Journal of Computational and Applied Mathematics 486 (2026) 117702

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

An adaptive numerical method for solving the Allen–Cahn equation on curved surfaces

Hyundong Kim [a,b], Qunzhi Jin [c], Xinpei Wu [d], Soobin Kwak [d], Zhengang Li [d], Junseok Kim [d,*]

[a] Department of Mathematics and Physics, Kangwon National University, Gangneung, 25457, Republic of Korea
[b] Institute for Smart Infrastructure, Kangwon National University, Gangneung, 25457, Republic of Korea
[c] College of Science, Yanbian University, Yanji, 133002, China
[d] Department of Mathematics, Korea University, Seoul, 02841, Republic of Korea

ARTICLE INFO

Keywords:
Surface Allen–Cahn equation
Adaptive method
Time-dependent surface domain

ABSTRACT

We develop an adaptive computational method for the Allen–Cahn (AC) equation on curved surfaces. The scheme is built on triangulated meshes using the cotangent formula for the discrete Laplace–Beltrami operator and applies an operator splitting method to decouple diffusion and reaction terms. The diffusion step is solved by an explicit Euler scheme, while the reaction step is treated analytically. An adaptive narrow-band method updates only the interface region, which makes the computation faster and keeps good accuracy. Numerical studies on simple and complex geometries verify accurate interface motion and phase separation resolution at reduced computational cost, and when compared with the global iteration algorithm, the proposed method achieves superior efficiency, which supports its suitability for phase-field models on curved surfaces.

1. Introduction

The evolution of phase interfaces or phase fields occurs not only in flat Euclidean spaces but also, more frequently, on curved surfaces or surfaces constrained by geometric boundaries, such as the outer surface of biological tissues [1,2], cell membranes [3,4], shape transformation [5], and fluid-surfactant systems [6,7]. In these scenarios, curvature [8], surface geometry [9], and their variations significantly affect interface motion, interfacial morphology, and phase-field distributions. Zamani-Gharaghoshi et al. [10] developed a mesh-free numerical method for the surface Allen–Cahn (AC) model that combines the generalized moving least-squares approximation with the closest point method and permits computations on arbitrarily distributed points independent of the surface structure. Sun and Zhang [11] proposed a meshless radial basis-function algorithm for conservative AC equations on smooth compact surfaces, which uses operator splitting, explicit time discretization, and kernel-based quadrature to ensure mass conservation and to preserve computational efficiency. Huang et al. [12] developed stabilized semi-implicit schemes with adaptive time-stepping for the binary fluid-surfactant equation. Yang et al. [13] presented an unconditionally energy-stable, second-order accurate numerical method for the generalized Swift–Hohenberg equation on surfaces. They established energy dissipation and unique solvability. Kim et al. [14] investigated a stability theory for the maximum principle preserving method for the surface AC equation on curved manifolds, with the Laplace–Beltrami operator discretized by cotangent weights and vertex areas obtained by mass lumping on a triangular mesh. A vertexwise admissible time-step bound is derived that guarantees a discrete maximum principle; the bound depends on the local mixed area at a vertex and on the cotangent-weight sum over its one-ring neighbors.

* Corresponding author.
E-mail addresses: hdkim@kangwon.ac.kr (H. Kim), cfdkim@korea.ac.kr (J. Kim).

https://doi.org/10.1016/j.cam.2026.117702
Received 29 November 2025; Received in revised form 15 March 2026
Available online 12 April 2026

To improve the efficiency and accuracy of numerical simulations of the AC equation, adaptive methods [15–17] have been proposed to concentrate computational resources near interfacial regions with sharp transitions. Xu et al. [18] incorporated an adaptive collocation strategy into physics-informed neural networks for solving the AC equation, and this modification improved robustness and accuracy in long-time simulations of nonlinear PDEs. Kim et al. [19] presented an unconditionally stable adaptive finite difference method for the AC equation, which combines operator splitting, finite difference discretization, and closed-form substeps. Li et al. [20] proposed an adaptive discontinuous finite volume element scheme for the AC equation, that achieves energy stability and accurately captures phase-transition dynamics in long-term simulations.

Phase-field models [21–23] on curved surfaces require resolution concentrated along diffuse interfaces whose width is small and shaped by curvature [24–26]. On triangular meshes, a uniform triangular mesh spreads computational effort uniformly over a problem that is intrinsically localized and anisotropic. It uses too much effort in smooth regions and too little effort near curved interfaces where errors are large, and it adds global costs and limits that are not related to the real physical behavior[27,28]. Space-adaptive methods for phase-field models [13,29–31] refine where tangential gradients steepen, interfaces move, or the second fundamental form is large, and coarsen elsewhere. For the AC and Cahn–Hilliard (CH) dynamics [32], adaptivity allows larger stable time steps, improves maximum-principle behavior and energy decay, and keeps mass conserved in bulk-surface coupling by using matched resolutions [33–35].

On such manifolds, the AC equation is a basic model for motion driven by curvature and for phase relaxation. Accurate simulation needs high spatial resolution near the thin interface of width $O(\epsilon)$, because this interface moves over time and interacts with surface curvature. Uniform meshes on triangulated surfaces use too many grid points where the solution changes little and too few where strong gradients appear near the interface. As a result, the computational cost depends on the total mesh size rather than on the interface itself. This becomes more serious in long-time simulations because the interface occupies only a small part of the surface but still controls the accuracy and stability. To resolve this problem, we use an adaptive narrow-band method for the surface AC equation. The method updates the numerical solutions only the vertices near the interface and their neighbors. It uses a cotangent-weight Laplace–Beltrami discretization and an operator splitting method. Diffusion is computed explicitly on the surface, and the nonlinear reaction is solved in closed form. This method is simple and robust. It keeps the structure of the gradient flow, allows time steps based on local mesh geometry, and reduces the computational cost because the work mainly depends on the interface length. Numerical tests on simple and complex surfaces show that the method reproduces analytic results, preserves energy decay, and greatly reduces computation time while maintaining good accuracy.

This work is organized as follows. Section 2 presents computational methods for the discretization of the Laplace–Beltrami operator and the construction of adaptive narrow-band surface domains. Section 3 reports numerical results for the surface AC computed with the proposed adaptive method. In Section 4, we conclude the study.

2. Numerical solution methods

We consider the AC equation on a closed smooth surface $S \subset \mathbb{R}^3$:

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = \frac{\phi(\mathbf{x},t) - \phi^3(\mathbf{x},t)}{\epsilon^2} + \Delta_S \phi(\mathbf{x},t), \qquad \mathbf{x} \in S,\ t > 0, \tag{1}$$

where Δ_S denotes the Laplace–Beltrami operator and $\epsilon > 0$ is a small parameter controlling the interfacial thickness.

2.1. Laplace–Beltrami operator discretization

Now, we present a discretized Laplace–Beltrami operator [36,37] on S. We define the surface vertex set $\{\mathbf{p}\}_{i=1}^{M}$ including M points. Let $V(i) = \{i_1, i_2, \dots, i_m\}$ be the one-ring neighbor surface vertex indices, with $i_1 = i_m$ for a given surface vertex $\mathbf{p}_i$, as shown in Fig. 1(a). The surface vertices $\mathbf{p}_i, \mathbf{p}_{j_-}$ and $\mathbf{p}_j$ constitute a triangle T_j. We compute $A(\mathbf{p}_i) = \{T_1, T_2, \dots, T_5, T_6\}$ in Fig. 1(a), the sum of triangles T_j in contact with the surface vertex $\mathbf{p}_i$.

Let $A(\mathbf{p}_i, T_j)$ be the area of T_j in contact with $\mathbf{p}_i$. Then, it can be expressed using $\mathbf{p}_i, \mathbf{p}_{i_-}$, and $\mathbf{p}_j$, three vertices of T_j, see Fig. 1(a), as follows:

$$A(\mathbf{p}_i, T_j) = \frac{1}{2}\left|\mathbf{p}_j - \mathbf{p}_{j_-}\right|\left|\mathbf{p}_i - \mathbf{p}_{j_-}\right| \sin\theta \tag{2}$$

$$= \frac{1}{2}\sqrt{\left|\mathbf{p}_j - \mathbf{p}_{j_-}\right|^2 \left|\mathbf{p}_i - \mathbf{p}_{j_-}\right|^2 - (\mathbf{p}_j - \mathbf{p}_{j_-}, \mathbf{p}_i - \mathbf{p}_{j_-})^2}, \tag{3}$$

where θ is the angle of vertex $\mathbf{p}_i$ and $(\mathbf{p}_j - \mathbf{p}_{j_-}, \mathbf{p}_i - \mathbf{p}_{j_-})$ is the inner product for the vectors $\mathbf{p}_j - \mathbf{p}_{j_-}$ and $\mathbf{p}_i - \mathbf{p}_{j_-}$. Therefore, $A(\mathbf{p}_i)$ can be calculated as $A(\mathbf{p}_i) = \sum_{j \in V(i)} A(\mathbf{p}_i, T_j)$. Set $\phi_i = \phi(\mathbf{p}_i)$. We derive the following discretized Laplace–Beltrami operator [38,39]:

$$\Delta_S \phi_i^n = \frac{3}{A(\mathbf{p}_i)} \sum_{j \in V(i)} \frac{\cot\theta_{ij} + \cot\theta_{ij_+}}{2} \left(\phi_j^n - \phi_i^n\right), \tag{4}$$

where θ_{ij} is the angle in triangle T_j and θ_{ij_+} is the angle in triangle T_{j_+} as displayed in Fig. 1(b).

