gravitational force, where (a) circular drop and (b) elliptical drop. Snapshots of ϕ are plotted at $t = 0, 1.5, 2.7$, and 4.65 , respectively.

between two phases, a different model must be employed, and the corresponding algorithms need to be redesigned due to the presence of distinct nonlinear terms, see [41,55].

We simulate the droplet's dripping behavior under the influence of gravity on the complex surface domain of the human head. We denote it as S_2 , which is embedded in $\Omega = (-0.346, 0.346) \times (-0.497, 0.497) \times (-0.495, 0.495)$. The total numbers of patches/vertices

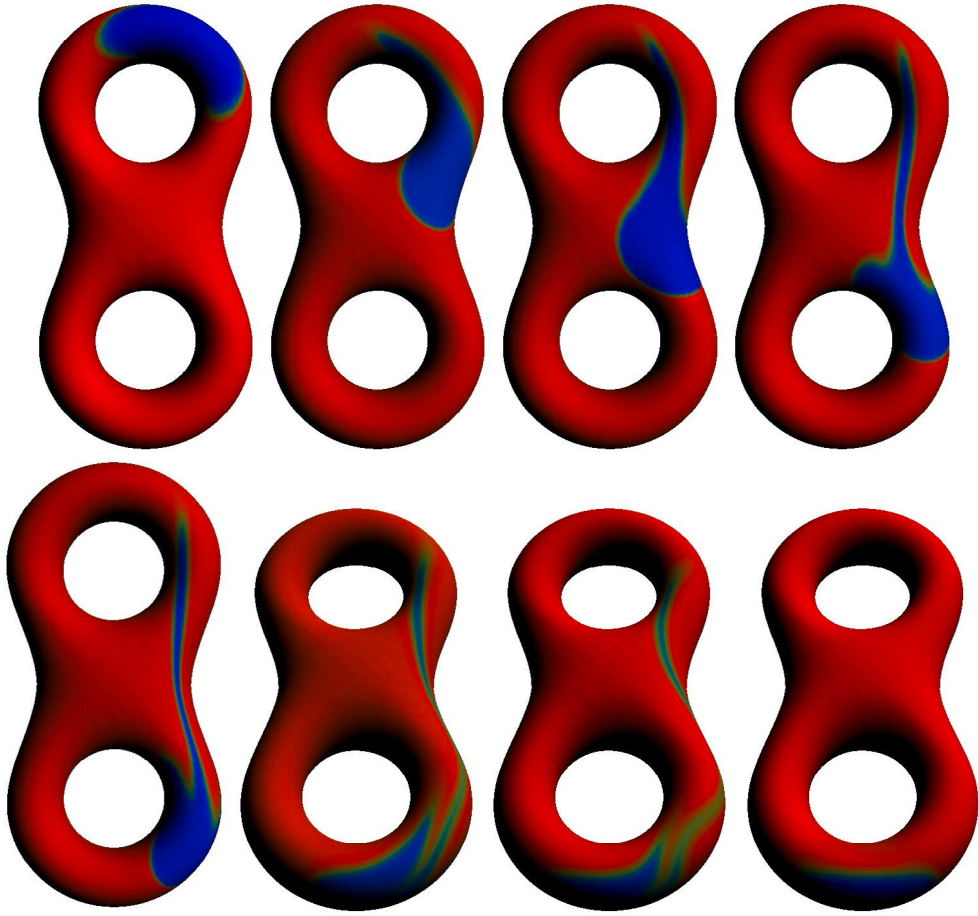


Fig. 4.5. The dripping dynamics of a droplet on the splayed surface under the gravitational force. Snapshots of the variable ϕ are taken at $t = 0, 0.4, 0.6, 0.8, 1, 1.4, 4$ and 8 , respectively.

for the surface model is 268 686/134 345, and the span of the vertex valence is 4 to 12. Two droplets, one initially circular and the other elliptical, are simulated to investigate how the dripping process evolves.

