



Research article

A simple and efficient numerical method for the Allen–Cahn equation on effective symmetric triangular meshes

Youngjin Hwang, Seokjun Ham, Chaeyoung Lee, Gyeonggyu Lee, Seungyoon Kang and Junseok Kim*

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

* **Correspondence:** Email: cfdkim@korea.ac.kr.

Abstract: In this paper, we propose a novel, simple, efficient, and explicit numerical method for the Allen–Cahn (AC) equation on effective symmetric triangular meshes. First, we compute the net vector of all vectors starting from each node point to its one-ring neighbor vertices and virtually adjust the neighbor vertices so that the net vector is zero. Then, we define the values at the virtually adjusted nodes using linear and quadratic interpolations. Finally, we define a discrete Laplace operator on triangular meshes. We perform several computational experiments to demonstrate the performance of the proposed numerical method for the Laplace operator, the diffusion equation, and the AC equation on triangular meshes.

Keywords: diffusion equation; discrete Laplace operator; triangular meshes; Allen–Cahn equation

1. Introduction

In this study, we present a novel explicit numerical method for solving the Allen–Cahn equation on triangular meshes. The method is characterized by its simplicity and efficiency. The AC equation, a partial differential equation (PDE), is employed in mathematical physics and materials science to represent the dynamics of phase transitions and pattern formation in binary systems [1]. The equation incorporates a free energy density function that characterizes the potential energy of the system and a linear diffusion term that drives the system toward a state of minimum energy [2]. In two-dimensional (2D) space, the AC equation with Dirichlet boundary condition on the domain Ω as follows

$$\begin{aligned}\frac{\partial u(x, y, t)}{\partial t} &= -\frac{F'(u(x, y, t))}{\epsilon^2} + \Delta u(x, y, t), \quad (x, y) \in \Omega, \quad t > 0, \\ u(x, y, t) &= f(x, y) \text{ on } \partial\Omega,\end{aligned}\tag{1.1}$$

where $u(x, y, t)$ is the phase-field function, ϵ is an interface transition layer related parameter, $F(u) = 0.25(u^2 - 1)^2$ is a double-well potential energy functional, and Δ is a Laplace operator.

Choi et al. [1] developed an unconditionally gradient-stable numerical system for the AC equation on uniformly square grid and showed the same property for the corresponding discrete problem by using the eigenvalues of the Hessian matrix of the energy functional. In [3], the authors developed a simple and practical adaptive finite difference scheme for AC equations in 2D. The developed method uses a temporally adaptive narrow band domain in a uniform discrete rectangular domain. Xiao et al. [4] introduced the lumped mass surface finite element method on a triangular mesh to solve the surface AC equation. To maintain mass conservation, Rubinstein and Sternberg [5] introduced a conservative AC equation by incorporating a Lagrange multiplier into the AC equation. Construction of the conservative AC equation addresses the issue of the AC equation inability to conserve total mass. Sun and Zhang [6] developed a meshless radial basis function approach to solve the conservative AC equation on smooth compact surfaces embedded in \mathbb{R}^3 . The developed method ensures mass conservation while employing an operator splitting scheme. The Laplace–Beltrami operator (LBO) is approximated iteratively by using radial basis functions. In addition, the time discretization of the diffusion equation is achieved using the Euler method. Xia et al. [7] developed a second-order accurate method in time and space for the AC and conservative AC equation using triangular discretization. The AC and conservative AC equation can also be solved using the explicit hybrid numerical method, as studied by Choi et al. [8]. Joshi and Jaiman [9] proposed an adaptive variational procedure for the AC equation with a two-phase field. This scheme is stable, robust, and general for the AC equation on a triangular mesh. Kwak et al. [10] proposed the conservative AC equation with a curvature-dependent Lagrange multiplier. Hong et al. [11] developed numerical algorithms with fully discrete structure-preserving of arbitrarily high order for the AC model with a nonlocal constraint subject to the Neumann boundary condition. Chai et al. [12] proposed a simple multiple-relaxation-time lattice-Boltzmann method for the local and nonlocal AC equation, presenting the property of mass conservation through numerical experiments. Inc et al. [13] performed Lie symmetry analysis, explicit solutions, and convergence analysis for the time-fractional Cahn–Allen equation. Kim et al. [14] proposed a temporally second-order unconditionally energy-stable computational method for the AC equation with high-order polynomial free energy potential. Bhatt et al. [15] devised an attractive and easy-to-implement alternative for integrating the multi-dimensional AC equation with no-flux boundary conditions by combining the Fourier spectral method with the strongly stable exponential time difference method. The method they developed has several key advantages, including the use of discrete fast Fourier transform for efficiency, the ability to extend to two and three spatial dimensions in a manner similar to 1D problems, and the capability to handle various boundary conditions. Sun and Gao [16] proposed a high order multiquadric trigonometric quasi-interpolation method for function approximation and derivative approximation based on periodic sampling data. Furthermore, the method was applied to solve different types of PDEs including the AC equation. Sun and Ling [17] developed a meshless conservative Galerkin method for solving Hamiltonian wave equations, which is a classical example of Hamiltonian PDEs. The equation was first discretized in space using radial basis functions in a Galerkin-type formulation. Their method used appropriate projection operators for the construction of the Galerkin equation and was shown to conserve global energies.

The AC equation without a nonlinear term is a diffusion equation that is a linear differential equation:

$$\frac{\partial u(x, y, t)}{\partial t} = D\Delta u(x, y, t), \quad (1.2)$$

where D is the diffusion coefficient parameter and we set $D = 1$ in this paper for simplicity of exposition.

There exist meshes that do not satisfy the properties of the continuous Laplace operator for continuous functions using the discrete Laplace operator, and it is more difficult to satisfy the properties of the continuous Laplace operator for triangular meshes [18]. The method we propose requires defining a discrete Laplace operator to solve the diffusion term. Thus, a simple and efficient algorithm is included that numerically satisfies the simple properties of the Laplace operator.