Remark 1. The cotangent Laplace–Beltrami operator in Eq. (4) is consistent on well-shaped triangulations, but mesh quality can affect robustness. On meshes with obtuse angles or strong anisotropy, cotangent weights can take negative values, which can weaken

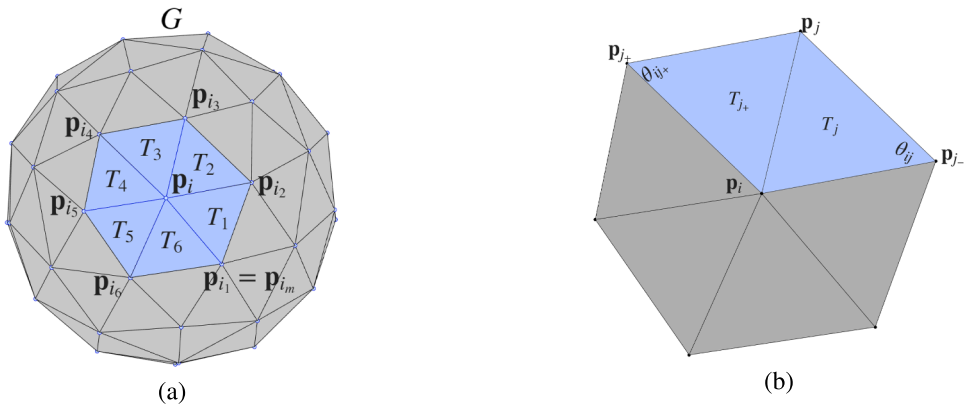


Fig. 1. Local one-ring notation on a triangulated surface. (a) Triangulated surface G with a vertex $\mathbf{p}_i$ and its one-ring neighborhood $\{\mathbf{p}_i\}_{j=1}^{m}$ ordered cyclically, with $\mathbf{p}_{i_m} = \mathbf{p}_{i_1}$. The incident triangles form the vertex fan T_j. (b) Two triangles adjacent to the edge $e_{ij} = (\mathbf{p}_i, \mathbf{p}_j)$, namely $T_j = (\mathbf{p}_i, \mathbf{p}_j, \mathbf{p}_{j_-})$ and $T_{j_+} = (\mathbf{p}_i, \mathbf{p}_{j_+}, \mathbf{p}_j)$. The angles opposite e_{ij} in T_j and T_{j_+} are denoted by θ_{ij} and θ_{ij_+}, respectively.

discrete maximum-principle behavior and can reduce robustness of explicit diffusion updates [14,40]. The adaptive narrow-band method only needs information from nearby nodes, so it can be used on general triangulated surfaces. For a surface with boundary, one can impose natural no-flux conditions by restricting stencils to interior neighbors and by defining vertex areas from the one-sided incident triangles adjacent to each boundary vertex [41–43]. The present work focuses on fixed surfaces, but the same algorithmic structure extends to time-dependent surfaces if one updates vertex positions and recomputes areas and cotangent weights at each time level [44].

2.2. Numerical method for the surface AC equation on adaptive narrow-band surface domains

The surface AC dynamics exhibits two distinct regimes: in bulk regions the order parameter rapidly relaxes to the pure states $\phi \approx \pm 1$, whereas steep tangential gradients occur only across a diffuse interface of thickness $O(\epsilon)$ near the zero level set $\{\phi = 0\}$. A global update over all mesh vertices therefore expends most computational effort in regions where ϕ remains nearly constant. To align the numerical work with the localized physics of interfacial motion, we restrict the update to a thin tubular neighborhood of the interface that we identify at each time level.

Let $t^n = n\Delta t$ and $\phi_i^n \approx \phi(\mathbf{p}_i, t^n)$ denote the phase-field value at surface vertex $\mathbf{p}_i$. For a prescribed threshold $\tau \in (0, 1)$, we define the narrow-band (active) vertex set

$$S^n = \left\{ \mathbf{p}_i \in S \mid |\phi_i^n| < \tau \right\}. \tag{5}$$

A theoretical interpretation of the threshold τ follows from the standard diffuse-interface profile for the AC model. Near the interface, the phase-field admits the one-dimensional approximation $\phi(d) \approx \tanh\left(d/(\sqrt{2}\epsilon)\right)$, where d is the signed geodesic distance to the zero level set $\{\phi = 0\}$. The level $|\phi| = \tau$ therefore corresponds to the distance $|d| = w(\tau)$, where $w(\tau) = \sqrt{2}\epsilon \tanh^{-1}(\tau)$. Hence, the set S^n approximates a tubular neighborhood of the interface with half-width $w(\tau) = O(\epsilon)$. This relation clarifies the effect of τ: larger τ yields a wider band and higher cost, whereas smaller τ yields a thinner band and can truncate the diffuse interface. A simple error bound also supports this choice. For $\tau \geq 1/\sqrt{3}$ and $|\phi| \geq \tau$, the reaction term satisfies $|\phi - \phi^3| \leq \tau(1 - \tau^2)$, so omission of the update outside the band introduces a per-step local error of order $\Delta t \tau(1 - \tau^2)/\epsilon^3$, which decreases as $\tau \to 1$.

In addition, we expand the active set by one-ring neighbors. This step adds an extra buffer of about one mesh layer and improves robustness when curvature and stiffness increase, for example near extinction or near discrete topological events. A practical choice rule follows from $w(\tau)$. On quasi-uniform meshes, we set a target half-width $w \approx k h_{geo}$ with $k \geq 2$ and then set $\tau = \tanh\left(w/(\sqrt{2}\epsilon)\right)$. The level sets $\phi = \tau$ and $\phi = -\tau$ form the boundaries of this band on the surface. Fig. 2(a) visualizes ϕ on the sphere together with these two contour lines (magenta); the region between them represents the narrow band $\{|\phi| < \tau\}$ that encloses the diffuse interface. The discrete Laplace–Beltrami operator in Eq. (4) requires a complete one-ring stencil. For any active vertex $\mathbf{p}_i \in S^n$, the cotangent formula involves its one-ring neighbors $V(i)$. Near the boundary of S^n, some neighbors that enter the stencil can lie outside S^n. To avoid stencil truncation and to preserve the same local operator as in the global scheme, we augment the narrow band by a one-ring buffer set

$$B^n = \left(\bigcup_{\mathbf{p}_i \in S^n} V(i) \right) \Bigg\backslash S^n, \quad S_h^n = S^n \bigcup B^n. \tag{6}$$

Fig. 2(b) highlights this discrete construction: vertices in S^n (narrow-band points) appear in black, whereas vertices in B^n (buffer points) appear in red. The union S_h^n provides the one-ring neighbor information needed to compute the cotangent Laplace–Beltrami stencil for each $\mathbf{p}_i \in S^n$.

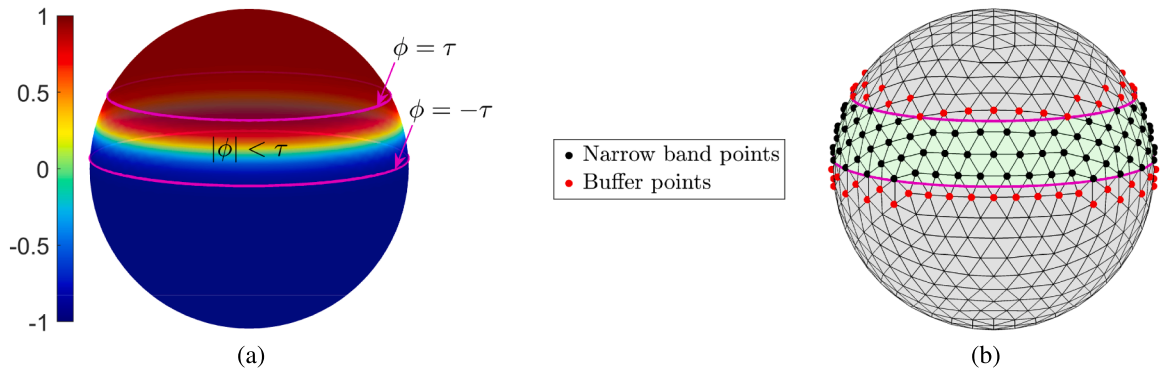


Fig. 2. Adaptive narrow-band update region on a triangulated sphere. (a) Surface phase-field ϕ. The level sets $\phi = \tau$ and $\phi = -\tau$ (magenta) delimit the narrow band $\{|\phi| < \tau\}$, which encloses the diffuse interface near $\phi = 0$. (b) Discrete vertex sets at a fixed time level n: narrow-band vertices $S^n = \{\mathbf{p}_i \in S : |\phi_i^n| < \tau\}$ (black) and buffer vertices $B^n = \left(\bigcup_{\mathbf{p}_i \in S^n} V(i)\right) \setminus S^n$ (red), where $V(i)$ denotes the one-ring neighbor set of $\mathbf{p}_i$. The adaptive computational domain $S_h^n = S^n \bigcup B^n$ supplies complete one-ring stencils for the cotangent Laplace–Beltrami operator, while updates of ϕ are confined to S^n.

