Available online at www.sciencedirect.com
ScienceDirect

Mathematics and Computers in Simulation 215 (2024) 338–356

MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original articles

Fast and efficient numerical method for solving the Allen–Cahn equation on the cubic surface

Youngjin Hwang, Junxiang Yang, Gyeongyu Lee, Seokjun Ham, Seungyoon Kang, Soobin Kwak, Junseok Kim*

Department of Mathematics, Korea University, Seoul, 02841, Republic of Korea

Received 25 January 2023; received in revised form 28 June 2023; accepted 28 July 2023
Available online 3 August 2023

Abstract

In this study, we present a fast and efficient finite difference method (FDM) for solving the Allen–Cahn (AC) equation on the cubic surface. The proposed method applies appropriate boundary conditions in the two-dimensional (2D) space to calculate numerical solutions on cubic surfaces, which is relatively simpler than a direct computation in the three-dimensional (3D) space. To numerically solve the AC equation on the cubic surface, we first unfold the cubic surface domain in the 3D space into the 2D space, and then apply the FDM on the six planar sub-domains with appropriate boundary conditions. The proposed method solves the AC equation using an operator splitting method that splits the AC equation into the linear and nonlinear terms. To demonstrate that the proposed algorithm satisfies the properties of the AC equation on the cubic surface, we perform the numerical experiments such as convergence test, total energy decrease, and maximum principle.

Keywords: Cubic surface; Finite difference method; Diffusion equation; Allen–Cahn equation

1. Introduction

In this paper, we present the fast and efficient numerical algorithm which solves the Allen–Cahn (AC) equation [1] on the cubic surface. To simply and efficiently solve the AC equation, we consider the unfolded cubic surface domain with appropriate boundary conditions in two-dimensional (2D) space and propose a fast and efficient finite difference method (FDM). The spherical surface can be locally approximated by plane. Using this property, solving the partial difference equation (PDE) on the cubic surface is an approximation for solving various PDEs on the spherical surface domain.

Over the past decades, many researchers have made efforts to solve the phase-field equations on curved surfaces, including spherical surfaces [6,29]. In [15], Lee and Kim presented a numerical method to solve the phase-field crystal equation on curved surfaces by using the closest point method and the pseudo-Neumann boundary condition. Yang and Kim [26] proposed an efficient and practical computational method to simulate the square phase field crystal dynamics on arbitrary surfaces. Sun et al. [22] rigorously verified the stability of first-order numerical

* Corresponding author.
E-mail address: cfdkim@korea.ac.kr (J. Kim).
URL: https://mathematicians.korea.ac.kr/cfdkim (J. Kim).

https://doi.org/10.1016/j.matcom.2023.07.024

schemes using the surface finite element method and the scalar auxiliary variable method for numerical simulation of the binary fluid–surfactant phase field model coupled with geometric curvature on a curved surface. There are also researches to numerically solve the AC equation on a planar or curved surface [24,30].

A cubic surface simplifies problems on the spherical surface in 3D space, because we can consider each face as a plane with zero curvature. Researchers have used cubic surfaces to solve problems on spherical surfaces. The cubed-sphere transform was originally introduced by Sadourny in 1971 [21] and is derived from the projection of a cube onto a sphere. A cube-sphere grid is a grid created by a cube-sphere transformation that maps a quasi-uniform grid of spheres onto a regular polyhedron by choosing a central projection. In addition, it has been further developed by [13,17,19,20,23,31]. In [17], McGregor demonstrated that the semi-Lagrangian advection technique may be efficiently applied to a cubic gnomonic grid of a sphere. Taylor et al. [23] used a gnomonic projection to evaluate the suitability of the spectral element method for climate modeling. Ronchi et al. [20] developed a numerical method to solve a PDE on the grid obtained by decomposing a sphere into six identical regions using centroid projection from a circumscribed cube onto a spherical surface. In [19], advection tests were performed using the finite-volume transport method on various cubed-sphere grids to evaluate various cubic projections and grid modifications. Ivan et al. [13] presented the fourth-order accurate finite volume method for conservation laws on the 3D cubed-sphere grids, based on the central essentially non-oscillatory finite volume method which they introduced in 2D space. Zhang et al. [31] developed a prolate-element method on a spherical surface grid based on a cubed-sphere transform to solve a PDE for a sphere. To the best of the authors' knowledge, the properties of the AC equation on the cubic surface, such as motion by mean curvature using the AC equation on the cubic surface domain, have not been considered in depth before. Therefore, we conduct numerical experiments and evaluate the properties of the AC equation and highlight the different dynamics of the AC equation on the cubic surface domain.

The AC equation is a phase field model that includes the diffusion and nonlinear terms. The AC equation was originally introduced as a mathematical model for anti-phase domain coarsening in a binary alloy. Because the AC equation has the properties as total energy decreasing and motion by mean curvature flow, it has been widely used to model various physical and applied problems, such as phase separation of binary alloys at a fixed temperature, crystal growth [3], vesicle membranes, image segmentation [2,14], the nucleation of solids, and the mixture of two incompressible fluids [9], phase transitions [8], and interfacial dynamics in materials science [7,10,12,25]. Xiao and Feng [25] presented the highly efficient spatial–temporal operator splitting finite element method to solve the AC equation. The main advantage of the presented method is to reduce the complexity of high-dimensional computation by splitting the high-dimensional problem into a series of one-dimensional subproblems. Deng and Zhao [7] developed two new energy-dissipation-preserving alternating direction implicit (ADI) methods for the 2D AC equation in combination with the invariant energy quadratization methods proposed by Xiaofeng Yang [28] and the ADI method. In [11], the authors devised second-order accurate, unconditionally uniquely solvable, and unconditionally energy stable methods for the nonlocal Cahn–Hilliard (CH) and nonlocal AC equations with periodic boundary conditions. For the time-dependent AC equation on surfaces with no boundary, Mohammadi et al. [18] presented the numerical simulation and error estimation based on radial basis functions. Choi et al. [4] proposed an accurate and fast numerical scheme for motion by mean curvature using the AC equation on curved surfaces in 3D space. In addition, Choi et al. proposed the hybrid numerical method for the motion by mean curvature of the AC equation on curved surfaces in 3D space in [5].

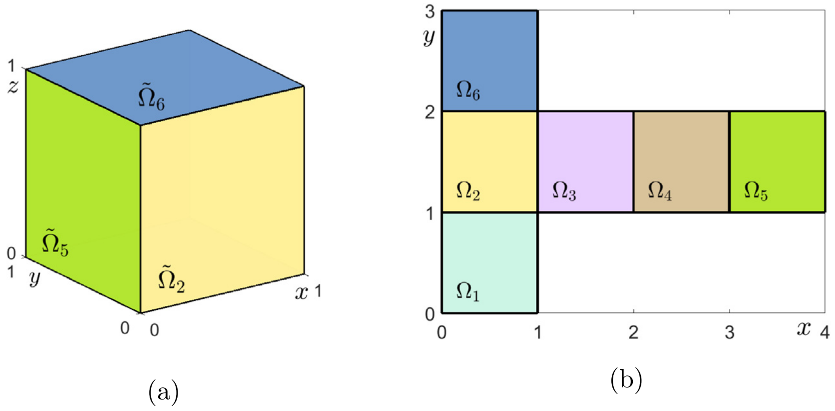
The primary purpose of this study is to present a fast and efficient finite difference method for solving the AC equation on the cubic surface.

The layout of this paper is structured as follows. The governing equations are discussed in Section 2. In Section 3, we proposes a numerical solution algorithm for solving the AC equation on the cubic surface. In Section 5, we present numerical simulations for the AC equations using the proposed method on cubic surfaces. Finally, we presents the conclusion of this paper in Section 6.

2. Governing equation

In this section, we consider the AC equation on the unit cubic surface. We define the unit cubic surface domain as follows:

$$\begin{aligned}\tilde{\Omega} = \{(x,y,z)|&(x,y,0),\ (x,y,1),\ (x,0,z),\ (x,1,z),\ (0,y,z),\ (1,y,z),\\ &0\le x\le 1,\ 0\le y\le 1,\ 0\le z\le 1\}.\end{aligned} \tag{1}$$



(a)

(b)

Fig. 1. (a) Cubic surface domain $\tilde{\Omega}$ in 3D and (b) unfolded cubic surface domain Ω in 2D.

We decompose cubic surface domain $\tilde{\Omega}$ into six sub-domain $\tilde{\Omega}_k$, $k = 1, 2, \ldots, 6$ as:

$$\begin{aligned}
\tilde{\Omega}_1 &= \{(x, y, 0) \mid 0 < x < 1,\ 0 < y < 1\},\ \tilde{\Omega}_2 = \{(x, 0, z) \mid 0 < x < 1,\ 0 < z < 1\},\\
\tilde{\Omega}_3 &= \{(1, y, z) \mid 0 < y < 1,\ 0 < z < 1\},\ \tilde{\Omega}_4 = \{(x, 1, z) \mid 0 < x < 1,\ 0 < z < 1\},\\
\tilde{\Omega}_5 &= \{(0, y, z) \mid 0 < y < 1,\ 0 < z < 1\},\ \tilde{\Omega}_6 = \{(x, y, 1) \mid 0 < x < 1,\ 0 < y < 1\},\\
\tilde{\Omega} &= \cup_{k=1}^{6} \overline{\tilde{\Omega}}_k,
\end{aligned}$$

where $\overline{\tilde{\Omega}}_k$ is the closure set of $\tilde{\Omega}_k$, $k = 1, \ldots, 6$ and schematically shown in Fig. 1(a). For an unfolded cubic surface in 2D space, we also define sub-domain Ω_k, $k = 1, 2, \ldots, 6$ and global domain Ω as follows:

$$\begin{aligned}
\Omega_1 &= \{(x, -y) \mid (x, y, z) \in \tilde{\Omega}_1\}, & \Omega_2 &= \{(x, z + 1) \mid (x, y, z) \in \tilde{\Omega}_2\},\\
\Omega_3 &= \{(y + 1, z + 1) \mid (x, y, z) \in \tilde{\Omega}_3\}, & \Omega_4 &= \{(3 - x, z + 1) \mid (x, y, z) \in \tilde{\Omega}_4\},\\
\Omega_5 &= \{(4 - y, z + 1) \mid (x, y, z) \in \tilde{\Omega}_5\}, & \Omega_6 &= \{(x, y + 2) \mid (x, y, z) \in \tilde{\Omega}_6\},\\
\Omega &= \cup_{k=1}^{6} \overline{\Omega}_k,
\end{aligned}$$

which is schematically shown in Fig. 1(b). We note that we can straightforwardly extend the proposed numerical scheme to surfaces of the arbitrary-sized cuboid. Unless otherwise noted, the unit cube is considered. The boundary condition for each Ω_k, where $k = 1, \ldots, 6$, is based on the cubic surface domain $\tilde{\Omega}$. Among the methods of unfolding the unit cube in Fig. 1(a), we chose to unfold it as in Fig. 1(b). Thus, the boundary condition for $\Omega_3 \subset \Omega$ as an example is shown in Fig. 2 and the boundary conditions for other $\Omega_k \subset \Omega$, $k = 1, 2, 4, 5, 6$ are applied based on the cubic surface domain $\tilde{\Omega}$, similar to Ω_3. Therefore, we obtain the boundary condition for domain Ω in 2D space. Detailed boundary conditions are described in Section 3. The AC equation is given as