Triangular meshes are widely used for representing complex domains or curved surfaces. Until recently, the discrete Laplace operator for triangular meshes has been a subject of numerous studies. Xu [19] proposed discretization schemes of the Laplace-Beltrami operator on triangulated surfaces with curvatures. Meanwhile, Caissard et al. [20] presented a new discretization of the Laplace-Beltrami operator over digital surfaces. Thampi et al. [21] demonstrated schemes for isotropic discrete Laplacian operators based on lattice hydrodynamics. McCartin [22] studied the discrete Laplacian on an equilateral triangle with both Dirichlet and Neumann boundary conditions using eigenvalues and eigenfunctions of the continuous Laplacian. A finite difference algorithm for solving the generalized Laplace equation on unstructured triangular meshes was developed by Ganzha et al. [23] using a support operator method combining discrete divergence and gradient operators. Yoon et al. [24] conducted numerical experiments on triangular meshes to investigate dendritic pattern formation in an isotropic crystal growth model on surfaces, and subsequently presented their findings. Paquet and Korikache [25] established an a priori error estimate for the fully discretized problem of the dual mixed method for the non-stationary heat equation in 2D polygonal domains. The Cahn–Hilliard (CH) equation is one of the famous phase-field models along with the AC equation [26]. Tian et al. [27] developed an adaptive finite element method based on superconvergent cluster recovery to solve the Laplace operator of the CH equation on a triangular mesh. Therefore, the construction of simple and efficient algorithms for solving Laplace operators on triangular meshes is important. We propose a numerical method involving the novel discrete Laplace operator and verify the proposed discrete Laplace operator through numerical experiments, including comparisons to general discrete Laplace operators on rectangular meshes or the LBO on triangular meshes.

The contents of this paper are as follows. In Section 2, the proposed numerical scheme is described. In Section 3, we present the numerical experiments. In Section 4, the conclusion is drawn.

2. Numerical solution algorithm

In this section, we present a numerical solution algorithm. The proposed algorithm solves the AC equation by using an explicit Euler method. The numerical algorithm requires computing a linear term with the Laplace operator. We propose a novel Laplace operator in triangular meshes to numerically compute the AC equation. Firstly, we introduce the discrete LBO on triangular mesh and propose a novel discrete Laplace operator. Because the plane is a special case of the curved surfaces, let us consider the discretization of LBO [19] on curved surfaces. Let N be the number of node points on the triangular mesh, $\mathbf{x}_k = (x_k, y_k)$, $k = 1, 2, \dots, N$ be node points on the triangular mesh, Δt be temporal step size, and u_k^n be an approximation of phase-field function $u(\mathbf{x}_k, n\Delta t)$. In Figure 1, \mathbf{x}_k denotes k -th node point of a triangular mesh and $\{\mathbf{x}_{k_1}, \mathbf{x}_{k_2}, \mathbf{x}_{k_3}, \mathbf{x}_{k_4}, \mathbf{x}_{k_5}\}$ are one-ring node points neighboring \mathbf{x}_k . For simplicity, we assume that there are five node points neighboring \mathbf{x}_k . The discrete LBO at node point

\mathbf{x}_k is given as

$$\Delta u_k \approx \frac{3}{A(\mathbf{x}_k)} \sum_{m=1}^{N_k} \frac{\cot \alpha_{k_m} + \cot \beta_{k_m}}{2} (u_{k_m} - u_k),$$

where $A(\mathbf{x}_k)$ is the sum of areas of triangles T_m sharing the node point \mathbf{x}_k , N_k is number of node points neighboring \mathbf{x}_k , $\mathbf{x}_{k_0} = \mathbf{x}_{k_{N_k}}$ and $\mathbf{x}_{k_{N_k+1}} = \mathbf{x}_{k_1}$. We also define angles $\alpha_{k_m} = \angle \mathbf{x}_k \mathbf{x}_{k_{m-1}} \mathbf{x}_{k_m}$ and $\beta_{k_m} = \angle \mathbf{x}_k \mathbf{x}_{k_{m+1}} \mathbf{x}_{k_m}$ for $m = 1, \dots, N_k$. For example, when $m = 5$, $\alpha_{k_5} = \angle \mathbf{x}_k \mathbf{x}_{k_4} \mathbf{x}_{k_5}$ and $\beta_{k_5} = \angle \mathbf{x}_k \mathbf{x}_{k_6} \mathbf{x}_{k_5}$ in triangles T_5 and T_6 , respectively, as shown in Figure 1.

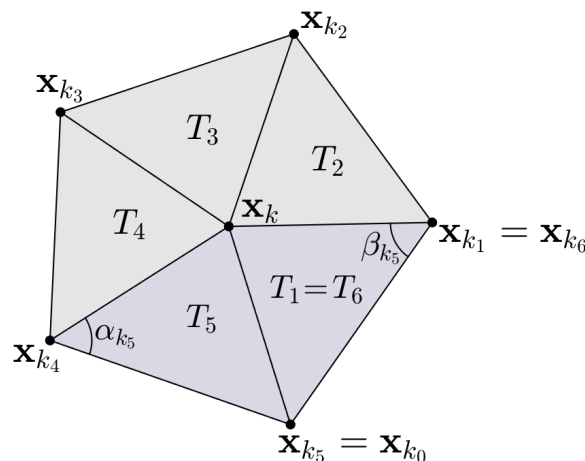


Figure 1. Schematic illustration of the proposed method.