The initial conditions for these two drops are set as follows:

$$\phi_0 = \tanh\left(\frac{r_0 - \sqrt{2.4 * (x - 0.084)^2 + 2.4 * (y + 0.497)^2 + 2.4 * (z - 0.143)^2}}{\epsilon}\right), \quad (4.8)$$

and

$$\phi_0 = \tanh\left(\frac{r_0 - \sqrt{6 * (x - 0.084)^2 + (y + 0.497)^2 + (z - 0.143)^2}}{\epsilon}\right), \quad (4.9)$$

where $r_0 = 0.14$, and $\mathbf{u}^0 = \mathbf{0}$, $p^0 = 0$. The model parameters are set as

$$\begin{cases} M = 1e-4, \lambda = 4e-5, S = 4, B = 1e5, \nu = 1, \\ \epsilon = 8e-3, \rho_1 = 1.0, \rho_2 = 1.5, g = 100, \delta t = 2e-3. \end{cases} \quad (4.10)$$

In Fig. 4.4(a) and (b), we plot snapshots of the variable ϕ for two initially differently shaped droplets at different times. It can be seen that the initially circular droplet essentially descends as a cohesive whole, whereas the initially elliptical droplet forms a more elongated filament. This observation suggests that the initial shape of the droplet can indeed have a notable impact on the dynamics of its dripping process on the surface.

We denote another surface as S_3 , which is a splayed surface, and embedded in $\Omega = (-0.61, 0.61) \times (-1.27, 1.27) \times (-0.26, 0.26)$. The total numbers of patches/vertices for the surface model is 98 304/49 150, and the span of the vertex valence is 4 to 8.

We set two initial conditions for this model, one of which reads as:

$$\phi_0 = \tanh\left(\frac{0.82 - \sqrt{(x - 0.255)^2 + (y - 1.097)^2 + (z - 0.842)^2}}{\epsilon}\right), \quad \mathbf{u}^0 = \mathbf{0}, \quad p^0 = 0, \quad (4.11)$$