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = -\frac{F'(u(\mathbf{x}, t))}{\epsilon^2} + \Delta u(\mathbf{x}, t), \quad \mathbf{x} \in \Omega,\ t > 0, \tag{2}$$

where Ω is a domain, u is the non-conserved order parameter as the phase-field, Δ is the Laplace operator, and ϵ is the small positive parameter related to interfacial transition thickness. The function $F(u) = 0.25(u^2 - 1)^2$ is the Helmholtz free-energy functional which has the double well potential. In Fig. 3, we illustrate profile of Helmholtz free energy $F(u)$. The AC equation is L^2 gradient flow of the following Ginzburg–Landau energy functional:

$$\mathcal{E}^{AC}(t) = \sum_{k=1}^{6} \int_{\Omega_k} \left(\frac{F(u(\mathbf{x}, t))}{\epsilon^2} + \frac{|\nabla u(\mathbf{x}, t)|^2}{2} \right) d\mathbf{x}.$$

The AC equation is a reaction–diffusion equation, therefore we obtain the diffusion equation by excluding the nonlinear term from the AC equation. The diffusion equation is a parabolic partial differential equation as

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = \Delta u(\mathbf{x}, t), \quad \mathbf{x} \in \Omega,\ t > 0, \tag{3}$$

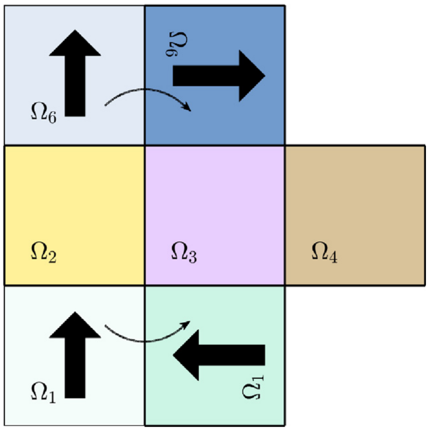


Fig. 2. Schematic diagram of the boundary condition for Ω_3.

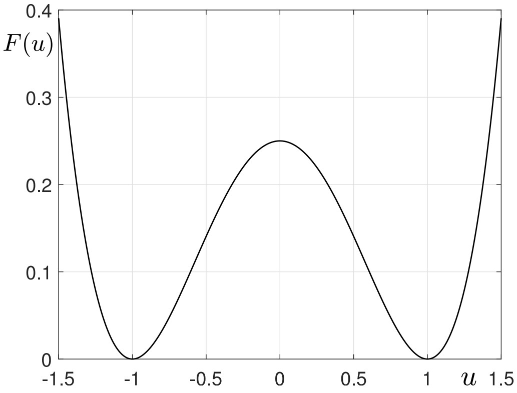


Fig. 3. Helmholtz free energy $F(u) = 0.25(u^2 - 1)^2$.

which is used primarily in mathematics and physics and is a special case of the convection–diffusion equation when bulk velocity is zero. The diffusion equation is derived from the following energy functional:

$$\mathcal{E}^D(u(\mathbf{x}, t)) = \sum_{k=1}^{6} \int_{\Omega_k} \frac{|\nabla u(\mathbf{x}, t)|^2}{2} d\mathbf{x}.$$

3. Numerical solution algorithm

We describe the proposed numerical solution algorithm to solve the AC equation (2) on the unfolded unit cubic surface domain. We unfold the unit cubic surface domain into a planar surface and discretize it as shown in Fig. 4.

We define index sets for discretization as

$$\begin{aligned}
I_1^d &= \{(i, j) \mid i = 1, 2, \ldots, N, \;\; j = 1, 2 \ldots, N\},\\
I_2^d &= \{(i, j) \mid i = 1, 2 \ldots, N, \;\; j = N+1, N+2, \ldots, 2N\},\\
I_3^d &= \{(i, j) \mid i = N+1, N+2, \ldots, 2N, \;\; j = N+1, N+2, \ldots, 2N\},\\
I_4^d &= \{(i, j) \mid i = 2N+1, 2N+2, \ldots, 3N, \;\; j = N+1, N+2, \ldots, 2N\},\\
I_5^d &= \{(i, j) \mid i = 3N+1, 3N+2, \ldots, 4N, \;\; j = N+1, N+2, \ldots, 2N\},\\
I_6^d &= \{(i, j) \mid i = 1, 2, \ldots, N, \;\; j = 2N+1, 2N+2, \ldots, 3N\},\\
I^d &= \cup_{k=1}^{6} I_k^d.
\end{aligned}$$

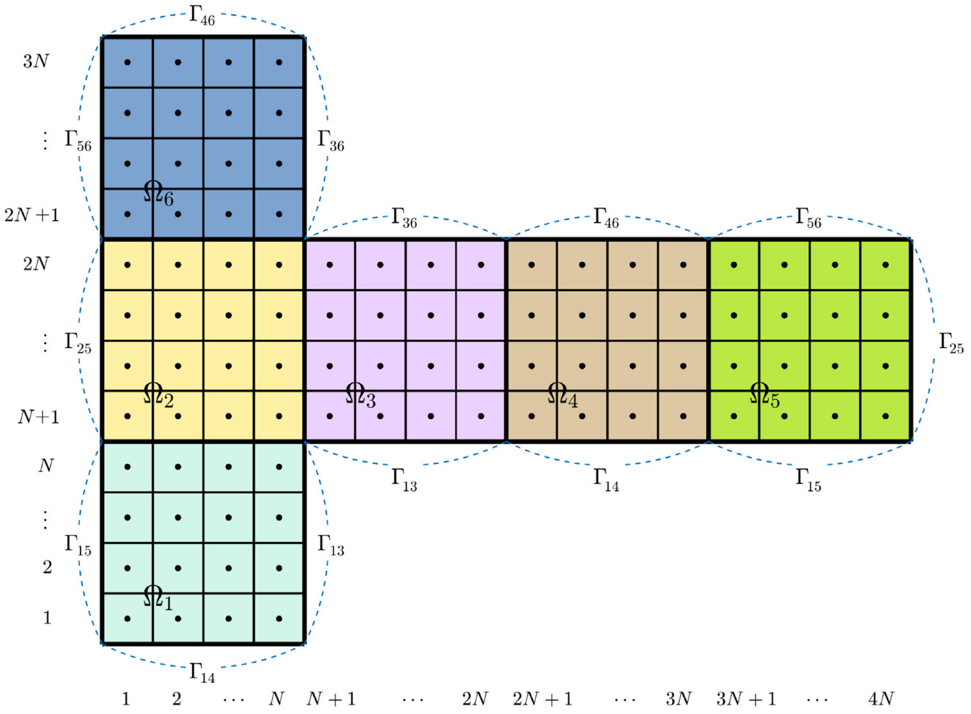


Fig. 4. Schematic of unfolded computational discrete domain Ω^d.

Additionally, for each Ω_k, $k = 1, \dots, 6$, we discretize it as follows:

$$\Omega_k^d = \{(x_i, y_j) \mid (i, j) \in I_k\}, \text{ for } k = 1, 2, \dots, 6, \tag{4}$$

where $x_i = (i - 0.5)h$, $y_j = (j - 0.5)h$, $h = 1/N$ is the uniform space step size, and N is a positive integer. Let u_{ij}^n be the numerical approximation of $u(x_i, y_j, n\Delta t)$, where Δt is the time step, $\Omega^d = \cup_{k=1}^6 \Omega_k^d$ is the discrete unfolded computational domain, and Γ_{ij} is denoted as the interface between the domains Ω_i and Ω_j.