Now, we propose the novel discrete Laplace operator. Let N be the total number of node points in the triangular mesh and N_k be the number of one-ring neighboring node points of each node \mathbf{x}_k , for $k = 1, 2, \dots, N$ in the triangular mesh. The neighboring node points of each node point \mathbf{x}_k are denoted as \mathbf{x}_{k_m} , $m = 1, 2, \dots, N_k$. We define the net force \mathbf{F}_k for each node point \mathbf{x}_k , $k = 1, 2, \dots, N$, which plays a crucial role in achieving symmetry within the proposed discrete Laplace operator.

$$\mathbf{F}_k = \sum_{m=1}^{N_k} (\mathbf{x}_{k_m} - \mathbf{x}_k).$$

The net force at each points in a symmetric object or system is zero because the forces acting on the object are symmetrically distributed. Additionally, the net force of the general five and nine stencil Laplace operators is $\mathbf{0}$. Therefore, we propose the following sub-algorithm using virtual nodes such that the net force \mathbf{F}_k of each node \mathbf{x}_k , $k = 1, 2, \dots, N_k$ becomes $\mathbf{0}$. We solve the AC equation on the irregular domain by using a sub-algorithm using virtual nodes to make each node point symmetric for the asymmetric triangular mesh generated on the irregular domain in 2D space.

The sub-algorithm of the proposed method is illustrated in Figure 2. As presented in Figure 2, for each node \mathbf{x}_k , one-ring neighboring node points are denoted by $\mathbf{x}_{k_1}, \mathbf{x}_{k_2}, \dots, \mathbf{x}_{k_6}$. First, find all the one-ring neighboring node points for a given point \mathbf{x}_k . Figure 2(a) presents a given point \mathbf{x}_k and one-ring neighboring node points. Second, find the net force \mathbf{F}_k and the triangular domain containing \mathbf{F}_k . Here, we use the following method to find the triangle. In the case of Figure 2(b), let the vectors from \mathbf{x}_k of

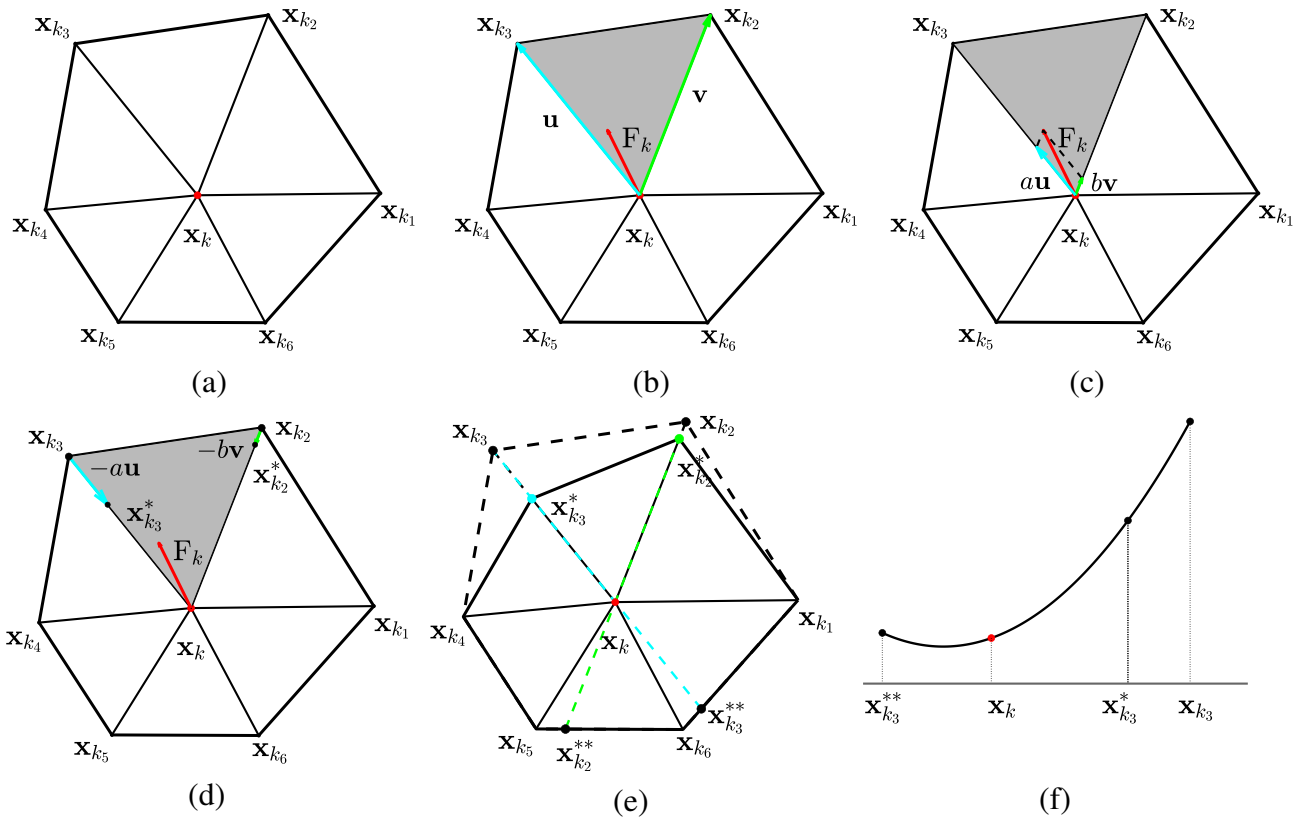


Figure 2. Adjusting the net force F to be 0 : (a)–(e) represent each step.

the found triangle to the other two node points \mathbf{x}_{k_3} and \mathbf{x}_{k_2} be \mathbf{u} and \mathbf{v} , respectively. Therefore, net force \mathbf{F}_k is the linear combination of \mathbf{u} and \mathbf{v} with constants a and b as follows:

$$a\mathbf{u} + b\mathbf{v} = \mathbf{F}, \quad a, b \in \mathbb{R},$$

where \mathbf{u} and \mathbf{v} are 2D vectors. Let $\mathbf{u} = (u_1, u_2)^T$ and $\mathbf{v} = (v_1, v_2)^T$. For the third step, we consider the following the matrix form to find values of a and b .

$$\begin{pmatrix} u_1 & v_1 \\ u_2 & v_2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \mathbf{F} \Rightarrow \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} u_1 & v_1 \\ u_2 & v_2 \end{pmatrix}^{-1} \mathbf{F}.$$