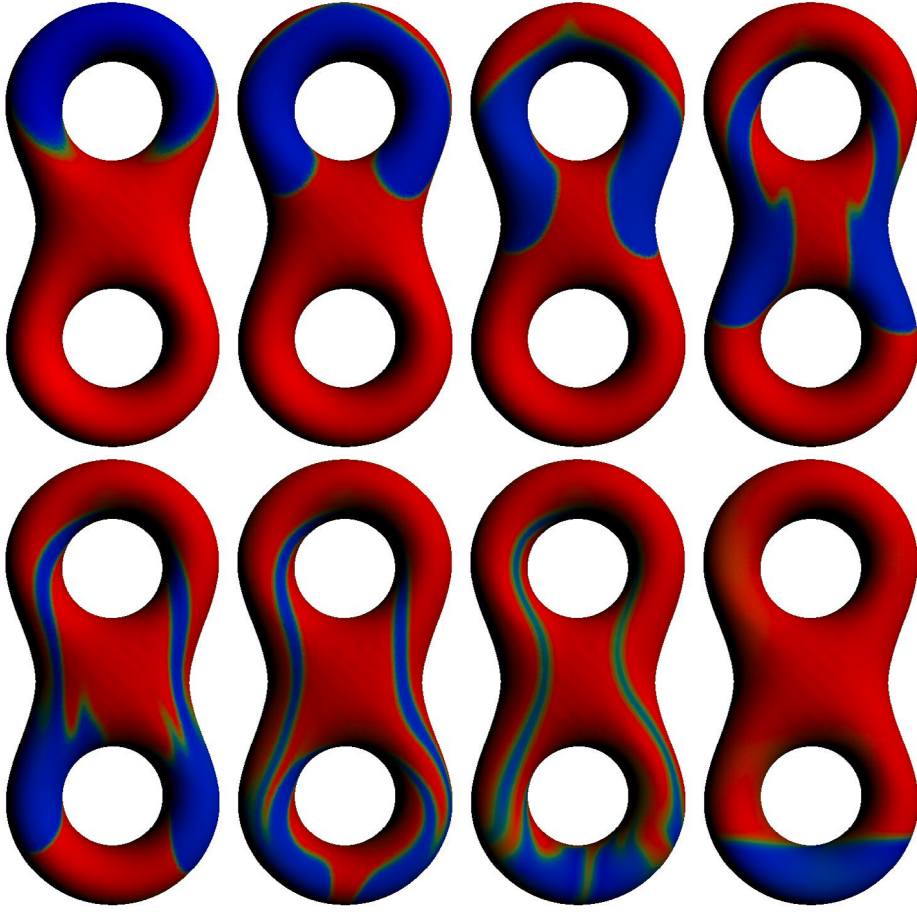


Fig. 4.6. The dripping dynamics of a droplet on the splayed surface under the gravitational force. Snapshots of the variable ϕ are taken at $t = 0, 0.4, 0.8, 1, 1.6, 2, 5, \text{ and } 9$, respectively.

and the other reads as:

$$\phi_0 = \tanh\left(\frac{1.0 - \sqrt{(x-0.0)^2 + (y-1.12)^2 + (z-0.85)^2}}{\epsilon}\right), \quad \mathbf{u}^0 = \mathbf{0}, \quad p^0 = 0. \quad (4.12)$$

The model parameters are set as

$$\begin{cases} M = 1.1e-3, \lambda = 2e-4, S = 4, B = 1e5, \nu = 0.1, \\ \epsilon = 1.8e-2, \rho_1 = 1.0, \rho_2 = 1.5, g = 100, \delta t = 2e-3. \end{cases} \quad (4.13)$$

In Figs. 4.5 and 4.6, snapshots of ϕ at different times are plotted. For these two scenarios, we observe distinct droplet dripping dynamics influenced by both the geometric shape of the surface and the initial position of the droplet. In the first scenario, at the initial moment, the droplet is positioned to the upper right of the surface of the first ring. In contrast, in the second case, the droplet starts right in the middle of the first ring. For the first case, the droplet consistently descends along the right side, while in the second case, it falls along both sides of the ring. During the descent in both cases, the center portion of the droplet undergoes significant elongation, forming a long filament. However, in the second case, the droplet falling along both sides is pinched off into two parts by gravity. Upon reaching the second ring, the two separated parts rejoin and accumulate at the bottom of the surface, resulting in a final fluid interface appearing perfectly horizontal in both cases.

5. Conclusions