The method is based on an operator splitting method, which is to split the original problem into a sequence of simpler problems. We first denote by $\mathcal{L}$ the exact solution operator associated with the linear equation

$$\mathcal{L} : u_t = \Delta u. \tag{5}$$

Then, we denote by $\mathcal{N}$ the exact solution operator associated with the nonlinear equation

$$\mathcal{N} : u_t = -\frac{u^3 - u}{\epsilon^2}. \tag{6}$$

Therefore, the solution of AC equation (2) from t to $t + \Delta t$ can be evolved in time in two sub-steps:

$$u(\mathbf{x}, t + \Delta t) = (\mathcal{N}(\Delta t) \circ \mathcal{L}(\Delta t))\, (u(\mathbf{x}, t)) + O((\Delta t)^2).$$

We use a method that solves linear terms with the explicit Euler's method and analytically solves the nonlinear terms. We solve the AC equation on Ω^d in two steps. Here, we use the discrete Laplace operator defined by $\Delta_h u_{ij}^n = (u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4u_{ij}^n)/h^2$, and the exact solution operators $\mathcal{L}$ and $\mathcal{N}$ are replaced by their numerical approximations $\mathcal{L}_h$ and $\mathcal{N}_h$, respectively. As a first step, we solve Eq. (5) by applying the explicit Euler's method. That is, for $(i, j) \in I^d$,

$$\frac{u_{ij}^* - u_{ij}^n}{\Delta t} = \Delta_h u_{ij}^n = \frac{u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4u_{ij}^n}{h^2}. \tag{7}$$

If we rewrite Eq. (7), then we have

$$\mathcal{L}_h : u_{ij}^* = u_{ij}^n + \frac{\Delta t}{h^2}\left(u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4u_{ij}^n\right). \tag{8}$$

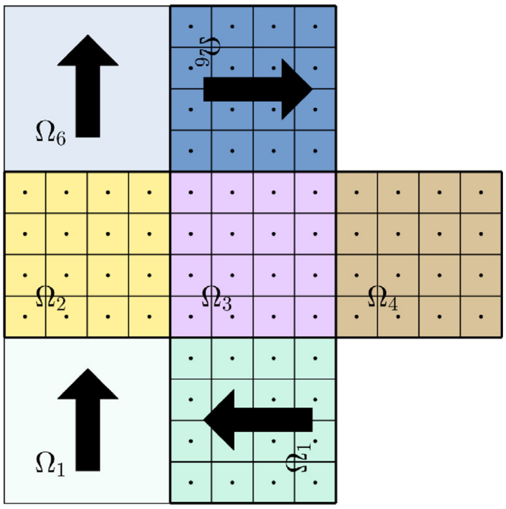


Fig. 5. Schematic diagram of hypothetical domains for domain Ω_3^d.

For the purpose of clarity, we consider only one sub-domain Ω_3^d. The appropriate boundary condition, which is the main focus of the proposed method, assume that Ω_6 and Ω_1, rotated 270 and 90 degrees counterclockwise, are above and below Ω_3, respectively, as shown in Fig. 5. The left and right boundary conditions are well defined by the discrete unfolded computational domain Ω^d. Therefore, only the upper and lower boundary conditions are considered. Then, by assumption, the boundary condition of the discrete domain Ω_3^d is as follows.

$$u^n_{N+i,N} = u^n_{N,N+1-m},\ u^n_{N+m,2N+1} = u^n_{N,2N+m},\ m = 1, \ldots, N,$$

Similarly, appropriate boundary conditions are applied for all discrete domains Ω_k^d, $k = 1, 2, 4, 5, 6$ as follows:

For Ω_1^d, for $m = 1, \ldots, N$,

$$u_{0,m} = u_{3N+m,N+1},\ u_{m,0} = u_{3N+1-m,N+1},\ u_{N+1,m} = u_{2N+1-m,N+1},$$

For Ω_2^d, for $m = 1, \ldots, N$,

$$u_{0,N+m} = u_{4N,N+m},$$

For Ω_4^d, for $m = 1, \ldots, N$,

$$u_{2N+m,N} = u_{N+1-m,1},\ u_{2N+m,2N+1} = u_{N+1-m,3N},$$

For Ω_5^d, for $m = 1, \ldots, N$,

$$u_{3N+m,N} = u_{1,m},\ u_{3N+m,2N+1} = u_{1,3N+1-m},\ u_{4N+1,N+m} = u_{1,N+m},$$

For Ω_6^d, for $m = 1, \ldots, N$,

$$u_{0,2N+m} = u_{4N+1-m,2N},\ u_{m,3N+1} = u_{3N+1-m,2N},\ u_{N+1,2N+m} = u_{N+m,2N}.$$

As a next step, for $(i, j) \in I^d$, Eq. (6) is solved analytically with the initial condition u^*_{ij} and the solution u^{n+1}_{ij} is given by

$$\mathcal{N}_h : u^{n+1}_{ij} = \frac{u^*_{ij}}{\sqrt{e^{\frac{-2\Delta t}{\epsilon^2}} + (u^*_{ij})^2\left(1 - e^{\frac{-2\Delta t}{\epsilon^2}}\right)}}. \tag{9}$$

In summary, an explicit operator splitting method for the AC equation (2) is proposed as follows.

$$\begin{cases} u^*_{ij} &= u^n_{ij} + \frac{\Delta t}{h^2}\left(u^n_{i+1,j} + u^n_{i-1,j} + u^n_{i,j+1} + u^n_{i,j-1} - 4u^n_{ij}\right), \\ u^{n+1}_{ij} &= \frac{u^*_{ij}}{\sqrt{e^{\frac{-2\Delta t}{\epsilon^2}} + (u^*_{ij})^2\left(1-e^{\frac{-2\Delta t}{\epsilon^2}}\right)}}. \end{cases}$$

4. Numerical analysis

4.1. The discrete maximum principle

In this section, we will show that the proposed method preserves the discrete maximum principle. We consider the discrete maximum principle for the proposed method and the maximum norm $\|\cdot\|_\infty$ is defined as

$$\|u^n\|_\infty = \max_{(i,j)\in I^d} |u^n_{ij}|.$$

Theorem 1. *The proposed method satisfies the maximum principle in the sense that for all $\Delta t > 0$ and $h > 0$ that satisfy $\Delta t/h^2 \le 1/4$, that is, it holds*

$$\|u^{n+1}\|_\infty \le 1$$

with a condition $\|u^n\|_\infty \le 1$.

Proof. Let $\alpha = \Delta t/h^2$. If $\alpha \le 1/4$, then we obtain the following inequality for all $(i, j) \in I^d$ by Eq. (8).

$$\begin{aligned}|u^*_{ij}| &= |u^n_{ij} + \alpha\left(u^n_{i+1,j} + u^n_{i-1,j} + u^n_{i,j+1} + u^n_{i,j-1} - 4u^n_{ij}\right)| \\ &\le \alpha|u^n_{i+1,j}| + \alpha|u^n_{i-1,j}| + \alpha|u^n_{i,j+1}| + \alpha|u^n_{i,j-1}| + (1-4\alpha)|u^n_{ij}| \\ &\le \|u^n\|_\infty, \end{aligned} \tag{10}$$

where we have used the following fact: For $(i + 1, j) \notin I^d$, there exists $(l, m) \in I^d$ such that $u^n_{lm} = u^n_{i+1,j}$ from the boundary condition. Therefore, $|u^n_{i+1,j}| = |u^n_{lm}| \le \|u^n\|_\infty$. The other cases $(i - 1, j),\ (i, j + 1),\ (i, j - 1) \notin I^d$ are similarly defined. From Eq. (10), we have

$$\|u^*\|_\infty \le \|u^n\|_\infty \le 1. \tag{11}$$

Considering the initial condition, the following inequality is satisfied for all $(i, j) \in I^d$.

$$e^{\frac{-2\Delta t}{\epsilon^2}} + (u^*_{ij})^2\left(1 - e^{\frac{-2\Delta t}{\epsilon^2}}\right) = \left(1 - \left(u^*_{ij}\right)^2\right)e^{\frac{-2\Delta t}{\epsilon^2}} + \left(u^*_{ij}\right)^2 \ge \left(u^*_{ij}\right)^2.$$

Consequently, we obtain

$$\sqrt{e^{\frac{-2\Delta t}{\epsilon^2}} + (u^*_{ij})^2\left(1 - e^{\frac{-2\Delta t}{\epsilon^2}}\right)} \ge |u^*_{ij}|,$$

and thus it follows that

$$|u^{n+1}_{ij}| = \frac{|u^*_{ij}|}{\sqrt{e^{\frac{-2\Delta t}{\epsilon^2}} + (u^*_{ij})^2\left(1 - e^{\frac{-2\Delta t}{\epsilon^2}}\right)}} \le 1.$$

That is, it means that the following inequality is satisfied.

$$\|u^{n+1}\|_\infty \le 1.$$

Therefore, for all Δt and h that satisfies $\alpha \le 1/4$, the following inequality is satisfied.

$$\|u^{n+1}\|_\infty \le 1.$$

This means that the numerical method satisfies the discrete maximum principle property and is stable. □

4.2. Error estimate

Let C be a generic constant independent Δt and h. We define the discrete L^2-norm on Ω^d as follows:

$$\|u\|_2 = \sqrt{\frac{1}{6N}\sum_{(i,j)\in I^d} u^2_{ij}}.$$

Theorem 2. *If* $\alpha \le 1/4$, *then for all* $n = 1, 2, \ldots,$

$$\|\tilde{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 \le C\left(\Delta t + h^2\right).$$

Proof. For $n \ge 0$, we obtain by Eq. (8)

$$\mathcal{L}(u_{ij}^n) = \mathcal{L}_h(u_{ij}^n) + O\left(\Delta t + h^2\right) = u_{ij}^* + O\left(\Delta t + h^2\right). \tag{12}$$

Then, by Eq. (9), we have

$$(\mathcal{N}_h(u_{ij}^*))^2 = \frac{(u_{ij}^*)^2}{e^{\frac{-2\Delta t}{\epsilon^2}} + (u_{ij}^*)^2\left(1 - e^{\frac{-2\Delta t}{\epsilon^2}}\right)} \le \frac{(u_{ij}^*)^2}{e^{\frac{-2\Delta t}{\epsilon^2}}} = e^{\frac{2\Delta t}{\epsilon^2}}(u_{ij}^*)^2. \tag{13}$$

We obtain the following inequality:

$$\begin{aligned}\|\tilde{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 &= \|\left(\mathcal{N} \circ \mathcal{L}\right)(u_{ij}^n) - \left(\mathcal{N}_h \circ \mathcal{L}_h\right)(u_{ij}^n)\|_2 \\ &\le \|\left(\mathcal{N} \circ \mathcal{L}\right)(u_{ij}^n) - \left(\mathcal{N}_h \circ \mathcal{L}\right)(u_{ij}^n)\|_2 + \|\left(\mathcal{N}_h \circ \mathcal{L}\right)(u_{ij}^n) - \left(\mathcal{N}_h \circ \mathcal{L}_h\right)(u_{ij}^n)\|_2 \\ &= \|\left(\mathcal{N}_h \circ \mathcal{L}\right)(u_{ij}^n) - \left(\mathcal{N}_h \circ \mathcal{L}_h\right)(u_{ij}^n)\|_2.\end{aligned}$$