To solve the modified surface AC Eq. (1), we apply the operator splitting method by decomposing the equation into two subproblems:

$$\frac{\partial \psi(\mathbf{x},t)}{\partial t} = \Delta_S \psi(\mathbf{x},t), \tag{7}$$

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = \frac{\phi(\mathbf{x},t) - \phi^3(\mathbf{x},t)}{\epsilon^2}. \tag{8}$$

Here, ψ and ϕ represent the solutions for the diffusion and reaction subproblems, respectively. The discretization is carried out in two steps. Firstly, we use the explicit Euler scheme to solve the diffusion Eq. (7) in the narrow band domain S^n:

$$\frac{\psi_i^* - \psi_i^n}{\Delta t} = \frac{3}{A(\mathbf{p}_i)} \sum_{j \in V(i)} \frac{\cot\theta_{ij} + \cot\theta_{ij_+}}{2} \left(\psi_j^n - \psi_i^n\right), \tag{9}$$

where $A(\mathbf{p}_i)$ is the area of the surface mesh around vertex $\mathbf{p}_i$, and θ_{ij} and θ_{ij+} are the angles in the triangles formed by the neighboring vertices. The term $V(i)$ denotes the set of one-ring neighbors for vertex $\mathbf{p}_i$, and the computation is carried out only for the vertices in the narrow band region defined by S_h^n. Secondly, we apply the separation of variables to solve the reaction Eq. (8):

$$\phi_i^{n+1} = \frac{\psi_i^*}{\sqrt{\left(1 - \left(\psi_i^*\right)^2\right) e^{-\frac{2\Delta t}{\epsilon^2}} + \left(\psi_i^*\right)^2}}. \tag{10}$$

At each time step, we perform the adaptive update on this set. First, we find S^n from ϕ^n. Then, we build B^n and S_h^n using one-ring neighbors. Second, for each $\mathbf{p}_i \in S^n$, we compute the diffusion step in Eq. (9). The summation over $V(i)$ uses values from S_h^n. Third, we apply the reaction formula in Eq. (10) and obtain ϕ_i^{n+1} for $\mathbf{p}_i \in S^n$. For vertices outside the narrow band, we set $\phi_i^{n+1} = \phi_i^n$. This step keeps bulk values unchanged and updates only the interface region where the solution changes quickly. After the time step, we compute S^{n+1} from ϕ^{n+1}. In this way, the adaptive region moves with the interface on the surface.

In conclusion, the diffusion and reaction steps give an efficient solution of the adaptive surface AC equation. The method updates values only near the interface. This approach reduces computational cost and focuses the computation on the interface region, which improves both efficiency and accuracy.

3. Computational tests

This section shows a systematic test of the proposed adaptive solver for the surface AC equation on triangulated manifolds. Our aims are threefold: (i) to verify accuracy on curved geometries using problems that admit analytic benchmarks; (ii) to confirm structural properties of the discretization, including monotone decay of the discrete Ginzburg–Landau energy, boundedness of the order parameter, and curvature-driven contractive dynamics; and (iii) to quantify the efficiency gains of the narrow-band update relative to a global computation, with a focus on how the cost scales with interfacial length rather than with the total mesh size. Unless otherwise specified, all numerical experiments were carried out on a 64-bit Windows workstation equipped with an Intel Core i7-13700 (13th generation, 2.10 GHz), 64GB DDR5-5600 RAM, a discrete GPU with 16GB of device memory.

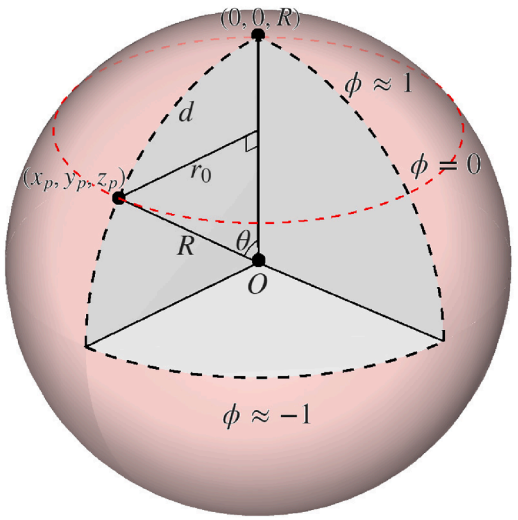


Fig. 3. Schematic of parameters upon the sphere where ϕ evolves.

3.1. Motion by mean curvature on a sphere

We conduct a series of computational tests to illustrate the performance of the proposed method on spherical domains in three dimensions. Let us consider a sphere of radius R centered at $(0, 0, 0)$. Let (x_p, y_p, z_p) denote an arbitrary point on the sphere, see Fig. 3.

We study the motion of a circular interface on a sphere. The interface evolves according to the mean-curvature flow defined on the surface [45,46]. Let $r(t)$ be the geodesic circle radius at time t. Using the geometric decomposition of the velocity into tangential and inward normal parts with the circle curvature $-1/r$, one obtains the ordinary differential equation as

$$\frac{dr}{dt} = \frac{r^2 - R^2}{rR^2}. \tag{11}$$

Integrating yields the closed-form evolution of the interface radius

$$r(t) = \sqrt{R^2 - \left(R^2 - r_0^2\right)e^{2t/R^2}}, \tag{12}$$

where $r_0 = r(0)$. To validate this law on a triangulated sphere, we initialize the phase-field by

$$\phi(x, y, z, 0) = \tanh\left(\frac{R_0(\frac{\pi}{3} - \arccos(z/R_0))}{\sqrt{2}\epsilon}\right). \tag{13}$$

The parameters are chosen as $R_0 = 7$, $h = h_{\text{geo}}$ and $\epsilon = \epsilon_m = mh_{\text{geo}}/2\sqrt{2}\text{arctanh}(0.9)$, $m = 12$, with time step size $\Delta t = 0.001$. For a detailed definition of geodesic h_{geo}, please refer to Kim et al. [14].

Fig. 4(a)–(h) display snapshots at $t = \Delta t$, $10000\Delta t$, $20000\Delta t$, and $30000\Delta t$ for the global and adaptive runs, respectively. The interface location and its regularity agree to graphical accuracy at all sampling times; no spurious ripples or delay of the front appear in the adaptive solution. The adaptive band remains localized near the interfacial region and does not introduce artifacts away from it.

Fig. 5 shows the evolution of the spherical-cap radius $r(t)$ from the initial configuration to extinction. The solid black curve shows the analytic solution. Blue circles and red stars show the adaptive and global numerical results on the triangulated surface. For most of the interval $[0, T)$, the numerical results agree well with the analytic curve. This numerical result shows that the method captures the correct shrinking motion on the sphere. The radius decreases steadily and the interface remains convex. A small difference appears only near the final time $T = 34000\Delta t$, where the curvature becomes large. The adaptive result stays closer to the analytic curve than the global result. These results show that the adaptive method gives good accuracy without using a very fine global mesh.

Remark 2. In Fig. 5, the adaptive result shows a slightly smaller radius error near extinction than the global result. Away from the interface, the AC dynamics quickly drives ϕ to the stable states $\phi = \pm 1$, so the solution in bulk regions stays almost constant. In the global method, the diffusion update is applied at all vertices. Small numerical errors in bulk regions can spread and grow over many time steps. In the narrow-band method, bulk vertices are not updated and changes occur only near the interface. This reduces numerical noise in the bulk and keeps the stable phases. Near extinction, where curvature becomes large, this leads to slightly better accuracy.

3.2. Sensitivity test for τ

We test how the adaptive method changes with the parameter τ on triangular curved surfaces. Fig. 6 shows that $\tau = 0.80$, 0.90, and 0.99 give similar results. The initial polar cap shrinks over time. The interface stays close to a zonal curve, and the upper phase