Then, subtract $a\mathbf{u}$ and $b\mathbf{v}$ from \mathbf{u} and \mathbf{v} to reduce the net force \mathbf{F}_k to $\mathbf{0}$ at \mathbf{x}_k . Let $x_{k_3}^*$ and $x_{k_2}^*$ be virtual node points corresponding to $\mathbf{u} - a\mathbf{u}$ and $\mathbf{v} - b\mathbf{v}$, respectively. In the context of the triangular mesh, it is important to note that the physical positions of the node points remain unchanged. Instead, a theoretical construct of imaginary node points is utilized to ensure that the net force acting on the mesh is balanced to zero. By introducing these virtual node points, it is possible to accurately calculate the net force and achieve a state of equilibrium, without the need to manipulate the physical node points. Fourth, we use interpolation for the points where the extension of the modified and original meshes meet on the opposite side of \mathbf{u} and \mathbf{v} , respectively. In Figure 2(e), the value of $\mathbf{x}_{k_2}^{**}$ is obtained by the linear interpolation of the values of \mathbf{x}_{k_5} and \mathbf{x}_{k_6} as follows:

$$u(\mathbf{x}_{k_2}^{**}) = \frac{u(\mathbf{x}_{k_6}) - u(\mathbf{x}_{k_5})}{\sqrt{(x_{k_6} - x_{k_5})^2 + (y_{k_6} - y_{k_5})^2}} \sqrt{(x_{k_2}^{**} - x_{k_5})^2 + (y_{k_2}^{**} - y_{k_5})^2} + u(\mathbf{x}_{k_5}).$$

Here, the virtual node point $\mathbf{x}_{k_2}^{**}$ is the intersection point of the line passing through \mathbf{x}_k and \mathbf{x}_{k_2} , and the line passing through \mathbf{x}_{k_5} and \mathbf{x}_{k_6} . In addition, $\mathbf{x}_{k_3}^{**}$ is obtained analogously. The net force at a point is zero in a symmetric object or system, as the forces acting on it are distributed symmetrically, ensuring that they cancel out each other. Therefore, when virtual node points are used, each node point of the triangular mesh becomes symmetrical with respect to one-ring neighboring points. Finally, the values of modified mesh points are obtained by quadratic interpolation. In Figure 2(f), the value of virtual node $\mathbf{x}_{k_3}^*$ is obtained by the quadratic interpolation using the values of $\mathbf{x}_{k_3}^{**}$, \mathbf{x}_3 , and \mathbf{x}_{k_3} . The value of virtual node $\mathbf{x}_{k_2}^*$ is similarly obtained. Next, we define the discrete Laplace operator with the values of the neighboring node points set such that the net force \mathbf{F}_k at a given point is zero. Let \mathbf{x}_{k_m} for $m = 1, \dots, N_k$ be the one-ring neighboring node points at a point \mathbf{x}_k with net force $\mathbf{F}_k = \mathbf{0}$, then we propose the discrete Laplace operator as follows:

$$\tilde{\Delta}u_k = \frac{\sum_{m=1}^{N_k} (u(\mathbf{x}_{k_m}) - u(\mathbf{x}_k))}{S_k},$$

where S_k is a constant given for each \mathbf{x}_k . The general Laplace operator satisfies $\Delta f(x, y) = 1$ if $f(x, y) = 0.25(x^2 + y^2)$. Because the proposed Laplace operator should satisfy the characteristics of the general Laplace operator, we define S_k as

$$S_k = \sum_{m=1}^{N_k} f(\mathbf{d}_{k_m}).$$

Here, $f(x, y) = 0.25(x^2 + y^2)$ and $\mathbf{d}_{k_m} = \mathbf{x}_{k_m} - \mathbf{x}_k$. The reason for using this function is that it is a symmetric function with respect to the x - and y -axes, and unlike a linear function, a non-zero solution exists, and unlike a cubic function, the solution is uniquely determined. Therefore, the proposed discrete Laplace operator satisfies the characteristic of general Laplace operator, $u(x, y) = 0.25(x^2 + y^2)$. Then we discretize Eq (1.1) in temporal and spatial as follows

$$\begin{aligned} \frac{u_k^{n+1} - u_k^n}{\Delta t} &= -\frac{F'(u_k^n)}{\epsilon^2} + \tilde{\Delta} u_k^n \\ &= -\frac{F'(u_k^n)}{\epsilon^2} + \frac{\sum_{m=1}^{N_k} (u_{k_m}^n - u_k^n)}{S_k}, \end{aligned} \quad (2.1)$$

where $S_k = \sum_{m=1}^{N_k} f(\mathbf{d}_{k_m})$. Therefore, we obtain the following equation by rearranging Eq (2.1) for u_k^{n+1} .

$$u_k^{n+1} = u_k^n + \Delta t \left(-\frac{F'(u_k^n)}{\epsilon^2} + \frac{\sum_{m=1}^{N_k} (u_{k_m}^n - u_k^n)}{S_k} \right). \quad (2.2)$$

3. Numerical experiments

3.1. Discrete Laplace operator

We consider the general five stencil, nine stencil and exact Laplace operators to compare the proposed Laplace operator with the conventional Laplace operators.

$$\Delta_1 u_{ij} = \frac{u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{ij}}{h^2}, \quad (3.1)$$

$$\Delta_2 u_{ij} = \frac{u_{i-1,j+1} + u_{i+1,j+1} + u_{i+1,j-1} + u_{i-1,j-1} - 4u_{ij}}{2h^2}, \quad (3.2)$$

$$\begin{aligned} \Delta_3 u_{ij} &= \frac{2}{3} \Delta_1 u_{ij} + \frac{1}{3} \Delta_2 u_{ij} = \frac{4(u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1})}{6h^2} \\ &+ \frac{u_{i-1,j+1} + u_{i+1,j+1} + u_{i+1,j-1} + u_{i-1,j-1} - 20u_{ij}}{6h^2}, \end{aligned} \quad (3.3)$$

which are the well-known standard discrete Laplace operators. Figures 3(a),(b),(c) represent the numerical stencils for Eqs (3.1), (3.2) and (3.3), respectively.