The last step is due to the fact that $\mathcal{N}_h$ is solved analytically, resulting in $\mathcal{N}_h = \mathcal{N}$. Using Eq. (13), we obtain

$$\|\tilde{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 \le e^{\frac{\Delta t}{\epsilon^2}}\|\mathcal{L}(u_{ij}^n) - \mathcal{L}_h(u_{ij}^n)\|_2.$$

By Eq. (12), the following inequality is satisfied.

$$\|\tilde{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 \le e^{\frac{\Delta t}{\epsilon^2}}O(\Delta t + h^2) \le e^{\frac{T}{\epsilon^2}}O(\Delta t + h^2).$$

That is, the following inequality holds.

$$\|\tilde{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 \le C\left(\Delta t + h^2\right). \quad \square$$

Theorem 3. *Suppose that the exact solution* $\hat{u}^{n+1}$ *is smooth and the initial value is smooth and bounded by* 1. *If* $\alpha \le 1/4$, *then for all* $n = 1, 2, \ldots,$

$$\|\hat{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 \le C\left(\Delta t + h^2\right).$$

Proof. For $n \ge 0$, we have

$$\|\hat{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 \le \|\hat{u}_{ij}^{n+1} - \tilde{u}_{ij}^{n+1}\|_2 + \|\tilde{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2. \tag{14}$$

The first term on the right-hand side of the above inequality is bounded by

$$\|\hat{u}_{ij}^{n+1} - \tilde{u}_{ij}^{n+1}\|_2 \le C_1\Delta t.$$

Then, by Theorem 2, we obtain the following inequality for the second term

$$\|\tilde{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 \le C_2\left(\Delta t + h^2\right).$$

Therefore, inequality (14) satisfies

$$\|\hat{u}_{ij}^{n+1} - u_{ij}^{n+1}\|_2 \le C\left(\Delta t + h^2\right),$$

where $C = \max\{C_1,\ C_2\}$ is a generic constant independent of Δt and h. □

5. Numerical experiments

In this section, various numerical tests are performed to validate our proposed method. The proposed method is based on the operator splitting method. Thus, we consider each step for Eqs. (5) and (6). We define the discrete total energy for the diffusion equation, the discrete maximum value, and the discrete minimum value as

$$
\begin{aligned}
&\mathcal{E}_h^D(u^n) = \\
&\quad \frac{h^2}{4}\left[\sum_{j=1}^{3N}\left(\left(\frac{u_{1,j}^n - u_{0,j}^n}{h}\right)^2 + 2\sum_{i=1}^{N-1}\left(\frac{u_{i+1,j}^n - u_{ij}^n}{h}\right)^2 + \left(\frac{u_{N+1,j}^n - u_{N,j}^n}{h}\right)^2\right)\right. \\
&\quad + \sum_{i=1}^{N}\left(\left(\frac{u_{i,1}^n - u_{i,0}^n}{h}\right)^2 + 2\sum_{j=1}^{3N-1}\left(\frac{u_{i,j+1}^n - u_{ij}^n}{h}\right)^2 + \left(\frac{u_{i,3N+1}^n - u_{i,3N}^n}{h}\right)^2\right) \\
&\quad + \sum_{j=N+1}^{2N}\left(\left(\frac{u_{N+1,j}^n - u_{N,j}^n}{h}\right)^2 + 2\sum_{i=N+1}^{4N-1}\left(\frac{u_{i+1,j}^n - u_{ij}^n}{h}\right)^2 + \left(\frac{u_{4N+1,j}^n - u_{4N,j}^n}{h}\right)^2\right) \\
&\quad \left. + \sum_{i=N+1}^{4N}\left(\left(\frac{u_{i,N+1}^n - u_{i,N}^n}{h}\right)^2 + 2\sum_{j=N+1}^{2N-1}\left(\frac{u_{i,j+1}^n - u_{ij}^n}{h}\right)^2 + \left(\frac{u_{i,2N+1}^n - u_{i,2N}^n}{h}\right)^2\right)\right], \\
&Max(u^n) = \max_{(i,j)\in I^d} u_{ij}^n, \quad Min(u^n) = \min_{(i,j)\in I^d} u_{ij}^n.
\end{aligned}
$$

The discrete total energy for the diffusion equation by boundary conditions is then rewritten as:

$$
\begin{aligned}
\mathcal{E}_h^D(u^n) = \frac{1}{2}&\left[\sum_{j=1}^{3N}\sum_{i=1}^{N-1}\left(u_{i+1,j}^n - u_{ij}^n\right)^2 + \sum_{j=N+1}^{2N}\sum_{i=N}^{4N}\left(u_{i+1,j}^n - u_{ij}^n\right)^2\right. \\
&\left. + \sum_{j=1}^{3N-1}\sum_{i=1}^{N}\left(u_{i,j+1}^n - u_{ij}^n\right)^2 + \sum_{j=N}^{2N}\sum_{i=N+1}^{4N}\left(u_{i,j+1}^n - u_{ij}^n\right)^2\right].
\end{aligned}
$$

In addition, normalized discrete total energy is defined as $\mathcal{E}_h(u^n)/\mathcal{E}_h(u^0)$. Then, we define the discrete total energy of the nonlinear term of the AC equation as follows:

$$
\mathcal{E}_h^{\mathcal{N}}(u^n) = \frac{h^2}{4\epsilon^2}\sum_{(i,j)\in I^d}\left[(u_{ij}^n)^2 - 1\right]^2. \tag{15}
$$

Therefore, we obtain the discrete total energy of AC equation.

$$
\mathcal{E}_h^{AC}(u^n) = \mathcal{E}_h^{\mathcal{N}}(u^n) + \mathcal{E}_h^D(u^n). \tag{16}
$$

In Section 5.1, we consider the diffusion equation (3) as the linear term of the AC equation. Furthermore, we consider the AC equation (2) in Section 5.2.

5.1. Diffusion equation

The discretization of the diffusion equation (5) using the explicit Euler's method is Eq. (7) with $u_{ij}^{n+1} = u_{ij}^*$ on Ω^d.

$$
\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t} = \frac{u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4u_{ij}^n}{h^2}. \tag{17}
$$

We perform numerical experiments to observe the appropriate boundary conditions, which is an important point of the proposed method.

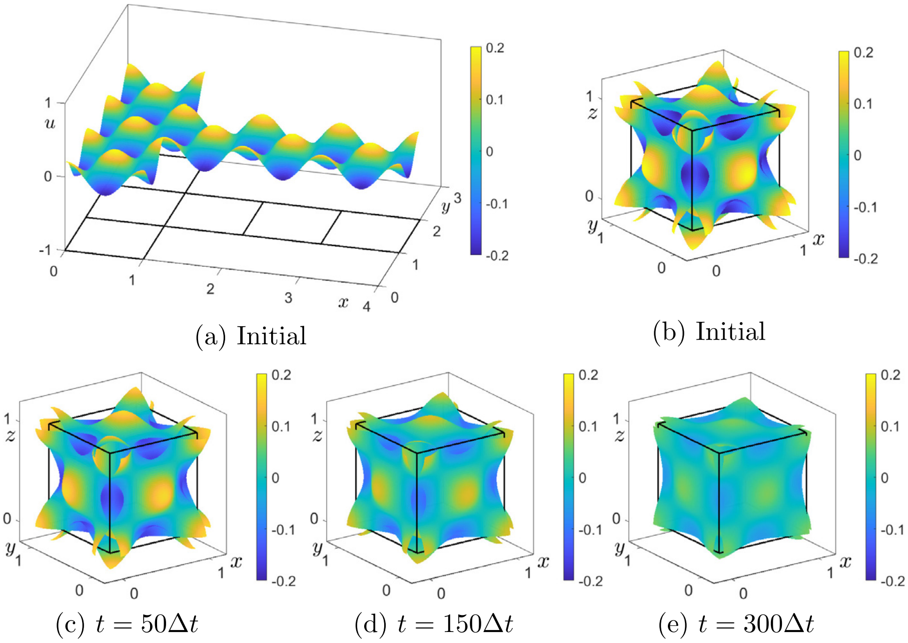
5.1.1. Convergence test

We verify the consistency between the numerical and exact solutions by considering the following initial condition in the domain Ω:

$$
u(x, y, 0) = \beta\cos(2\pi x)\cos(2\pi y). \tag{18}
$$

Table 1
L^2-norms of numerical error for different space steps and the corresponding rate of convergence.

h	1/32	1/64	1/128	1/256	1/512	1/1024
L^2-norm	4.908e−5	1.217e−5	2.901e−6	7.333e−7	1.830e−7	4.579e−8
Rate		2.01	2.07	1.98	2.00	2.00



(a) Initial

(b) Initial

(c) $t = 50\Delta t$

(d) $t = 150\Delta t$

(e) $t = 300\Delta t$

Fig. 6. The initial condition with $h = 1/64$ and a snapshot of numerical solutions. (a) Initial condition on the unfolded cubic surface, (b) Initial condition on the cubic surface, (c) $t = 50\Delta t$, (d) $t = 150\Delta t$, (e) $t = 300\Delta t$.