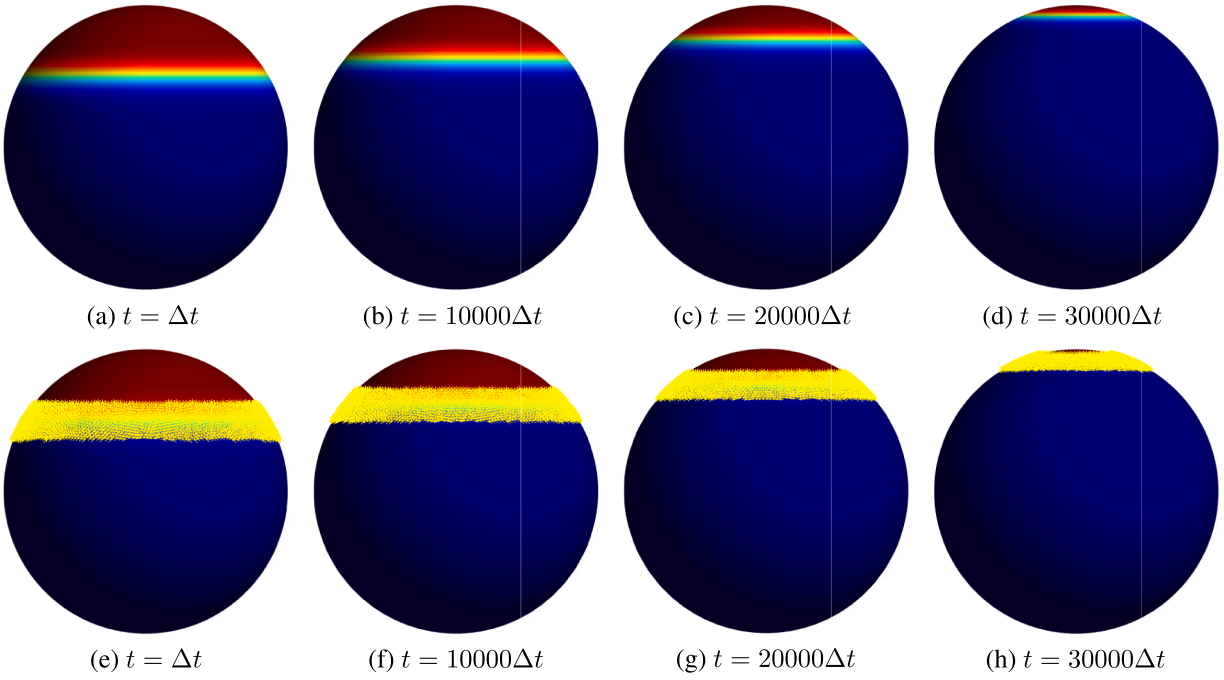


Fig. 4. Global computation (a)–(d), and adaptive computation (e)–(h) for the spherical cap test.

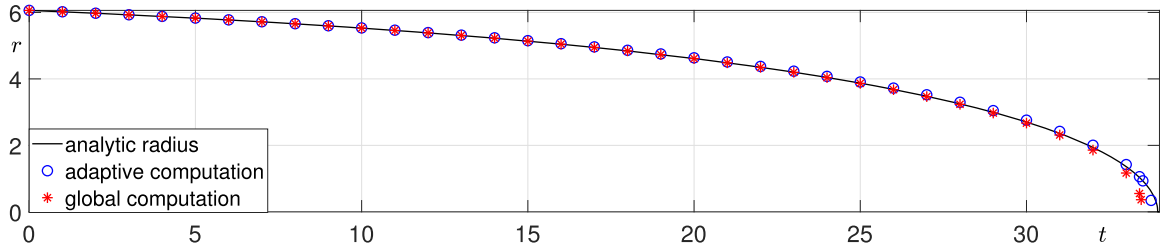


Fig. 5. Evolution of the radius $r(t)$ for the adaptive and global computations. The solid line shows the analytic radius, blue circles show the adaptive result, and red stars show the global result.

almost disappears at late time. This behavior matches curvature-driven relaxation in the surface AC model. The interface also stays nearly axisymmetric during the simulation, and no clear mesh anisotropy or oscillations appear. These numerical results show that the proposed method works well on curved triangular meshes.

Fig. 7 shows clearer differences among the three values of τ. For $\tau = 0.80$, the numerical radius begins to differ from the analytic radius at later times, and the difference becomes larger near extinction. This happens because the active band is too narrow. For $\tau = 0.90$, the numerical radius matches the analytic curve very well over the whole time interval. For $\tau = 0.99$, the result is also close to the analytic curve, but the improvement over $\tau = 0.90$ is very small and a slight difference appears at late time. Therefore, $\tau = 0.90$ gives the best agreement with the analytic solution.

Fig. 8 explains this trend using adaptive localization. In all cases, the number of active points N_p^n decreases over time. This happens because the interface becomes smaller as the cap radius shrinks. A larger τ activates a wider region near the interface and therefore uses more active points. In particular, $\tau = 0.99$ uses many more active points than the other cases, while $\tau = 0.80$ uses the fewest points. However, using too few points causes lower accuracy near extinction. The case $\tau = 0.90$ gives a better balance. It uses fewer active points than $\tau = 0.99$; however, the numerical radius still matches the analytic result well.

Now, let us define the normalized degrees of freedom $t_{dof} := T_{cpu} \Big/ \sum_{n=0}^{N-1} N_p^n$, which is total CPU time T_{cpu} per active-vertex update N_p^n. The efficiency indicator t_{dof} in Fig. 9 leads to the same conclusion. The smallest value occurs at $\tau = 0.90$, whereas $\tau = 0.80$ yields a slightly larger value and $\tau = 0.99$ produces a substantial increase. Therefore, the most conservative choice of τ does not provide a commensurate gain in accuracy, while the most aggressive reduction of the active region undermines the late-time approximation. The numerical evidence identifies $\tau = 0.90$ as the most effective parameter in the present adaptive method. It offers the best compromise

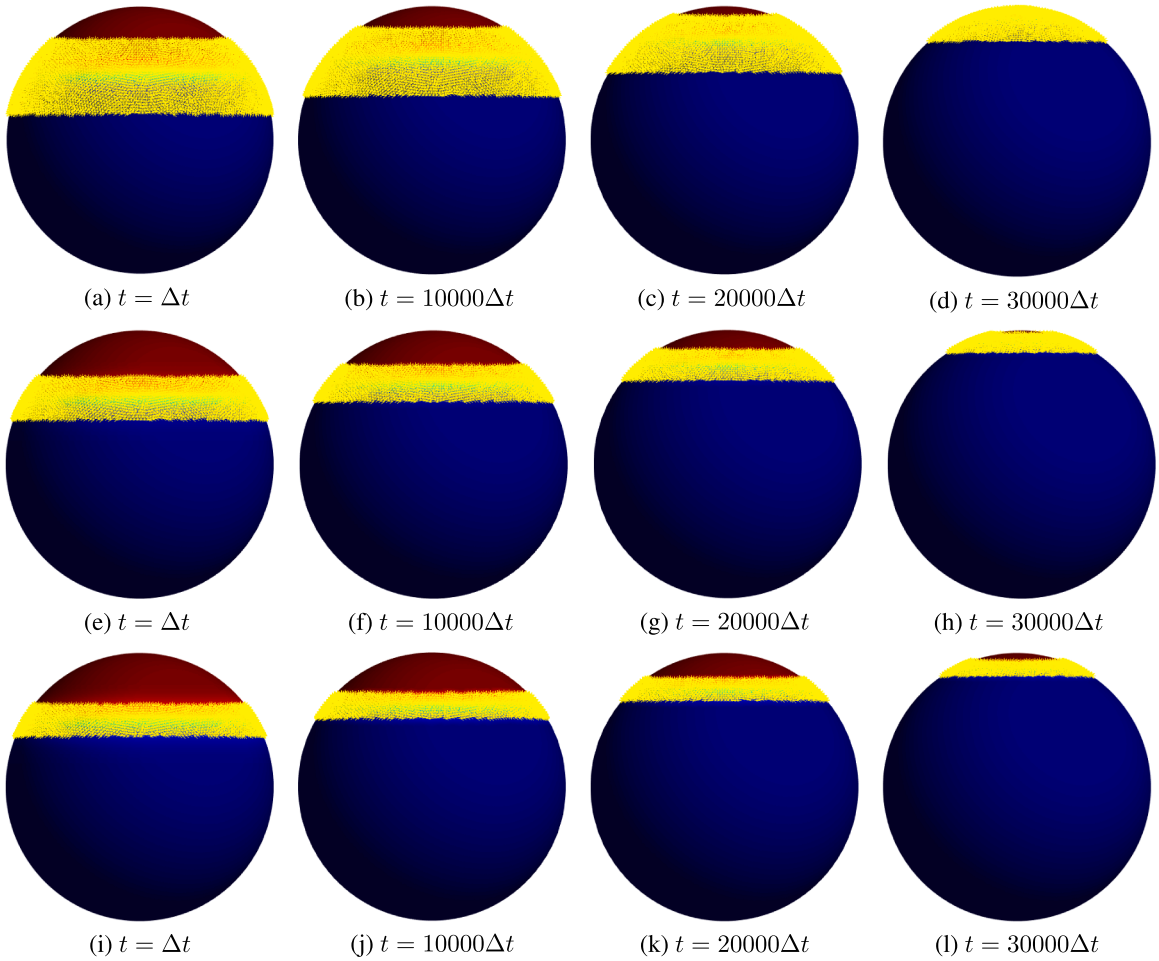


Fig. 6. Sensitivity test for (a)–(d) $\tau = 0.99$, (e)-(h) $\tau = 0.90$ and (i)–(l) $\tau = 0.80$.