3.1.1. Convergence test on rectangular mesh

In this section, each discrete Laplace operator is compared with the C^2 function to verify the performance of the proposed discrete Laplace operator. The proposed discrete Laplace operator $\tilde{\Delta}$ at each node is made by neighboring nodes as given in Figure 3(c). This is one of the benefits of our proposed discrete Laplace operator on a triangular mesh. Therefore, we obtain the discrete Laplace equation as follows using the proposed method.

$$\begin{aligned} \tilde{\Delta} u_{ij} &= (u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} + u_{i+1,j+1} + u_{i+1,j-1} \\ &+ u_{i-1,j+1} + u_{i-1,j-1} - 8u_{ij})/3h^2. \end{aligned}$$

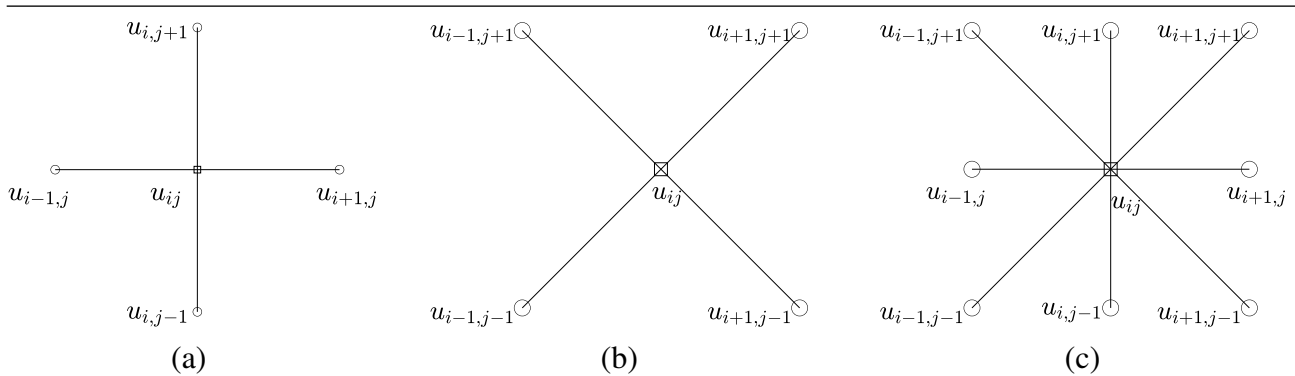


Figure 3. Stencils for (a) $\Delta_1 u_{ij}$ using five points, (b) $\Delta_2 u_{ij}$ using rotated five points, and (c) $\Delta_3 u_{ij}$ using nine points.

We consider the following function on the domain $\Omega = (0, 1) \times (0, 1)$:

$$u(x, y) = \sin(\pi x) \sin(\pi y), \quad (x, y) \in \Omega \quad (3.4)$$

with zero Dirichlet boundary condition. Then, the exact solution of Δu is

$$\Delta u(x, y) = -2\pi^2 \sin(\pi x) \sin(\pi y).$$

We use different parameter values $N_x = 9, 17, 33, 65$ to consider the convergence rates for different spatial mesh $h_x = h_y = 1/(N_x - 1)$ and discrete domain $x_i = (i - 1)h_x$ for $i = 1, 2, \dots, N_x$ and $y_j = (j - 1)h_y$ for $j = 1, 2, \dots, N_y$. Next, to calculate the convergence rate for Δu_{ij} , we define the discrete l_2 -error and discrete maximum-error in 2D space, respectively:

$$\begin{aligned} \|\mathbf{e}_{N_x N_y}\|_{l_2} &= \sqrt{\frac{1}{N_x} \frac{1}{N_y} \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} (e_{ij})^2}, \\ \|\mathbf{e}_{N_x N_y}\|_{max} &= \max_{1 \leq i \leq N_x, 1 \leq j \leq N_y} |e_{ij}|, \end{aligned}$$

where $e_{ij} = \hat{\Delta} u_{ij} - \Delta u_{ij}$ for $i = 1, 2, \dots, N_x$ and $j = 1, 2, \dots, N_y$. Figures 4(a),4(b) present the l_2 -error and the maximum-error for different Laplace operators, respectively. We make a list of the convergence rates for the maximum-error and l_2 -error of the Laplace operator in Tables 1 and 2, respectively. We observed that the proposed discrete Laplace operator achieved second-order accuracy in spatial directions on a rectangular mesh. The proposed discrete Laplace operator can consider triangular meshes on irregular domains rather than rectangular meshes, while other discrete Laplace operators can only be used on rectangular meshes.

3.1.2. Laplace operator for functions on triangular mesh

The proposed method is for 2D triangular mesh, therefore we need to define some conditions. Let N be the number of nodes in the triangular mesh and $\mathbf{x}_k = (x_k, y_k)$ be k -th node in the triangular mesh for $k = 1, \dots, N$. Set I be the set of indices of the interior nodes and N_k is the number of node points neighboring \mathbf{x}_k . The average of spatial mesh is

$$h_{ave} = \frac{\sum_{k \in I} \sum_{m=1}^{N_k} |\mathbf{x}_{k_m} - \mathbf{x}_k|}{\sum_{k \in I} N_k}, \quad (3.5)$$