The exact solution is given as

$$u_{exact}(x, y, t) = \beta \cos(2\pi x) \cos(2\pi y) \mathrm{e}^{-8\beta\pi^2 t}.$$

Here, different space steps $h = 1/32,\ 1/64,\ 1/128,\ 1/256,\ 1/512,\ 1/1024$, time step $\Delta t = 0.25h^2$, and amplitude $\beta = 0.2$ are used. We define the numerical error with a space step h on Ω^d as $e_{ij}^h = u_{ij}^{N_t} - u(x_i, y_j, T)$, where T is the final time and $N_t = T/\Delta t$. The L^2-norm of the numerical error is then given by

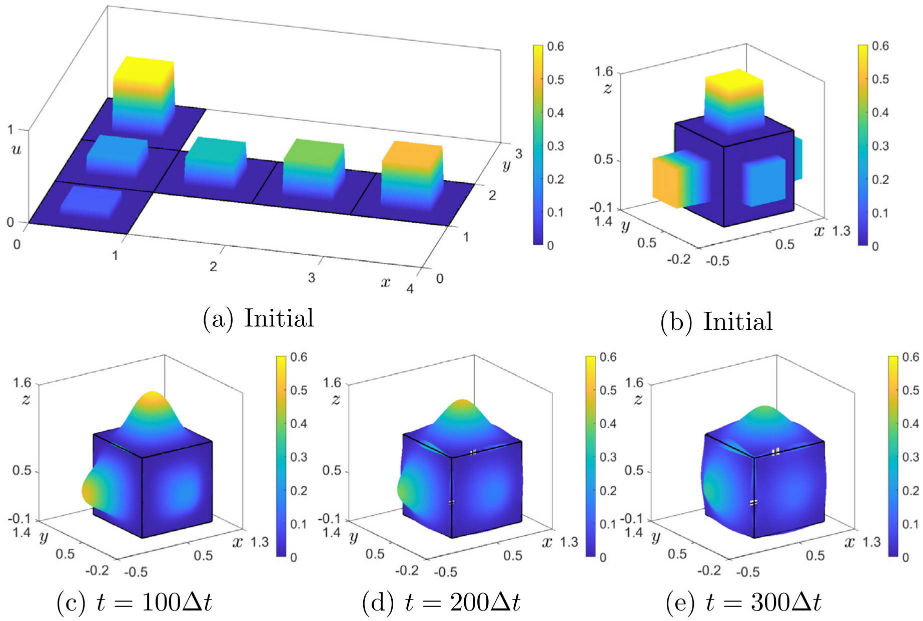
$$\|e^h\|_2 = \sqrt{\frac{1}{6N^2} \sum_{(i,j)\in I^d} (e_{ij}^h)^2},$$

The rate of convergence is calculated as $\log_2(\|e^h\|_2/\|e^{h/2}\|_2)$. Table 1 lists the discrete L^2-norms of numerical error and rate of convergence at $T = 0.001$. We observe that the numerical solutions solved by the proposed method for different space steps are second-order accurate. This implies that the scheme is first-order accurate in time and second-order accurate in space.

Fig. 6(a) shows the initial condition given by Eq. (18) with space step $h = 1/64$. Figs. 6(b)–6(e) show snapshots of the numerical solutions on the cubic surface at $t = 0,\ 50\Delta t,\ 150\Delta t,\ 300\Delta t$, respectively. By calculating the discrete diffusion equation for the initial condition, it can be observed that the properties of the diffusion equation are satisfied.

5.1.2. Numerical solutions for various initial conditions

We perform numerical simulations to verify that the proposed algorithm is an appropriate boundary condition. We consider three cases for the initial condition. In the first and second tests, we consider a simple initial conditions



(a) Initial

(b) Initial

(c) $t = 100\Delta t$

(d) $t = 200\Delta t$

(e) $t = 300\Delta t$

Fig. 7. The initial condition and a snapshot of numerical solutions. (a) Initial condition on the unfolded cubic surface, (b) Initial condition on the cubic surface, (c) $t = 100\Delta t$, (d) $t = 200\Delta t$, (e) $t = 300\Delta t$.

and use the same parameters $h = 1/64$, $\Delta t = 0.25h^2$, and $T = 300\Delta t$. Fig. 7 shows the results of the numerical simulation when the initial condition $u(x_i, y_j, 0) = 0.1k$ in the square region on each domain Ω_k^d, $k = 1, 2, \ldots, 6$. Figs. 7(c)–7(e) show the results at time $t = 100\Delta t$, $t = 200\Delta t$, $t = 300\Delta t$. We observe that the numerical solution on each domain Ω_k^d, $k = 1, 2, \ldots, 6$ diffuses faster or slower depending on k, while simultaneously affecting the numerical solutions in other Ω_l^d where $l \neq k$.

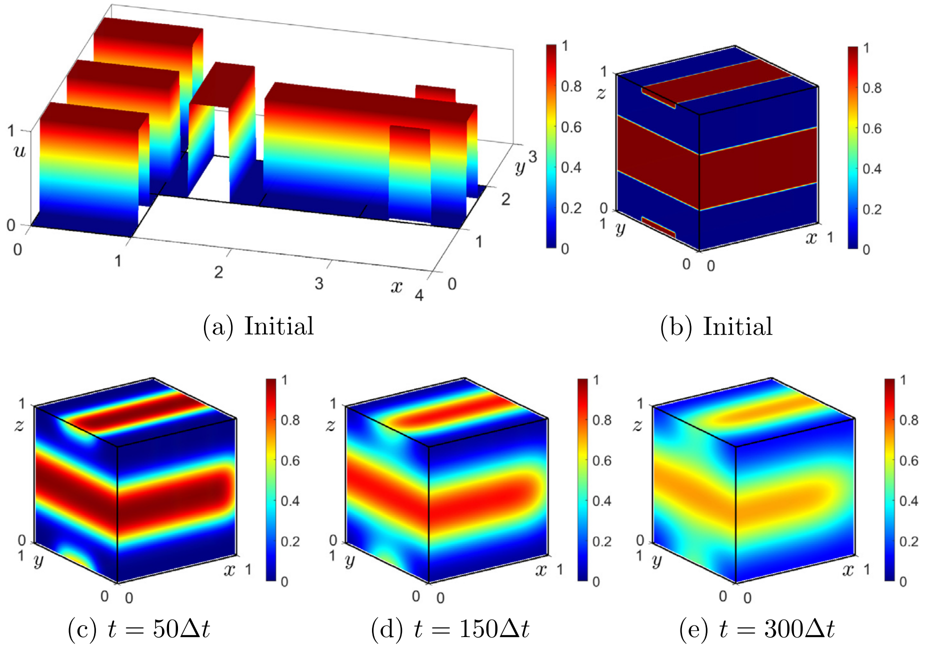
The second simulation considers the case where a connected phase field exists on two or more Ω_k, $k = 1, 2, \ldots, 6$, and this is an initial condition to verify whether the boundary conditions of the proposed method are appropriate. Figs. 8(a) and 8(b) show the initial condition on the unfolded cubic surface and the initial condition on the cubic surface, respectively. Figs. 8(c)–8(e) show the snapshot of the numerical solution on the cubic surface at times $t = 50\Delta t$, $150\Delta t$, and $300\Delta t$, respectively. The results demonstrate that the proposed algorithm accurately performs calculations near each edge through the implementation of appropriate boundary conditions.

The third initial condition is a random perturbation as $u(x, y, 0) = \text{rand}(x, y)$. Here, rand (x, y) is a random value between -1 and 1. The parameters used for this simulation are $N = 64$, $h = 1/N$, $\Delta t = 0.01h^2$, and $T = 120\Delta t$. For the diffusion equation, both the total free energy function and the maximum value of the solution decrease.

In Fig. 9, the left column and right column show the numerical solution for the unfolded cubic surface and 10 contours of the numerical solution on the cubic surface, respectively. Figs. 9(a)–9(d) show the numerical solution on the unfolded and the cubic surface at times $t = 0$, $40\Delta t$, $80\Delta t$, and $120\Delta t$, respectively. Here, we demonstrate that the numerical solution gradually diffuses and becomes uniform. Therefore, we can observe that the boundary conditions in each domain Ω_k^d, $k = 1, \ldots, 6$ are satisfied and the numerical solutions of the interface between different domains appeared naturally. In Fig. 10, we solve for Eq. (3) using the proposed method and observe that the total energy decreases and the discrete maximum principle is satisfied.

5.2. The Allen–Cahn equation

In this section, we confirm that the proposed method accurately solves the AC equation for a cubic surface domain and investigate the dynamics and properties of the AC equation.



(a) Initial (b) Initial

(c) $t = 50\Delta t$ (d) $t = 150\Delta t$ (e) $t = 300\Delta t$

Fig. 8. The initial condition and the numerical solutions with $t = 0,\ 50\Delta t,\ 150\Delta t,\ 300\Delta t$.

5.2.1. *Properties of the AC equation*

The AC equation satisfies the total energy decreasing property and the maximum principle [16]. We perform numerical experiments to verify that the total energy decreasing property and the maximum principle of the AC equation are satisfied for the random initial condition $u(x, y, 0)$ =rand(x, y) with the following parameters: $N = 128$, $h = 1/N$, $\Delta t = 0.2h^2$, $T = 500\Delta t$, and $\epsilon = \epsilon_8$. Here, we consider the interface parameter $\epsilon_m = mh/(2\sqrt{2}\tanh^{-1}(0.9))$, which means setting m grid points behavior on the interface. Fig. 11(a) shows the initial condition on the unfolded cubic surface and the cubic surface. In Fig. 11, each row from top to bottom represents the numerical solution for Eq. (2). Figs. 11(b)–11(d) show the results at the times $100\Delta t$, $150\Delta t$, and $500\Delta t$, respectively. In Fig. 12, we observe that the discrete AC equation solved by the proposed method satisfies the discrete total energy decreasing property and maximum principle.