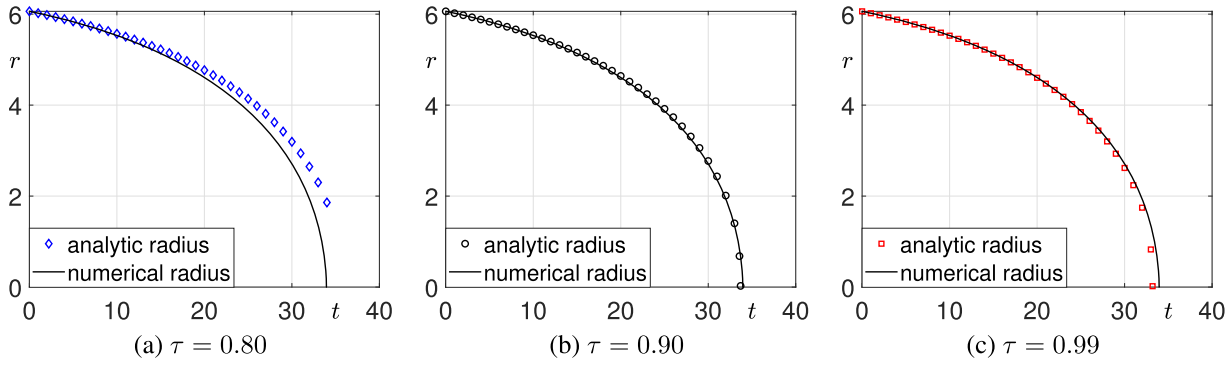


Fig. 7. Analytic and numerical radii for (a) $\tau = 0.80$, (b) $\tau = 0.90$, and (c) $\tau = 0.99$.

between interface accuracy, adaptive compression, and computational efficiency, and thus serves as the most suitable default choice for the surface AC model on triangular curved surfaces.

Table 1 reports the normalized computational cost t_{dof} for the global and adaptive approaches. The results indicate a substantial reduction in computational cost when the adaptive strategy is used. In the global computation, the measured value is $t_{dof} = 5.9847e - 02$, whereas the adaptive computation with $\tau = 0.90$ case produces $t_{dof} = 1.1136e - 05$. The ratio between the two values reaches approximately $5.3741e + 03$. This difference indicates that the adaptive solver requires more than three orders of magnitude less computational effort per effective degree of freedom update than the global solver. This improvement originates from the localization

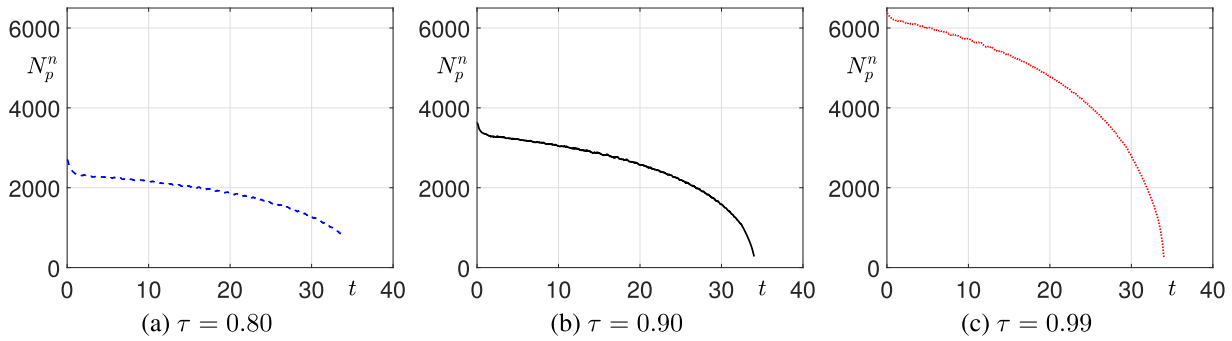


Fig. 8. Temporal evolution of number of active points N_p^n for (a) $\tau = 0.80$, (b) $\tau = 0.90$, and (c) $\tau = 0.99$.

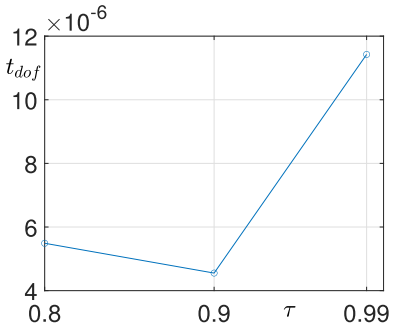


Fig. 9. Degrees of freedom t_{dof} for N_p^n for $\tau = 0.80$, $\tau = 0.90$, and $\tau = 0.99$.

Table 1
Comparison of the normalized computational cost t_{dof} (in seconds) between the global and adaptive computations.

	(A) Global computation	(B) Adaptive computation	Ratio (A/B)
t_{dof} (s)	5.9847e-02	1.1136e-05	5.3741e+03

of the computation around the diffuse interface. The global solver performs updates over all vertices of the surface mesh at every time step, regardless of whether they contribute to the interface dynamics. In contrast, the adaptive method restricts the updates to the active set, which consists of vertices located in a narrow band surrounding the diffuse interface. Since the AC dynamics evolve primarily near this interfacial region, the adaptive restriction eliminates a large number unnecessary updates in the bulk phases. The large ratio observed in Table 1 therefore confirms the principal advantage of the proposed adaptive method. The method preserves the relevant interfacial dynamics while significantly reducing the effective computational workload. As a consequence, the adaptive scheme provides a highly efficient approach for solving the surface AC equation on curved triangular meshes, particularly for problems in which the interface occupies only a small portion of the surface.

The memory usage of the adaptive method is also reduced compared with the global approach. While both methods share the same mesh connectivity and geometric data, the solver-specific arrays in the global computation are allocated for all vertices of the mesh. In contrast, the adaptive method allocates these arrays only for the active vertices near the diffuse interface. Consequently, the adaptive solver requires substantially less working memory when the active region occupies only a small portion of the surface.

3.3. Spinodal decomposition on a sphere

We examine spinodal decomposition [14] on the surface of a sphere of radius $R = 7$. The time step is $\Delta t = 0.001$, and the interface parameter is $\epsilon = \epsilon_m = m h_{\text{geo}}/2\sqrt{2}\text{arctanh}(0.9)$, $m = 8$. The surface AC dynamics on the sphere start from a zero-mean random field with values as

$$\phi(x, y, z, 0) = \text{rand}(x, y, z), \tag{14}$$

where $\text{rand}(x, y, z)$ denotes a uniformly distributed random number in $[-0.5, 0.5]$ (see Fig. 10(a)). The evolution shows two main stages of phase separation. In the early stage (Fig. 10(b)), small fluctuations separate into two almost pure phases, and thin interfaces appear. In the later stage (Fig. 10(c)-(d)), the interfaces move by mean curvature, and domains merge or thin channels disappear. The total interfacial length on the closed surface decreases monotonically, and the late-time morphology tends to two macroscopic domains separated by a smooth closed curve.

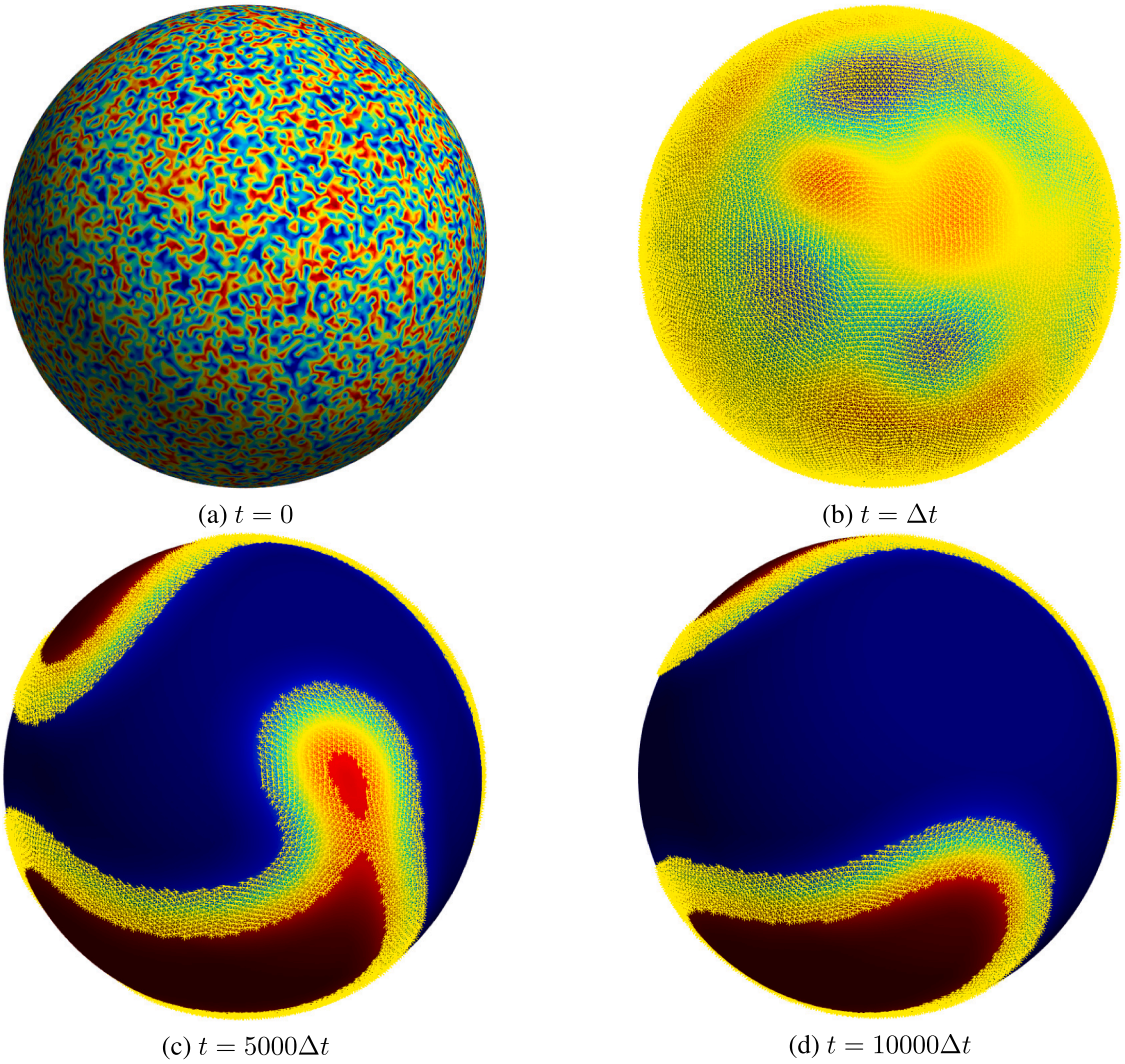


Fig. 10. Spinodal decomposition on a sphere and discrete energy dissipation. (a) Random perturbation. (b) Early spinodal decomposition: rapid formation of diffuse interfaces. (c) Intermediate coarsening by curvature-driven motion and topological reconnections. (d) Late-time morphology with two large domains separated by a smooth closed curve.