In this work, we solve the hydrodynamically coupled Cahn–Hilliard phase-field model for a two-phase fluid flow system on complex curved surfaces. We conduct the spatial discretization using the IGA methodology, specifically employing the Loop subdivision technique. This technique possesses high smoothness and hierarchical refinement properties, enabling accurate representation of complex surfaces with arbitrary topologies. We integrate the EIEQ method, the ZEC method, and the projection method to discretize

the time evolution of the system. This combination results in a full discretization with desired properties, including geometric accuracy in the domain of interest, linearity, second-order time accuracy, full decoupling, and unconditional energy stability. We also provide strict proof for the unconditional energy stability and present several examples of two-phase fluid flow dynamics on complex surface models.

CRediT authorship contribution statement

Qing Pan: Writing – review & editing, Writing – original draft, Resources, Methodology, Formal analysis, Conceptualization. **Yunqing Huang:** Writing – review & editing, Investigation. **Timon Rabczuk:** Writing – review & editing. **Xiaofeng Yang:** Writing – review & editing, Writing – original draft.

Declaration of competing interest

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

Data availability

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

Acknowledgments

The work of Q. Pan was partially supported by the National Natural Science Foundation of China with grant number 12171147. The work of Y. Huang was partially supported by the National Key R&D Program of China (No. 2020YFA0713500). The work of X. Yang was partially supported by the National Science Foundation of USA with grant number DMS-2012490 and DMS-2309731.