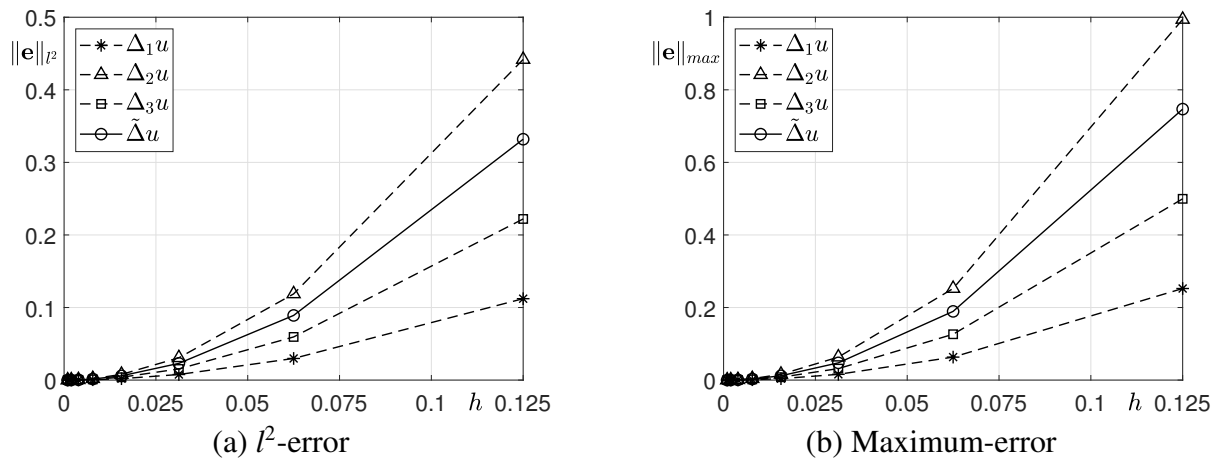


Figure 4. The value of error for each Laplace operator.

Table 1. Maximum-error of different Laplace operators with respect to different spatial steps.

spatial step	$h(= 1/8)$	rate	$h/2$	rate	$h/4$	rate	$h/8$
$\Delta_1 u$	2.5237×10^{-1}	1.99	6.3336×10^{-2}	2.00	1.5849×10^{-2}	2.00	3.9633×10^{-3}
$\Delta_2 u$	9.9404×10^{-1}	1.98	2.5237×10^{-1}	1.99	6.3336×10^{-2}	2.00	1.5849×10^{-2}
$\Delta_3 u$	4.9959×10^{-1}	1.98	1.2635×10^{-1}	2.00	3.1678×10^{-2}	2.00	7.9253×10^{-3}
$\tilde{\Delta} u$	7.4682×10^{-1}	1.98	1.8936×10^{-1}	1.99	4.7507×10^{-2}	2.00	1.1887×10^{-2}

Table 2. l_2 -error of different Laplace operators with respect to different spatial steps.

spatial step	$h(= 1/8)$	rate	$h/2$	rate	$h/4$	rate	$h/8$
$\Delta_1 u$	1.1216×10^{-1}	1.91	2.9805×10^{-2}	1.96	7.6845×10^{-3}	1.98	1.9511×10^{-3}
$\Delta_2 u$	4.4180×10^{-1}	1.90	1.1876×10^{-1}	1.95	3.0708×10^{-2}	1.98	7.8027×10^{-3}
$\Delta_3 u$	2.2204×10^{-1}	1.90	5.9457×10^{-2}	1.95	1.5359×10^{-2}	1.98	3.9017×10^{-3}
$\tilde{\Delta} u$	3.3192×10^{-1}	1.90	8.9110×10^{-2}	1.95	2.3034×10^{-2}	1.98	5.8522×10^{-3}

where \mathbf{x}_{k_m} is neighboring node points of \mathbf{x}_k for $m = 1, \dots, N_k$.

Let the initial function $u(x, y) = ax + by + c$ for $a, b, c \in \mathbb{R}$ be a plane, then $\Delta u = 0$. Therefore, we consider the initial condition on the domain $\Omega = (0, 1) \times (0, 1)$:

$$u(x, y) = x + 2y + 3, \quad (x, y) \in \Omega. \quad (3.6)$$

Here, we apply Dirichlet boundary conditions. To compute the discrete Laplace operator of u , we use a triangular mesh with $h_{\text{ave}} = 0.1013$. Figures 5(a),(b),(c) present the triangular mesh, $u(x, y) = x + 2y + 3$, and proposed discrete Laplace operator for u , respectively.

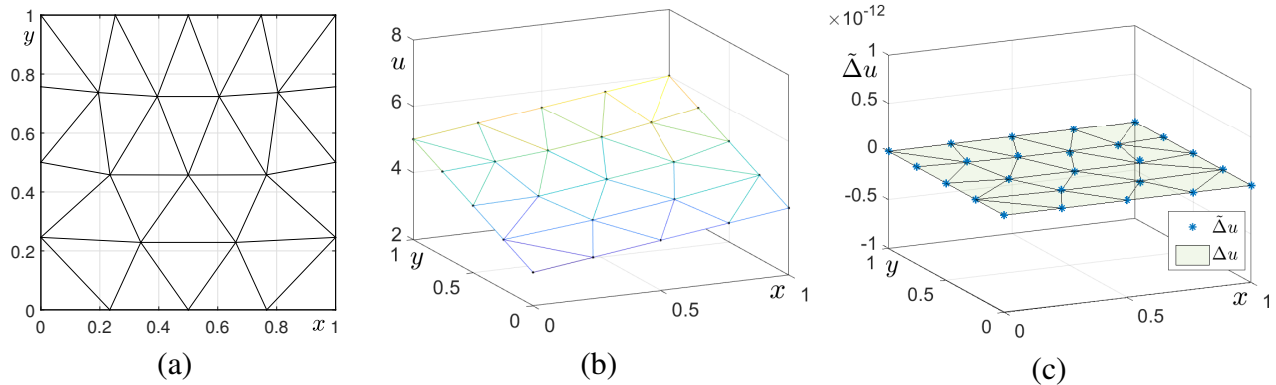


Figure 5. Discrete Laplace operator for plane equation: (a) triangular mesh, (b) $u(x, y) = x + 2y + 3$, and (c) $\tilde{\Delta}u(x, y)$.