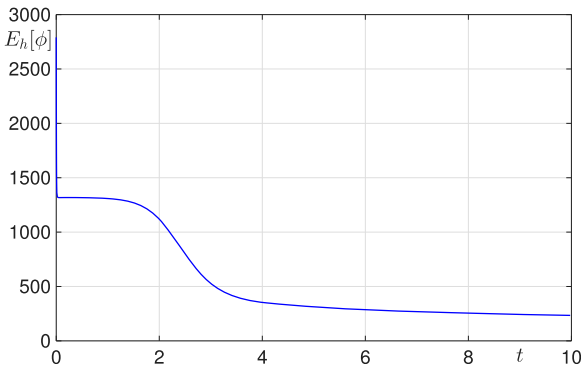
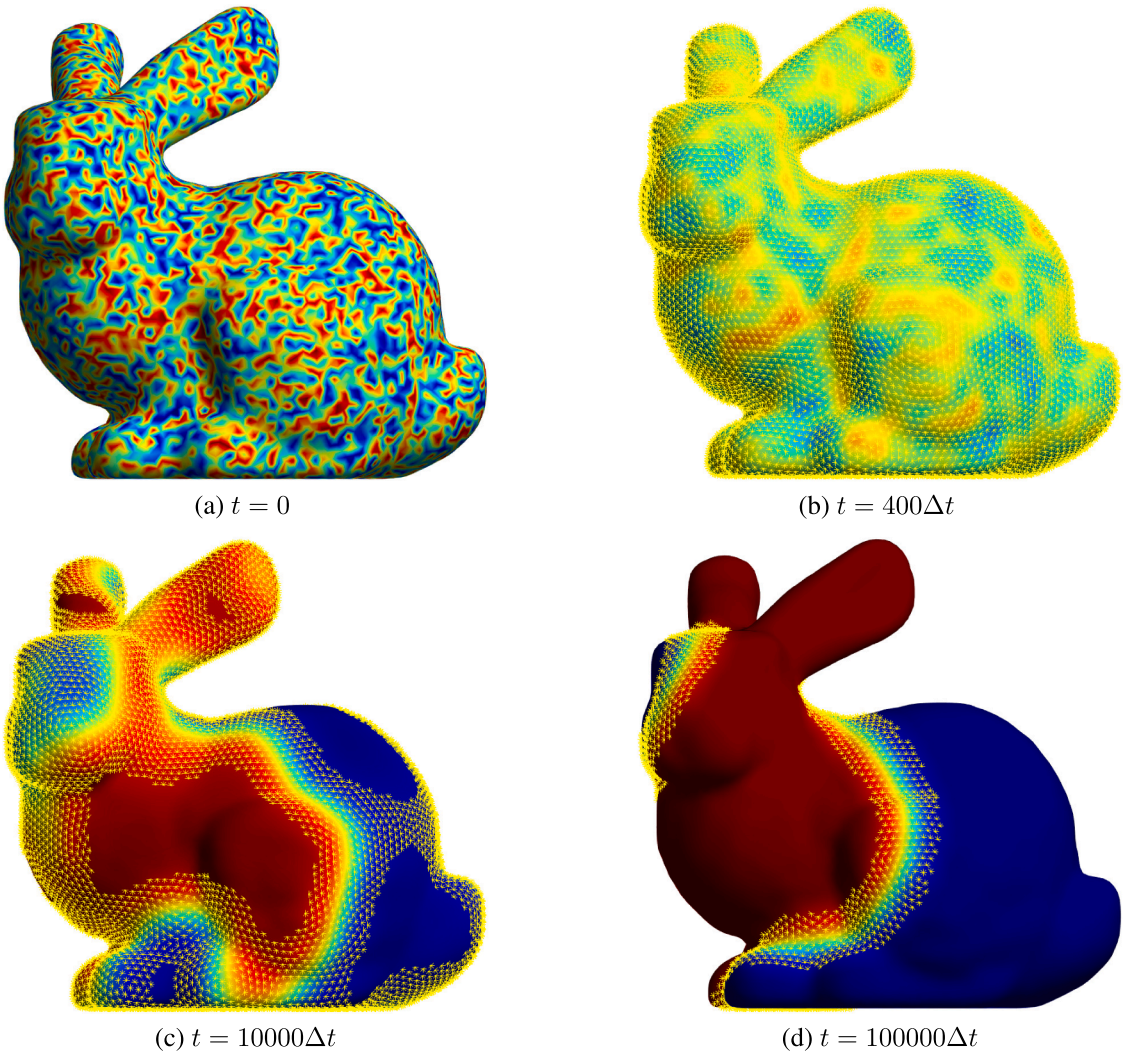


Fig. 11. Discrete surface Ginzburg–Landau energy $E_h[\phi_i]$ computed using cotangent weights and lumped areas.



(a) $t = 0$ (b) $t = 400\Delta t$

(c) $t = 10000\Delta t$ (d) $t = 100000\Delta t$

Fig. 12. Evolutionary dynamics of randomly perturbed numerical solutions for surface AC on the bunny surface.

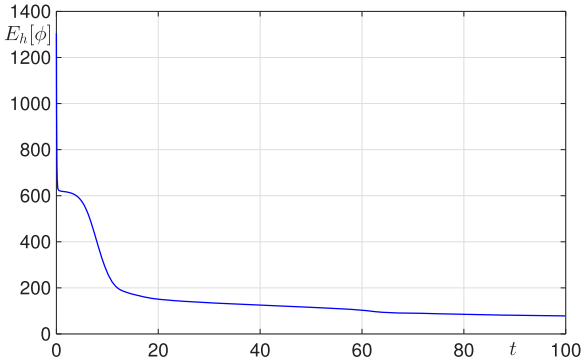


Fig. 13. Discrete surface Ginzburg–Landau energy $E_h[\phi_i]$ on the bunny surface, evaluated every time steps with cotangent weights and lumped areas; the curve decays monotonically.

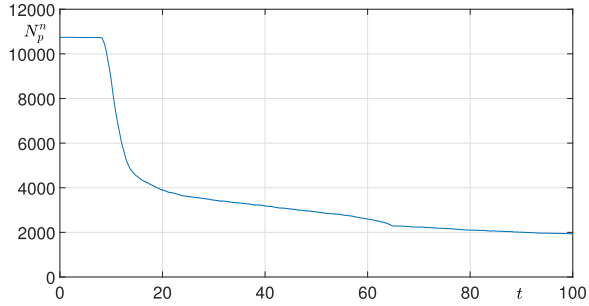


Fig. 14. Temporal evolution of the number of active surface vertices N_p^n used by the narrow band update on the bunny surface.

To verify the gradient-flow structure in discrete form, we compute the discrete Ginzburg–Landau energy for the phase-field ϕ as

$$E_h[\phi_i] = \frac{1}{2}\sum_{i<j} w_{ij}(\phi_i - \phi_j)^2 + \frac{1}{4\epsilon^2}\sum_{i}^{M} A_i(\phi_i^2 - 1)^2, \tag{15}$$

with weights $w_{ij} = 0.5\left(\cot\theta_{ij} + \cot\theta_{ij_+}\right)$ and lumped vertex areas A_i.

Remark 3. The surface AC equation is the L^2-gradient flow of the Ginzburg–Landau energy. The discrete energy E_h in Eq. (15) mirrors this structure via cotangent weights and lumped vertex areas. The proposed algorithm uses a forward Euler update for the diffusion subproblem and an exact pointwise map for the reaction term. The method does not claim unconditional energy stability in the sense of convex-splitting, IEQ, or SAV schemes [47–53]. In general, monotone decay of E_h requires a CFL-type restriction on Δt that controls the explicit diffusion update on the mesh. A sufficient vertexwise admissible time-step bound, which also implies a discrete maximum principle for cotangent Laplacians on admissible meshes, appears in Kim et al. [14]. Under such a restriction, our experiments exhibit strict monotone decay of $E_h(t)$ on both the sphere and bunny surfaces (Figs. 11 and 13).