5.2.2. *Motion by mean curvature*

In 2D space, the normal velocity of circular interface satisfies the following geometric law [27].

$$V = -\kappa = -\frac{1}{R},$$

where V is the velocity, κ is the curvature, and R is the radius. If R_0 is the initial radius, the analytic solution can be expressed as $R(t) = \sqrt{R_0^2 - 2t}$. To verify the proposed method, we consider the following initial condition on Ω^d.

$$u_1(x_i, y_j, 0) = \begin{cases} \tanh\left(\frac{r_1-\sqrt{x_i^2+(y_j-2)^2}}{\sqrt{2}\epsilon}\right), & (x_i, y_j) \in \Omega_2^d \cup \Omega_6^d, \\ \tanh\left(\frac{r_1-\sqrt{(x_i-4)^2+(y_j-2)^2}}{\sqrt{2}\epsilon}\right), & (x_i, y_j) \in \Omega_5^d, \end{cases}$$

$$u_2(x_i, y_j, 0) = \begin{cases} \tanh\left(\frac{r_2-\sqrt{(x_i-0.5)^2+(y_j-2.5)^2}}{\sqrt{2}\epsilon}\right), & (x_i, y_j) \in \Omega_6^d, \end{cases}$$

$$u_3(x_i, y_j, 0) = \begin{cases} \tanh\left(\frac{r_3-\sqrt{(x_i-0.5)^2+(y_j-2)^2}}{\sqrt{2}\epsilon}\right), & (x_i, y_j) \in \Omega_6^d \end{cases}$$

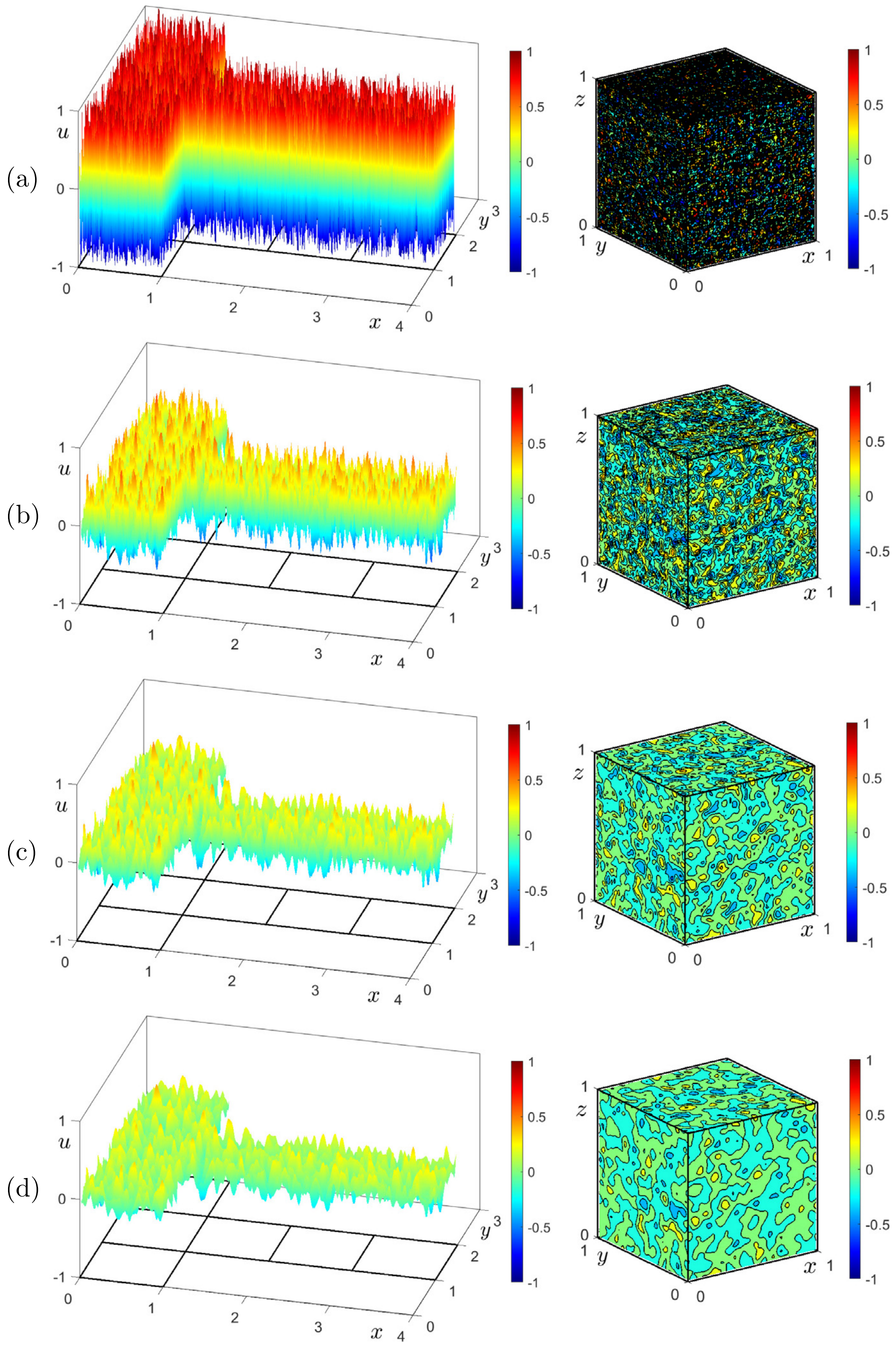


Fig. 9. The initial condition and numerical solutions at the time $t = 40\Delta t,\ 80\Delta t,\ 120\Delta t$. Each row (a)–(d) is the result at time $t = 0,\ 40\Delta t,\ 80\Delta t,\ 120\Delta t$, respectively.

where r_1, r_2, and r_3 are the initial radii. We used the parameters $r_1 = 0.5$, $r_2 = 0.45$, $r_3 = 0.4$, $N = 128$, $h = 1/N$, $\Delta t = 0.2h^2$, $T = 10000\Delta t$, and $\epsilon = \epsilon_8$. The initial conditions are depicted in Fig. 13(a). Fig. 13(b) shows snapshots of the zero-contour levels of the numerical solutions for the initial conditions u_1, u_2, and u_3, displayed from left to

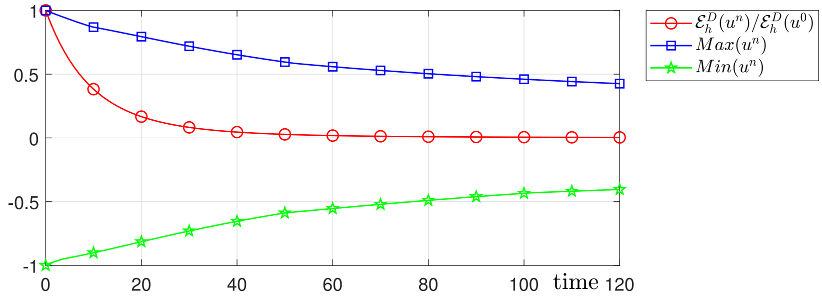


Fig. 10. The normalized discrete total energy, maximum, and minimum values.

right, respectively. Fig. 13(c) shows the temporal evolution of the analytic and numerical radii over time t. From the results shown in Fig. 13, we observed that the numerical solutions for the AC equation, obtained using the proposed method, satisfy the motion by mean curvature.

To confirm the flow of AC equation (2) on the global cubic surface, we set the initial condition as follows:

$$u(x_i, y_j, 0) = \begin{cases} \tanh\left(\frac{r-\sqrt{(x_i-0.5)^2+(y_j-2.5)^2}}{\sqrt{2}\epsilon}\right), & (x_i, y_j) \in \Omega_2^d \cup \Omega_6^d, \\ \tanh\left(\frac{r-\sqrt{(x_i-1.5)^2+(y_j-2.5)^2}}{\sqrt{2}\epsilon}\right), & (x_i, y_j) \in \Omega_3^d, \\ \tanh\left(\frac{r-\sqrt{(x_i-2.5)^2+(y_j-2.5)^2}}{\sqrt{2}\epsilon}\right), & (x_i, y_j) \in \Omega_4^d, \\ \tanh\left(\frac{r-\sqrt{(x_i-3.5)^2+(y_j-2.5)^2}}{\sqrt{2}\epsilon}\right), & (x_i, y_j) \in \Omega_5^d. \end{cases}$$

We used the parameters $N = 128$, $h = 1/N$, $\Delta t = 0.2h^2$, $T = 17000\Delta t$, and $\epsilon = \epsilon_8$. In Fig. 14, we consider two different initial conditions with radii $r = 0.65$ and $r = 1$. The initial conditions are depicted in Fig. 14(a). Fig. 14(b) shows snapshots or zero-contour levels of the numerical solutions for $r = 0.65$ and $r = 1$ from left to right. On the cubic surface, the solution of the AC equation locally follows the mean curve flow but exhibits a distinct dynamic globally.

5.2.3. Numerical solution for initial conditions

The initial condition to be considered first is given by:

$$u(x_i, y_j, 0) = \begin{cases} 1 & \text{if } y_j > 0.3\sin(2\pi x_i) + 1.5 \\ -1 & \text{otherwise,} \end{cases} \quad (x_i, y_j) \in \Omega^d,$$

which is shown in Fig. 15(a). We use the parameters $N = 128$, $h = 1/N$, $\epsilon = \epsilon_8$, $\Delta t = 0.2h^2$, and $T = 10000\Delta t$. Figs. 13(b), 13(c), and 13(d) show the numerical solutions on the cubic surface at $t = 1000\Delta t$, $2000\Delta t$, and $10000\Delta t$, respectively. We observed that the numerical solution at $t = 10000\Delta t$ approximates the equilibrium solution.