References

- [1] J. Cahn, J. Hilliard, Free energy of a nonuniform system, I. interfacial free energy, *J. Chem. Phys.* 28 (1958) 258–267.
- [2] Z. Feng, L. Leal, Nonlinear bubble dynamics, *Annu. Rev. Fluid Mech.* 29 (1) (1997) 201–243.
- [3] H. Kellay, Hydrodynamics experiments with soap films and soap bubbles: a short review of recent experiments, *Phys. Fluids* 29 (11) (2017) 111113.
- [4] S. Kim, S. Karrila, *Microhydrodynamics: Principles and Selected Applications*, Butterworth-Heinemann, 1991.
- [5] U. Kornek, F. Mauller, K. Harth, A. Hahn, S. Ganesan, L. Tobiska, R. Stannarius, Oscillations of soap bubbles, *New J. Phys.* 12 (7) (2010) 073031.
- [6] A. Honerkamp-Smith, F. Woodhouse, V. Kantsler, R. Goldstein, Membrane viscosity determined from shear-driven flow in giant vesicles, *Phys. Rev. Lett.* 111 (3) (2013) 038103.
- [7] T. Powers, G. Huber, R. Goldstein, Fluid-membrane tethers: minimal surfaces and elastic boundary layers, *Phys. Rev. E* 65 (4) (2002) 041901.
- [8] F. Quemener, J. Sigurdsson, M. Renner, P. Atzberger, P. Bassereau, D. Lacoste, Shape matters in protein mobility within membranes, *Proc. Natl. Acad. Sci.* 111 (14) (2014) 5083–5087.
- [9] P. Saffman, M. Delbrück, Brownian motion in biological membranes, *Proc. Natl. Acad. Sci. USA* 72 (1975) 3111–3113.
- [10] J. Sigurdsson, P. Atzberger, Hydrodynamic coupling of particle inclusions embedded in curved lipid bilayer membranes, *Soft Matter* 12 (32) (2016) 6685–6707.
- [11] F. Bresme, M. Oettel, Nanoparticles at fluid interfaces, *J. Phys.: Condens. Matter* 19 (41) (2007) 413101.
- [12] M. Cavallaro, L. Botto, E. Lewandowski, M. Wang, K. Stebe, Curvature-driven capillary migration and assembly of rod-like particles, *Proc. Natl. Acad. Sci. USA* 108 (52) (2011) 20923–20928.
- [13] S. Choi, S. Steltenkamp, J. Zasadański, T. Squires, Active microrheology and simultaneous visualization of sheared phospholipid monolayers, *Nature Commun.* 2 (2011) 312.
- [14] D. Ershov, J. Sprakel, J. Appel, M. Stuart, J. Gucht, Capillarity-induced ordering of spherical colloids on an interface with anisotropic curvature, *Proc. Natl. Acad. Sci.* 110 (23) (2013) 9220–9224.
- [15] S. Panzuela, Q. Xiao, R. Delgado-Buscalioni, A. Donev, R. P. Aze, F. Usabiaga, Hydrodynamic fluctuations in quasi-two dimensional diffusion, *J. Stat. Mech.* 2018 (2018) 063207.
- [16] J. Yang, J. Kim, A phase-field model and its efficient numerical method for two-phase flows on arbitrarily curved surfaces in 3D space, *Comput. Methods Appl. Mech. Engrg.* 372 (2020) 113382.
- [17] T. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement, *Comput. Methods Appl. Mech. Engrg.* 194 (2005) 4135–4195.
- [18] Y. Bazilevs, L. Beirão da Veiga, J. Cottrell, T. Hughes, G. Sangalli, Isogeometric analysis: approximation, stability and error estimates for h-refined meshes, *Math. Models Methods Appl. Sci.* 16 (7) (2006) 1031–1090.
- [19] E. Catmull, J. Clark, Recursively generated B-spline surfaces on arbitrary topological meshes, *Comput.-Aided Des.* 10 (6) (1978) 350–355.
- [20] C. Loop, *Smooth Subdivision Surfaces Based on Triangles* (Master's thesis), Department of Mathematics, University of Utah, 1978.
- [21] J. Stam, Fast evaluation of loop triangular subdivision surfaces at arbitrary parameter values, in: *SIGGRAPH '98 Proceedings*, CD-ROM Supplement, 1998.
- [22] F. Cirak, M. Ortiz, P. Schröder, Subdivision surfaces: a new paradigm for thin-shell finite-element analysis, *Int. J. Numer. Methods Eng.* 47 (2000) 2039–2072.
- [23] F. Cirak, M. Scott, E. Antonsson, M. Ortiz, P. Schröder, Integrated modeling, finite-element analysis, and engineering design for thin-shell structures using subdivision, *Comput. Aided Des.* 34 (2) (2002) 137–148.
- [24] Q. Pan, T. Rabczuk, G. Xu, C. Chen, Isogeometric analysis for surface PDEs with extended loop subdivision, *J. Comput. Phys.* 398 (2019) 108892.
- [25] Q. Pan, T. Rabczuk, X. Yang, Subdivision-based isogeometric analysis for second order partial differential equations on surfaces, *Comput. Mech.* 68 (2021) 1205–1221.
- [26] Q. Pan, T. Rabczuk, C. Chen, Subdivision based isogeometric analysis for geometric flows, *Internat. J. Numer. Methods Engrg.* 123 (2022) 610–633.
- [27] X. Feng, A. Prohl, Numerical analysis of the Allen-Cahn equation and approximation for mean curvature flows, *Numer. Math.* 94 (2003) 33–65.
- [28] J. Shen, X. Yang, Numerical approximations of Allen-Cahn and Cahn-Hilliard equations, *Discrete Contin. Dyn. Syst. A* 28 (2010) 1669–1691.