The energy curve $E_h(t)$ decreases strictly over the whole time interval, which confirms the discrete analogue

$$\frac{d}{dt}E[\phi(t)] = \int_S \left(\nabla_S\phi\cdot\nabla_S\frac{\partial\phi}{\partial t} + \frac{(\phi^3-\phi)}{\epsilon^2}\frac{\partial\phi}{\partial t}\right)dS = \int_S\left(-\Delta_S\phi + \frac{(\phi^3-\phi)}{\epsilon^2}\right)\frac{\partial\phi}{\partial t}dS \tag{16}$$

$$= \int_S \frac{\delta E}{\delta\phi}\frac{\partial\phi}{\partial t}dS = -\int_S\left|\frac{\partial\phi}{\partial t}\right|^2 dS \approx -\sum_{i}^{M} A_i\left(\frac{\partial\phi_i}{\partial t}\right)^2 \le 0, \tag{17}$$

where S denotes the spherical surface, and the last approximation uses mass-lumped quadrature on the mesh. The decay is steep at early time because rapid interface formation dominates, and the rate becomes slower with an algebraic-like regime in the domain-growth stage. During the computation, the order parameter stays in the range $[-1, 1]$, and the interface width stays nearly constant. These results show that the time step and the spatial discretization are stable and satisfy the maximum-principle condition.

In this test, we simulate spinodal decomposition of the AC equation on a triangulated bunny surface. The initial condition is

$$\phi(x, y, z, 0) = \mathrm{rand}(x, y, z), \tag{18}$$

where $\mathrm{rand}(x, y, z)$ is a random number in $[-0.5, 0.5]$. We use $\Delta t = 0.001$ and $\epsilon = \epsilon_m = mh_{\mathrm{geo}}/[2\sqrt{2}\mathrm{arctanh}(0.9)]$ with $m = 8$. The evolution shows the typical two stages of phase separation. In the early stage, small fluctuations separate into two phases and smooth interfaces appear (Fig. 12(b)). Subsequently, curvature-driven coarsening dominates: interfaces migrate over the surface; thin channels pinch off, and domains merge, which reduces the total interfacial length (Fig. 12(c)-(d)).

The discrete Ginzburg–Landau energy

$$E_h[\phi_i] = \frac{1}{2}\sum_{i<j} w_{ij}(\phi_i - \phi_j)^2 + \frac{1}{4\epsilon^2}\sum_{i}^{M} A_i(\phi_i^2 - 1)^2 \tag{19}$$

computed with cotangent weights and lumped areas decreases monotonically over the entire run (Fig. 13), consistent with the gradient-flow structure of the proposed scheme.

Fig. 14 reports the time evolution of N_p^n, the number of active surface vertices updated by the adaptive narrow band strategy on the bunny surface. The active set is defined as the union of vertices with $|\phi_i| < \tau$ and their one-ring neighborhood; it, therefore, approximates the interfacial region in which the phase-field varies most rapidly. Since assembly and updates are restricted to this set, N_p^n is a direct proxy for the per-step computational work. The adaptive workload is reduced by a factor of about five to six between the start ($N_p^0 = 10742$) and the quasi-steady states ($N_p^T = 1928$). The curve decreases over time with small changes from topological events. This result shows that the narrow band follows the interface. The computational cost mainly depends on the interface length rather than the full mesh size.

4. Conclusions

In this work, we developed an adaptive method to solve the AC equation on curved surfaces. The method used triangulated meshes, the cotangent Laplace–Beltrami operator, and an operator-splitting approach. The adaptive narrow-band strategy updated only the interface region, which reduced computational cost while keeping good accuracy. Numerical tests on spherical and complex surfaces showed that the method captured interface motion and phase separation well. The results agreed with analytic solutions and showed faster computation than global methods. The method was accurate and efficient for phase-field simulations on curved surfaces. Future work will extend the approach to higher-order time discretization methods [54]. It will also consider coupling with bulk-surface dynamics and applications in biological and materials science.

Data availability

The data used to support the findings of this study are available from the corresponding author upon request.

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.

Acknowledgment

The first author (Hyundong Kim) was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (RS-2026-25469327, RS-2025-25415913). The corresponding author (J.S. Kim) was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2022R1A2C1003844). The authors thank the reviewers for their helpful comments and suggestions.

References

[1] S. Kwak, Y. Choi, J. Wang, Y. Nam, J. Kim, Phase-field modeling for curvature-dependent tissue growth on surfaces, Eng. Anal. Bound. Elem. 171 (2025) 106090. https://doi.org/10.1016/j.enganabound.2024.106090
[2] P.F. Lenne, V. Trivedi, Sculpting tissues by phase transitions, Nat. Commun. 13 (1) (2022) 664. https://doi.org/10.1038/s41467-022-28151-9
[3] F. Yuan, Membrane bending by protein phase separation, Biophys. J. 120 (3) (2021) 5a–6a. https://doi.org/10.1073/pnas.2017435118
[4] M.A. Hamed, A.A. Nepomnyashchy, Three-dimensional phase field model for actin-based cell membrane dynamics, Math. Model. Nat. Phenom. 16 (2021) 56. https://doi.org/10.1051/mmnp/2021048
[5] H. Kim, S. Kang, G. Lee, S. Yoon, J. Kim, Shape transformation on curved surfaces using a phase-field model, Commun. Nonlinear Sci. Numer. Simul. 133 (2024) 107956. https://doi.org/10.1016/j.cnsns.2024.107956
[6] B. Jiang, Q. Xia, J. Kim, Y. Li, Efficient second-order accurate scheme for fluid–surfactant systems on curved surfaces with unconditional energy stability, Commun. Nonlinear Sci. Numer. Simul. 135 (2024) 108054. https://doi.org/10.1016/j.cnsns.2024.108054
[7] 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 (2) (2021) B479–B507. https://doi.org/10.1137/20M1336734
[8] T. Philippe, H. Henry, M. Plapp, Regularized anisotropic motion-by-curvature in phase-field theory: interface phase separation of crystal surfaces, Phys. Rev. E 106 (3) (2022) 034119. https://doi.org/10.1103/PhysRevE.106.034119
[9] D. Caca, 3D Geometry Reconstruction of Multi-Phase Materials with Limited Experimental Data Using the Phase-Field Method, Bachelor thesis, Eindhoven University of Technology, 2023.
[10] H. Zamani-Gharaghoshi, M. Dehghan, M. Abbaszadeh, Numerical solution of Allen–Cahn model on surfaces via an effective method based on generalized moving least squares (GMLS) approximation and the closest point approach, Eng. Anal. Bound. Elem. 152 (2023) 575–581. https://doi.org/10.1016/j.enganabound.2023.04.019
[11] Z. Sun, S. Zhang, A radial basis function approximation method for conservative Allen–Cahn equations on surfaces, Appl. Math. Lett. 143 (2023) 108634. https://doi.org/10.1016/j.aml.2023.108634
[12] S. Huang, X. Xiao, X. Feng, An adaptive time-stepping method for the binary fluid–surfactant phase-field model on evolving surfaces, J. Sci. Comput. 95 (1) (2023) 29. https://doi.org/10.1007/s10915-023-02150-4
[13] J. Yang, Y. Li, J. Kim, Phase-field modeling and linearly energy-stable Runge–Kutta algorithm of colloidal crystals on curved surfaces, J. Comput. Appl. Math. 443 (2024) 115750. https://doi.org/10.1016/j.cam.2023.115750
[14] H. Kim, J. Wang, S. Ham, Y. Hwang, J. Kim, Stability analysis for the maximum principle preserving finite difference approach of the surface Allen–Cahn equation on curved surfaces, Electron. Res. Arch. 33 (9) (2025) 5748–5768. https://doi.org/10.3934/era.2025255
[15] C. Chen, T. Dang, J. Hong, Strong convergence of adaptive time-stepping schemes for the stochastic Allen–Cahn equation, IMA J. Numer. Anal. 45 (1) (2025) 404–450. https://doi.org/10.1093/imanum/drae009
[16] Y. Chen, D. Luo, W. Pei, Y. Xing, Efficient variable time-stepping adaptive DLN algorithms for the Allen–Cahn equation, J. Sci. Comput. 104 (2) (2025) 67. https://doi.org/10.1007/s10915-025-02980-4
[17] S. Guo, J. Ren, A novel adaptive Crank–Nicolson-type scheme for the time fractional Allen–Cahn model, Appl. Math. Lett. 129 (2022) 107943. https://doi.org/10.1016/j.aml.2022.107943
[18] H. Xu, J. Chen, F. Ma, Adaptive deep learning approximation for Allen–Cahn equation, in: International Conference on Computational Science, Springer International Publishing, Cham, 2022, pp. 271–283. https://doi.org/10.1007/978-3-031-08760-8_23
[19] H. Kim, S. Ham, S. Kwak, J. Kim, An unconditionally stable adaptive finite difference scheme for the Allen–Cahn equation, Comput. Phys. Commun. 315 (2025) 109712. https://doi.org/10.1016/j.cpc.2025.109712
[20] J. Li, J. Zeng, R. Li, An Adaptive Discontinuous Finite Volume Element Method for the Allen–Cahn Equation, Vol. 49, Springer Science and Business Media LLC, 2023. https://doi.org/10.1007/s10444-023-10031-5
[21] W. Xie, Z. Wang, J. Kim, X. Sun, Y. Li, A novel ensemble Kalman filter based data assimilation method with an adaptive strategy for dendritic crystal growth, J. Comput. Phys. 524 (2025) 113711. https://doi.org/10.1016/j.jcp.2024.113711
[22] B. Xia, S. Lai, Q. Xia, X. Liu, Y. Li, J. Kim, Phase-field modeling of fiber-based thermal diffusion and phase transitions in the fused deposition modeling process, Commun. Nonlinear Sci. Numer. Simul. 151 (2025) 109071. https://doi.org/10.1016/j.cnsns.2025.109071