Next, we consider the following two initial conditions to investigate the behavior of a traveling wave.

$$u_4(x_i, y_j, 0) = \begin{cases} 1 & \text{if } \sqrt{(x_i-0.4)^2+(y_j-2.2)^2} < 0.2 \\ 0 & \text{otherwise,} \end{cases} \quad (x_i, y_j) \in \Omega^d,$$

$$u_5(x_i, y_j, 0) = \begin{cases} 1 & \text{if } \sqrt{(x_i-0.4)^2+(y_j-2.3)^2} < 0.1 \\ 1 & \text{if } \sqrt{(x_i-0.6)^2+(y_j-1.7)^2} < 0.05 \\ 1 & \text{if } \sqrt{(x_i-2.6)^2+(y_j-1.3)^2} < 0.2 \\ 1 & \text{if } \sqrt{(x_i-3.2)^2+(y_j-1.7)^2} < 0.05 \\ 1 & \text{if } \sqrt{(x_i-0.7)^2+(y_j-0.5)^2} < 0.05 \\ 0 & \text{otherwise,} \end{cases} \quad (x_i, y_j) \in \Omega^d,$$

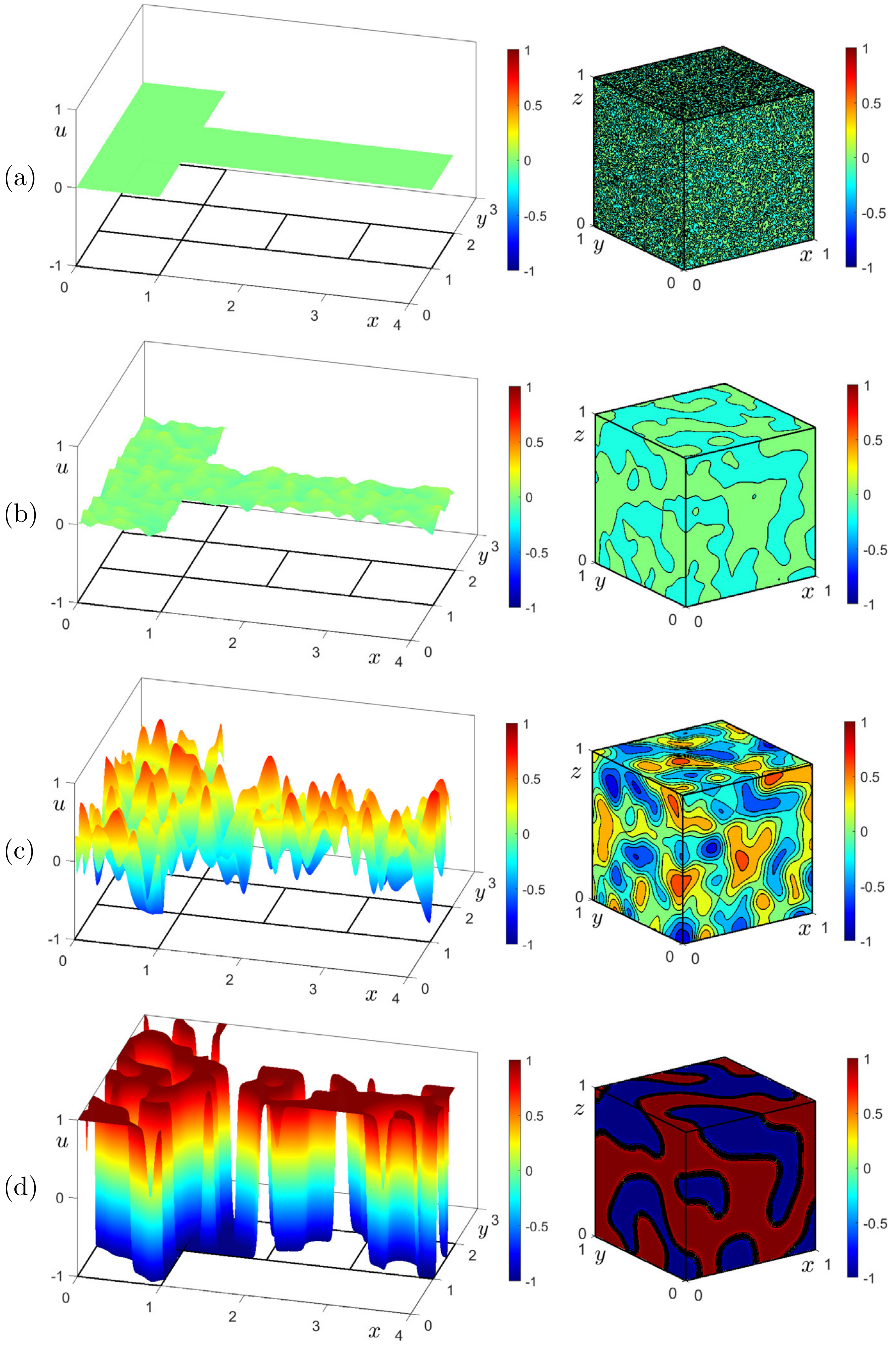


Fig. 11. The initial condition and numerical solutions at the time $t = 100\Delta t,\ 150\Delta t,\ 500\Delta t$.

Figs. 16(a) and 16(b) show the numerical solutions for the initial conditions u_4 and u_5, respectively. For the numerical experiment, we used the parameters $N = 128$, $h = 1/N$, $\epsilon = \epsilon_8$, $\Delta t = 0.2h^2$, and $T = 800\Delta t$. Fig. 16 shows the temporal evolution of the dynamics for the AC equation. From left to right, we show the results at the times $t = 200\Delta t,\ 500\Delta t,\ 800\Delta t$. In order to observe the traveling wave, the AC equation is solved using the proposed method with the given initial conditions. We observed the motion of the transition layer in the results of the numerical experiments.

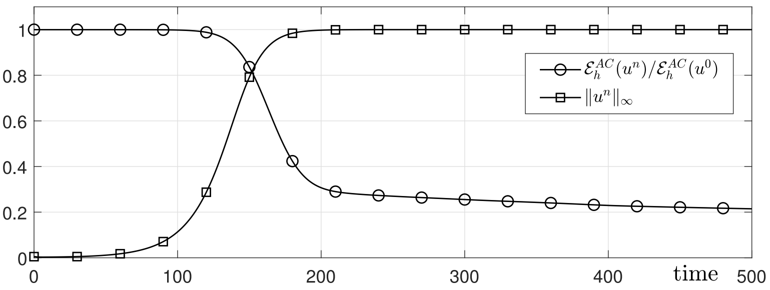


Fig. 12. The normalized discrete total energy and maximum norm of numerical solution for the AC equation.

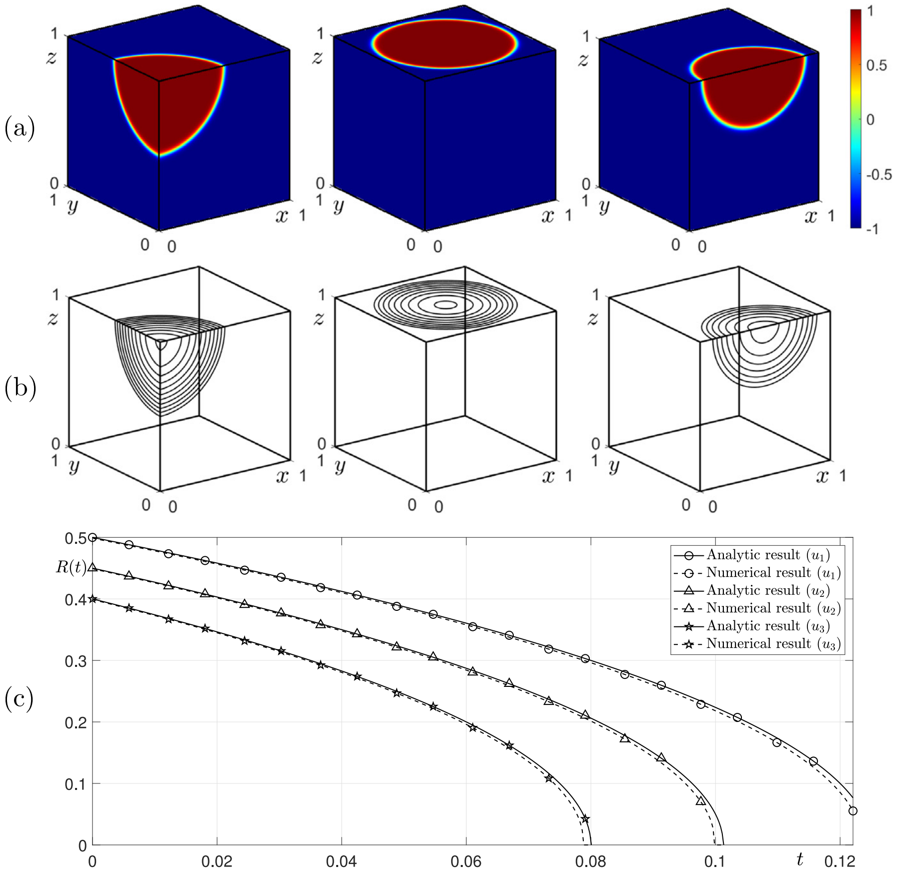


Fig. 13. Motion by mean curvature with AC equation. (a) The initial conditions u_1, u_2, and u_3 on the cubic surface domain. (b) Snapshot of the zero-level contour of numerical solutions at the time $t = 0$, $1000\Delta t$, …, and $10000\Delta t$. (c) Temporal evolution of the radii.

6. Conclusions

We presented FDM for solving the AC equation on cubic surfaces. In order to solving the AC equation on the cubic surface, firstly, the cubic surface in 3D space was expanded into six planar sub-domains with appropriate boundary conditions in 2D space. Next, FDM was applied to the planar sub-domain. We solved the AC equation by splitting it into a linear term and a nonlinear term using the operator splitting method. We verified that the boundary condition of the proposed method is an appropriate boundary condition on the cubic surface through numerical experiments on the diffusion equation, which is the linear term of the AC equation. Through numerical experiments on the proposed method, we observed the properties of the AC equation on the cubic surface domain in