- [29] D. Han, X. Wang, A second order in time, uniquely solvable, unconditionally stable numerical scheme for Cahn-Hilliard-Navier-Stokes equation, *J. Comput. Phys.* 290 (2015) 139–156.
- [30] D. Eyre, Unconditionally gradient stable time marching the Cahn-Hilliard equation, in: *Computational and Mathematical Models of Microstructural Evolution* (San Francisco, CA, 1998), in: *Mater. Res. Soc. Sympos. Proc.*, vol. 529, MRS, 1998, pp. 39–46.
- [31] J. Shen, C. Wang, S. Wang, X. Wang, Second-order convex splitting schemes for gradient flows with ehrlich-schwoebel type energy: application to thin film epitaxy, *SIAM J. Numer. Anal.* 50 (2012) 105–125.
- [32] Q. Pan, C. Chen, Y. Zhang, X. Yang, A novel hybrid IGA-EIEQ numerical method for the Allen-Cahn/Cahn-Hilliard equation on complex curved surfaces, *Comput. Methods Appl. Mech. Engrg.* 404 (2023) 115767.
- [33] Q. Pan, C. Chen, T. Rabczuk, J. Zhang, X. Yang, The subdivision-based IGA-EIEQ numerical scheme for the binary surfactant Cahn-Hilliard phase-field model on complex curved surfaces, *Comput. Methods Appl. Mech. Engrg.* 406 (2023) 115905.
- [34] X. Yang, Linear, first and second order and unconditionally energy stable numerical schemes for the phase field model of homopolymer blends, *J. Comput. Phys.* 327 (2016) 294–316.
- [35] X. Yang, Numerical approximations of the Navier-Stokes equation coupled with volume-conserved multi-phase-field vesicles system: Fully-decoupled, linear, unconditionally energy stable and second-order time-accurate numerical scheme, *Comput. Methods Appl. Mech. Engrg.* 375 (2021) 113600.
- [36] C. Chen, J. Zhang, X. Yang, Efficient numerical scheme for a new hydrodynamically-coupled conserved Allen-Cahn type Ohta-Kawaski phase-field model for diblock copolymer melt, *Comput. Phys. Comm.* 256 (2020) 107418.
- [37] X. Yang, A novel fully-decoupled scheme with second-order time accuracy and unconditional energy stability for the Navier-Stokes equations coupled with mass-conserved Allen-Cahn phase-field model of two-phase incompressible flow, *Internat. J. Numer. Methods Engrg.* 122 (2021) 1283–1306.
- [38] 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.
- [39] 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) B479–B507.
- [40] 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.
- [41] X. Yang, Efficient linear, fully-decoupled and energy stable numerical scheme for a variable density and viscosity, volume-conserved, hydrodynamically coupled phase-field elastic bending energy model of lipid vesicles, *Comput. Methods Appl. Mech. Engrg.* 400 (2022) 115479.
- [42] C. Chen, X. Yang, Efficient fully-decoupled and fully-discrete explicit-IEQ numerical algorithm for the two-phase incompressible flow-coupled cahn-hilliard phase-field model, *Sci. China Math.* (2023) <http://engine.scichina.com/doi/10.1007/s11425-022-2096-x>.
- [43] X. Yang, Fully-discrete spectral-Galerkin scheme with decoupled structure and second-order time accuracy for the anisotropic phase-field dendritic crystal growth model, *Int. J. Heat Mass Transf.* 180 (2021) 121750.
- [44] J. Shen, X. Yang, Decoupled energy stable schemes for phase field models of two phase complex fluids, *SIAM J. Signal Comput.* 36 (2014) B122–B145.
- [45] P. Yue, J. Feng, C. Liu, J. Shen, A diffuse-interface method for simulating two-phase flows of complex fluids, *J. Fluid Mech.* 515 (2004) 293–317.
- [46] X. Yang, A novel decoupled second-order time marching scheme for the two-phase incompressible Navier-Stokes/Darcy coupled nonlocal Allen-Cahn model, *Comput. Methods Appl. Mech. Engrg.* 377 (2021) 113597.
- [47] 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) B479–B507.
- [48] L. Piegl, W. Tiller, *The NURBS Book*, second ed., *Monographs in Visual Communication*, Springer-Verlag, New york, 1997.
- [49] T. Sederberg, G. Finnigan, X. Li, H. Lin, H. Ipson, Watertight trimmed NURBS, *ACM Trans. Graph.* 27 (3) (2008) 1–8.
- [50] T. Sederberg, J. Zheng, A. Bakenov, A. Nasri, T-splines and T-NURCCs, *ACM Trans. Graph.* 22 (3) (2003) 477–484.
- [51] Y. Bazilevs, V. Calo, J. Cottrell, J. Evans, T. Hughes, S. Lipton, M. Scott, T. Sederberg, Isogeometric analysis using T-splines, *Comput. Methods Appl. Mech. Engrg.* 5–8 (2010) 229–263.
- [52] J. Shen, On error estimates of the projection methods for the Navier-Stokes equations: second-order schemes, *Math. Comp.* 65 (215) (1996) 1039–1065.
- [53] W. E, J. Liu, Projection method. I. Convergence and numerical boundary layers, *SIAM J. Numer.* 32 (1995) 1017–1057.
- [54] X. Yang, J. Feng, C. Liu, J. Shen, Numerical simulations of jet pinching-off and drop formation using an energetic variational phase-field method, *J. Comput. Phys.* 218 (2006) 417–428.
- [55] C. Chen, X. Yang, Highly efficient and unconditionally energy stable semi-discrete time-marching numerical scheme for the two-phase incompressible flow phase-field system with variable-density and viscosity, *Sci. China Math.* 65 (2022) 2631–2656.