Next, we consider the initial function on the domain $\Omega = (0, 1) \times (0, 1)$:

$$u(x, y) = \sin(\pi x) \sin(\pi y). \quad (3.7)$$

Then we have

$$\Delta u(x, y) = -2\pi^2 \sin(\pi x) \sin(\pi y). \quad (3.8)$$

We apply the zero Dirichlet boundary condition. To compute the discrete Laplace operator of u , we use the parameter $h_{\text{ave}} = 0.1013$. Figures 6(a),(b),(c) show the triangular mesh, $u(x, y) = \sin(\pi x) \sin(\pi y)$, and the proposed discrete Laplace operator for u , i.e., $\tilde{\Delta}u(x, y)$, respectively.

3.2. Diffusion equations on triangular mesh

3.2.1. Convergence test on square domain

In this section, we compare the numerical and analytical solutions to verify the proposed method. We consider a diffusion equation (1.2) with the following initial condition on the domain $\Omega = (0, 1) \times (0, 1)$:

$$u(x, y, 0) = \sin(\pi x) \sin(\pi y) \quad (3.9)$$

with zero Dirichlet boundary condition. Then, the exact solution is

$$u_{\text{exact}}(x, y, t) = \sin(\pi x) \sin(\pi y) e^{-2\pi^2 t}.$$

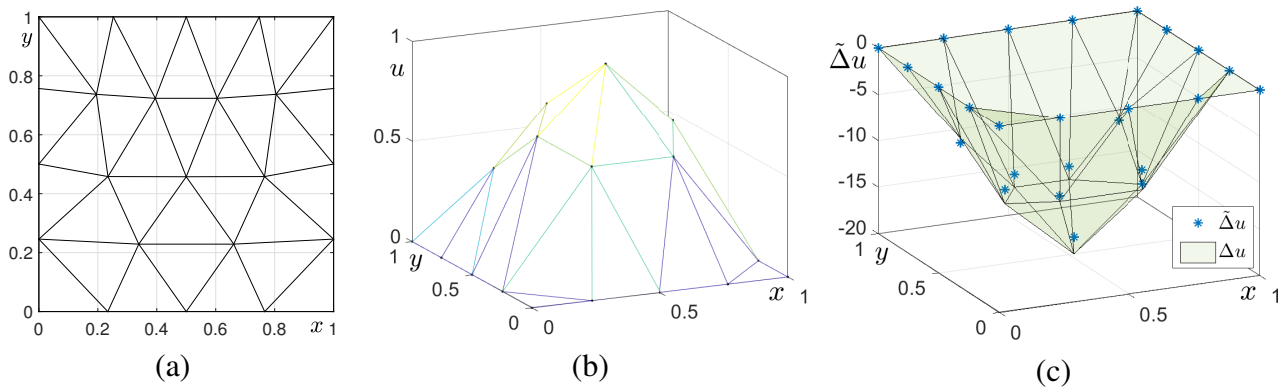


Figure 6. (a) Triangular mesh, (b) $u(x, y) = \sin(\pi x) \sin(\pi y)$, and (c) $\tilde{\Delta}u(x, y)$.

Let u_k^n be an approximation of $u(x_k, y_k, n\Delta t)$, where Δt is a time step. Let N be the number of nodes of the triangular mesh, $\mathbf{x}_k = (x_k, y_k)$ be k -th node in triangular mesh for $k = 1, \dots, N$. For this test, we use different triangular meshes with $h_{\text{ave}} = 0.1254, 0.1013, 0.0762$. Using the explicit Euler method and the proposed discrete Laplace operator, Eq (1.2) is as follows:

$$\frac{u_k^{n+1} - u_k^n}{\Delta t} = \tilde{\Delta}u_k^n. \quad (3.10)$$

Let us define the l_2 -error and maximum-error as follows:

$$\|\mathbf{e}_N^{N_t}\|_{l_2} = \sqrt{\frac{1}{N} \sum_{k=1}^N (e_k^{N_t})^2},$$

$$\|\mathbf{e}_N^{N_t}\|_{\max} = \max_{1 \leq k \leq N} |e_k^{N_t}|,$$

where $e_k^{N_t} = u_k^{N_t} - u_{\text{exact}}(x_k, y_k, T)$, $k = 1, 2, \dots, N$. Figure 7 presents the triangular mesh and numerical solution using the proposed method for each h_{ave} . We list the maximum-error and l_2 -error of the diffusion equation with $T = 0.02$, $N_t = 2048$, $\Delta t = T/N_t$ in Table 3.

Table 3. Maximum-error and l_2 -error for proposed discrete Laplace operator with respect to different spatial steps.

Spatial step	$h_{\text{ave}} = 0.1254$	rate	$h_{\text{ave}} = 0.1013$	rate	$h_{\text{ave}} = 0.0762$
Maximum-error	1.8338×10^{-2}	1.19	1.4246×10^{-2}	1.92	8.2534×10^{-3}
l_2 -error	4.2098×10^{-3}	1.64	2.9665×10^{-3}	1.77	1.7904×10^{-3}

Next, we perform numerical experiments for a larger time evolution with a final time $T = 0.1$ using the same triangular meshes as in the previous test. Figure 8 shows the numerical solutions for different spatial steps at the final time $T = 0.1$. Table 4 lists the maximum-error and l_2 -error for the proposed discrete Laplace operator. We observed that the l_2 -error and maximum-error of the proposed discrete Laplace operator decreased as the spatial step decreased for a larger final time $T = 0.1$.