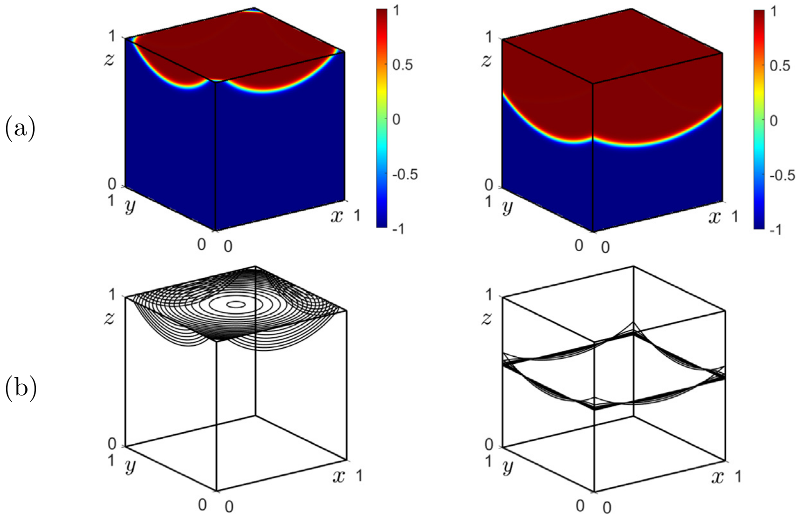
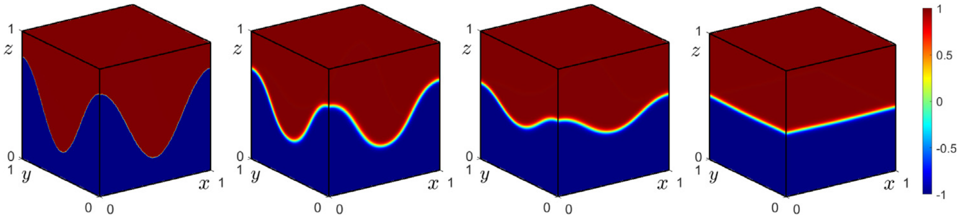
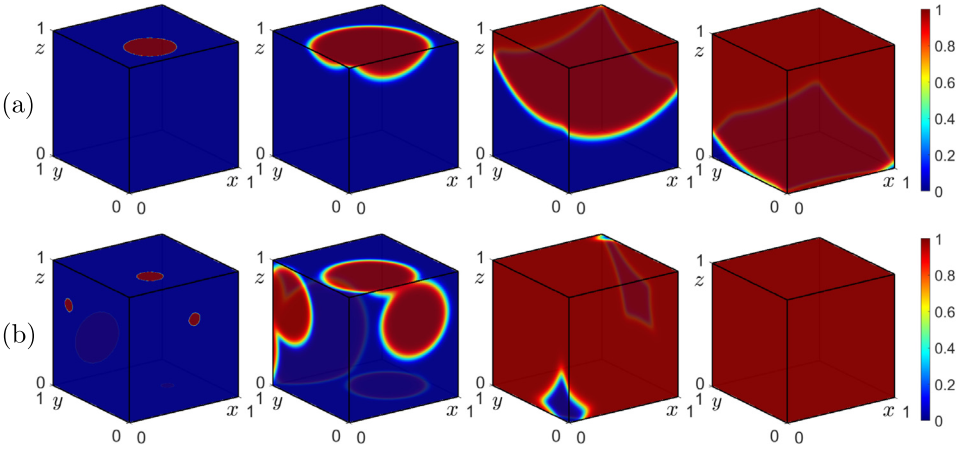


Fig. 14. Dynamics of AC equation on the cubic surface. (a) The initial conditions with $r = 0.65$ and $r = 1$ on the cubic surface domain. (b) Snapshot of the zero-level contour of numerical solutions at the time $t = 0$, $1000\Delta t$, $\ldots$, and $17000\Delta t$.



(a) $t = 0$ (b) $t = 1000\Delta t$ (c) $t = 3000\Delta t$ (d) $t = 10000\Delta t$

Fig. 15. Temporal evolution of the dynamics of the AC equation.



$t = 0$ $t = 200\Delta t$ $t = 500\Delta t$ $t = 800\Delta t$

Fig. 16. The initial condition and numerical solutions for the two initial conditions (a) u_1. (b) u_2 at the time $t = 200\Delta t,\ 500\Delta t,\ 800\Delta t$ from left to right.

3D. The AC equation on the cubic surface domain solved by the proposed method showed different characteristics than on the planar domain in 2D space. We conducted a theoretical analysis on the discrete maximum principle and error of the proposed method and verified it through numerical experiments. Additionally, the discrete energy decay was verified through numerical experiments. The energy dissipation analysis for the operator splitting method is a non-trivial problem. As future work, we plan to analyze the energy decreasing properties of the operator splitting method. In this paper, we focused on validating a numerical method and considering appropriate boundaries to solve cubic surfaces in 2D space. In future studies, the proposed method will be combined with various numerical methods such as the implicit Euler method, Crank–Nicolson method, and multigrid method to efficiently solve the AC equation and the reaction–diffusion equation on the cubic surfaces.

Declaration of competing interest

None declared.

Acknowledgments

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 are grateful to the referees whose comments greatly improved the paper.

References

[1] S.M. Allen, J.W. Cahn, A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening, Acta Metall. 27 (6) (1979) 1085–1095.

[2] M. Beneš, V. Chalupecký, K. Mikula, Geometrical image segmentation by the Allen–Cahn equation, Appl. Numer. Math. 51 (2–3) (2004) 187–205.

[3] M. Cheng, J.A. Warren, An efficient algorithm for solving the phase field crystal model, J. Comput. Phys. 227 (12) (2008) 6241–6248.

[4] Y. Choi, D. Jeong, S. Lee, M. Yoo, J. Kim, Motion by mean curvature of curves on surfaces using the Allen–Cahn equation, Internat. J. Engrg. Sci. 97 (2015) 126–132.

[5] Y. Choi, Y. Li, C. Lee, H. Kim, J. Kim, Explicit hybrid numerical method for the Allen–Cahn type equations on curved surfaces, Numer. Math. Theory Methods Appl. 14 (3) (2021) 797–810.

[6] D. Deng, Q. Wang, A class of weighted energy-preserving Du Fort–Frankel difference schemes for solving sine-Gordon-type equations, Commun. Nonlinear Sci. Numer. Simul. 117 (2023) 106916.

[7] D. Deng, Z. Zhao, Efficiently energy-dissipation-preserving ADI methods for solving two-dimensional nonlinear Allen–Cahn equation, Comput. Math. Appl. 128 (2022) 249–272.

[8] S. Dipierro, J. Serra, E. Valdinoci, Improvement of flatness for nonlocal phase transitions, Am. J. Math. 142 (4) (2020) 1083–1160.

[9] Q. Du, R.B. Lehoucq, A.M. Tartakovsky, Integral approximations to classical diffusion and smoothed particle hydrodynamics, Comput. Methods Appl. Mech. Engrg. 286 (2015) 216–229.

[10] L. Golubović, A. Levandovsky, D. Moldovan, Interface dynamics and far-from-equilibrium phase transitions in multilayer epitaxial growth and erosion on crystal surfaces: Continuum theory insights, East Asian J. Appl. Math. 1 (4) (2011) 297–371.

[11] Z. Guan, J.S. Lowengrub, C. Wang, S.M. Wise, Second order convex splitting schemes for periodic nonlocal Cahn–Hilliard and Allen–Cahn equations, J. Comput. Phys. 277 (2014) 48–71.

[12] D. He, K. Pan, Maximum norm error analysis of an unconditionally stable semi-implicit scheme for multi-dimensional Allen–Cahn equations, Numer. Methods Partial Differential Equations 35 (3) (2019) 955–975.

[13] L. Ivan, H. De Sterck, A. Susanto, C.P. Groth, High-order central ENO finite-volume scheme for hyperbolic conservation laws on three-dimensional cubed-sphere grids, J. Comput. Phys. 282 (2015) 157–182.

[14] D.A. Kay, A. Tomasi, Color image segmentation by the vector-valued Allen–Cahn phase-field model: a multigrid solution, IEEE Trans. Image Process. 18 (10) (2009) 2330–2339.

[15] H.G. Lee, J. Kim, A simple and efficient finite difference method for the phase-field crystal equation on curved surfaces, Comput. Methods Appl. Mech. Engrg. 307 (2016) 32–43.

[16] S. Lee, S. Yoon, C. Lee, S. Kim, H. Kim, J. Yang, S. Kwak, Y. Hwang, J. Kim, Effective time step analysis for the Allen–Cahn equation with a high-order polynomial free energy, Internat. J. Numer. Methods Engrg. 123 (19) (2022) 4726–4743.

[17] J.L. McGregor, Semi-Lagrangian advection on conformal-cubic grids, Mon. Weather Rev. 124 (6) (1996) 1311–1322.

[18] V. Mohammadi, D. Mirzaei, M. Dehghan, Numerical simulation and error estimation of the time-dependent Allen–Cahn equation on surfaces with radial basis functions, J. Sci. Comput. 79 (1) (2019) 493–516.

[19] W.M. Putman, S.J. Lin, Finite-volume transport on various cubed-sphere grids, J. Comput. Phys. 227 (1) (2007) 55–78.

[20] C. Ronchi, R. Iacono, P.S. Paolucci, The cubed sphere: A new method for the solution of partial differential equations in spherical geometry, J. Comput. Phys. 124 (1) (1996) 93–114.

[21] R. Sadourny, Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids, Mon. Weather Rev. 100 (2) (1972) 136–144.

[22] M. Sun, X. Feng, K. Wang, Numerical simulation of binary fluid–surfactant phase field model coupled with geometric curvature on the curved surface, Comput. Methods Appl. Mech. Engrg. 367 (2020) 113123.

[23] M. Taylor, J. Tribbia, M. Iskandarani, The spectral element method for the shallow water equations on the sphere, J. Comput. Phys. 130 (1) (1997) 92–108.

[24] Z. Weng, Q. Zhuang, Numerical approximation of the conservative Allen–Cahn equation by operator splitting method, Math. Methods Appl. Sci. 40 (12) (2017) 4462–4480.

[25] X. Xiao, X. Feng, A second-order maximum bound principle preserving operator splitting method for the Allen–Cahn equation with applications in multi-phase systems, Math. Comput. Simulation 202 (2022) 36–58.

[26] J. Yang, J. Kim, Numerical approximation of the square phase-field crystal dynamics on the three-dimensional objects, J. Comput. Phys. 471 (2022) 111652.

[27] J. Yang, Y. Li, C. Lee, Y. Choi, J. Kim, Fast evolution numerical method for the Allen–Cahn equation, J. King Saud Univ. Sci. 35 (1) (2023) 102430.

[28] 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) 1–28.

[29] L.Y. Yu, The stochastic web on a spherical surface generated by simple, 3-dimensional rotational transformations, Commun. Nonlinear Sci. Numer. Simul. 17 (7) (2012) 2905–2913.

[30] S. Zhai, X. Feng, Y. He, Numerical simulation of the three dimensional Allen–Cahn equation by the high-order compact ADI method, Comput. Phys. Comm. 185 (10) (2014) 2449–2455.

[31] J. Zhang, L.L. Wang, Z. Rong, A prolate-element method for nonlinear PDEs on the sphere, J. Sci. Comput. 47 (1) (2011) 73–92.