[23] B. Xia, K. Wang, Y. Nam, Z. Li, X. Wu, S. Kwak, J. Ma, J. Kim, Numerical analysis of the Allen–Cahn equation on non-uniform cell sizes, Comput. Appl. Math. 45 (1) (2026) 161. https://doi.org/10.1007/s40314-025-03490-7
[24] Q. Pan, Y. Huang, T. Rabczuk, Y. Yang, X. Yang, Fully-discrete decoupled subdivision-based IGA-IEQ-ZEC numerical scheme for the binary surfactant phase-field model coupled with Darcy flow equations on surfaces, Comput. Meth. Appl. Mech. Eng. 436 (2025) 117733. https://doi.org/10.1016/j.cma.2025.117733
[25] S. Kim, S. Kwak, S. Ham, Y. Hwang, J. Kim, Simulation of dendritic growth on a spherical surface using a multi-component phase-field model, Int. Commun. Heat Mass Transf. 167 (2025) 109195. https://doi.org/10.1016/j.icheatmasstransfer.2025.109195
[26] M. Kloppe, S. Aland, A phase-field model of elastic and viscoelastic surfaces in fluids, Comput. Meth. Appl. Mech. Eng. 428 (2024) 117090. https://doi.org/10.1016/j.cma.2024.117090
[27] J. Yang, J. Wang, S. Kwak, S. Ham, J. Kim, A modified Allen–Cahn equation with a mesh size-dependent interfacial parameter on a triangular mesh, Comput. Phys. Commun. 304 (2024) 109301. https://doi.org/10.1016/j.cpc.2024.109301
[28] Y. Hwang, S. Ham, C. Lee, G. Lee, S. Kang, J. Kim, A simple and efficient numerical method for the Allen–Cahn equation on effective symmetric triangular meshes, Electron. Res. Arch. 31 (8) (2023) 4557–4578. https://doi.org/10.3934/era.2023233
[29] W. Xu, D. Jiang, C. Zhang, H. Li, S. Qiang, Y. Li, M. Yuan, C. Zhang, An adaptive mesh refinement strategy for 3D phase modeling of brittle fracture, Eng. Fract. Mech. 284 (2023) 109241. https://doi.org/10.1016/j.engfracmech.2023.109241
[30] H. Saberi, H. Saberi, T.Q. Bui, Y. Heider, M.N. Nguyen, A multi-level adaptive mesh refinement method for phase-field fracture problems, Theor. Appl. Fract. Mech. 125 (2023) 103920. https://doi.org/10.1016/j.tafmec.2023.103920
[31] H.Y. Kim, H.G. Kim, A novel adaptive mesh refinement scheme for the simulation of phase-field fracture using trimmed hexahedral meshes, Int. J. Numer. Methods Eng. 122 (6) (2021) 1493–1512. https://doi.org/10.1002/nme.6587
[32] Y. Li, D. Jeong, H. Kim, C. Lee, J. Kim, Comparison study on the different dynamics between the Allen–Cahn and the Cahn–Hilliard equations, Comput. Math. Appl. 77 (2) (2019) 311–322. https://doi.org/10.1016/j.camwa.2018.09.034
[33] Y. Chen, Y. Huang, N. Yi, P. Yin, Recovery-type a posteriori error estimation of an adaptive finite element method for the Cahn–Hilliard equation, J. Sci. Comput. 98 (2) (2024) 35. https://doi.org/10.1007/s10915-023-02418-9
[34] S. Ham, Y. Li, D. Jeong, C. Lee, S. Kwak, Y. Hwang, J. Kim, An explicit adaptive finite difference method for the Cahn–Hilliard equation, J. Nonlinear Sci. 32 (6) (2022) 80. https://doi.org/10.1007/s00332-022-09844-3
[35] C. Bracco, C. Giannelli, A. Reali, M. Torre, R. Vázquez, Adaptive isogeometric phase-field modeling of the Cahn–Hilliard equation: suitably graded hierarchical refinement and coarsening on multi-patch geometries, Comput. Meth. Appl. Mech. Eng. https://doi.org/10.1016/j.cma.2023.116355
[36] G. Xu, Discrete Laplace–Beltrami operators and their convergence, Comput. Aided Geom. Des. 21 (2004) 767–784. https://doi.org/10.1016/j.cagd.2004.07.007
[37] G. Xu, Convergence of discrete Laplace–Beltrami operators over surfaces, Comput. Math. Appl. 48 (2004) 347–360. https://doi.org/10.1016/j.camwa.2004.05.001
[38] H. Kim, A. Yun, S. Yoon, C. Lee, J. Park, J. Kim, Pattern Formation in Reaction–Diffusion Systems on Evolving Surfaces, Vol. 80, Elsevier BV, 2020. https://doi.org/10.1016/j.camwa.2020.08.026
[39] H. Kim, Explicit Time-stepping Approach for Pattern Formation on Evolving Curved Surfaces, Ph.d. thesis, Korea University, 2022.
[40] H. Kim, Numerical dynamics for the Allen–Cahn equation with variable mobility on curved surfaces, J. Korean Soc. Ind. Appl. Math. 29 (4) (2025) 360–376. https://doi.org/10.12941/jksiam.2025.29.360
[41] A.I. Bobenko, B.A. Springborn, A discrete Laplace–Beltrami operator for simplicial surfaces, Discret. Comput. Geom. 38 (4) (2007) 740–756. https://doi.org/10.1007/s00454-007-9006-1
[42] M. Belkin, J. Sun, Y. Wang, Discrete Laplace operator on meshed surfaces, in: Proceedings of the Twenty-Fourth Annual Symposium on Computational Geometry, 2008, pp. 278–287. https://doi.org/10.1145/1377676.1377725
[43] M. Reuter, S. Biasotti, D. Giorgi, G. Patane, M. Spagnuolo, Discrete Laplace–Beltrami operators for shape analysis and segmentation, Comput. Graph. 33 (3) (2009) 381–390. https://doi.org/10.1016/j.cag.2009.03.005
[44] H. Kim, S. Kwak, S. Ham, J. Kim, In silico investigation of the formation of multiple intense zebra stripes using extending domain, Math. Comput. Simul. 225 (2024) 648–658. https://doi.org/10.1016/j.matcom.2024.06.010
[45] S.J. Ruuth, B. Merriman, A simple embedding method for solving partial differential equations on surfaces, J. Comput. Phys. 227 (3) (2008) 1943–1961. https://doi.org/10.1016/j.jcp.2007.10.009
[46] Y. Choi, D. Jeong, S. Lee, M. Yoo, J. Kim, Motion by mean curvature of curves on surfaces using the Allen–Cahn equation, Int. J. Eng. Sci. 97 (2015) 126–132. https://doi.org/10.1016/j.ijengsci.2015.10.002
[47] J. Shen, J. Xu, J. Yang, The scalar auxiliary variable (SAV) approach for gradient flows, J. Comput. Phys. 353 (2018) 407–416. https://doi.org/10.1016/j.jcp.2017.10.021
[48] Z. Liu, X. Li, Efficient modified techniques of invariant energy quadratization approach for gradient flows, Appl. Math. Lett. 98 (2019) 206–214. https://doi.org/10.1016/j.aml.2019.06.006
[49] J. Shen, J. Xu, J. Yang, A new class of efficient and robust energy stable schemes for gradient flows, SIAM Rev. 61 (3) (2019) 474–506. https://doi.org/10.1137/17M1150153
[50] Y. Gong, J. Zhao, Q. Wang, Arbitrarily high-order unconditionally energy stable schemes for thermodynamically consistent gradient flow models, SIAM J. Sci. Comput. 42 (1) (2020) B135–B156. https://doi.org/10.1137/18M1213579
[51] X. Yang, G.D. Zhang, Convergence analysis for the invariant energy quadratization (IEQ) schemes for solving the Cahn–Hilliard and Allen–Cahn equations with general nonlinear potential, J. Sci. Comput. 82 (3) (2020) 55. https://doi.org/10.1007/s10915-020-01151-x
[52] Z. Liu, X. Li, Step-by-step solving schemes based on scalar auxiliary variable and invariant energy quadratization approaches for gradient flows, Numer. Algorithms 89 (1) (2022) 65–86. https://doi.org/10.1007/s11075-021-01106-9
[53] H.G. Lee, J. Shin, J.Y. Lee, A high-order and unconditionally energy stable scheme for the conservative Allen–Cahn equation with a nonlocal Lagrange multiplier, J. Sci. Comput. 90 (1) (2022) 51. https://doi.org/10.1007/s10915-021-01735-1
[54] C. Jia, J. Yang, An effective numerical method for solving ternary Cahn–Hilliard-type Nakazawa-Ohta phase-field model on three-dimensional surfaces, Commun. Nonlinear Sci. Numer. Simul. 152 (2026) 109427. https://doi.org/10.1016/j.cnsns.2025.109427