We perform a convergence test on the discrete LBO to compare the discrete LBO and the proposed discrete Laplace operator. The initial conditions given are the same as in the previous convergence test

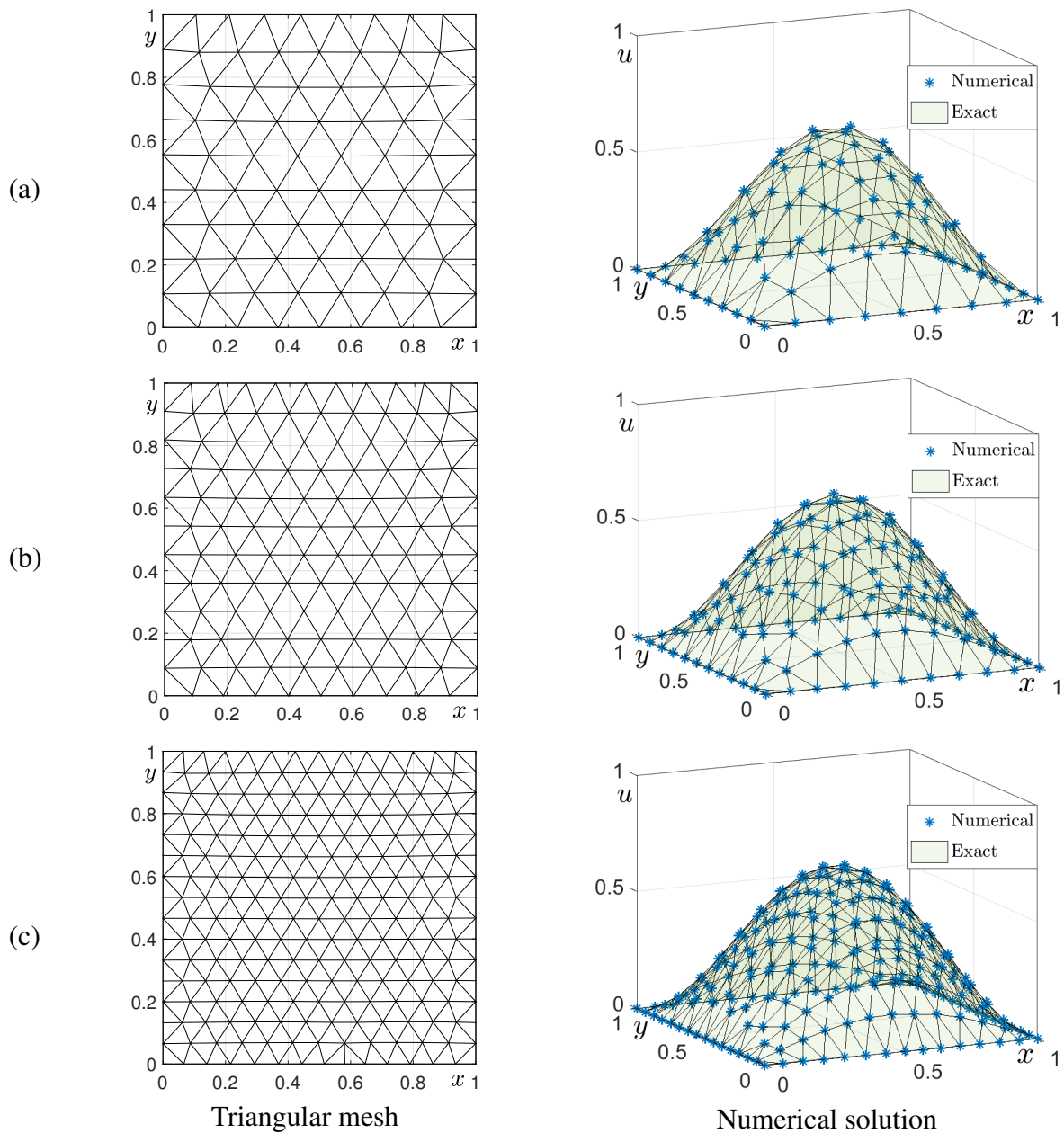


Figure 7. Triangular mesh and numerical solution using the proposed method for each different spatial mesh with h_{ave} . (a) $h_{ave} = 0.1254$, (b) $h_{ave} = 0.1013$, (c) $h_{ave} = 0.0762$.

Table 4. Maximum-error and l_2 -error for proposed discrete Laplace operator with respect to different spatial steps.

Spatial step	$h_{ave} = 0.1254$	rate	$h_{ave} = 0.1013$	rate	$h_{ave} = 0.0762$
Maximum-error	4.6855×10^{-3}	2.07	3.0121×10^{-3}	1.98	1.7142×10^{-3}
l_2 -error	2.0184×10^{-3}	2.89	1.0894×10^{-3}	1.09	7.9989×10^{-4}

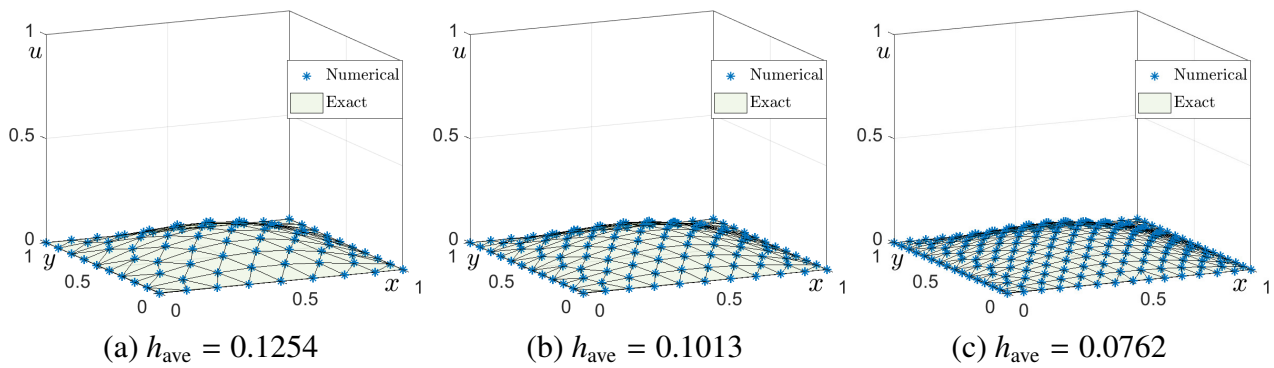


Figure 8. Triangular mesh and numerical solution for a larger final time $T = 0.1$.

with the final time $T = 0.1$. Figure 9 shows the numerical solutions obtained using the discrete LBO for different spatial steps. Table 5 lists the maximum-error and l_2 -error for the discrete LBO.

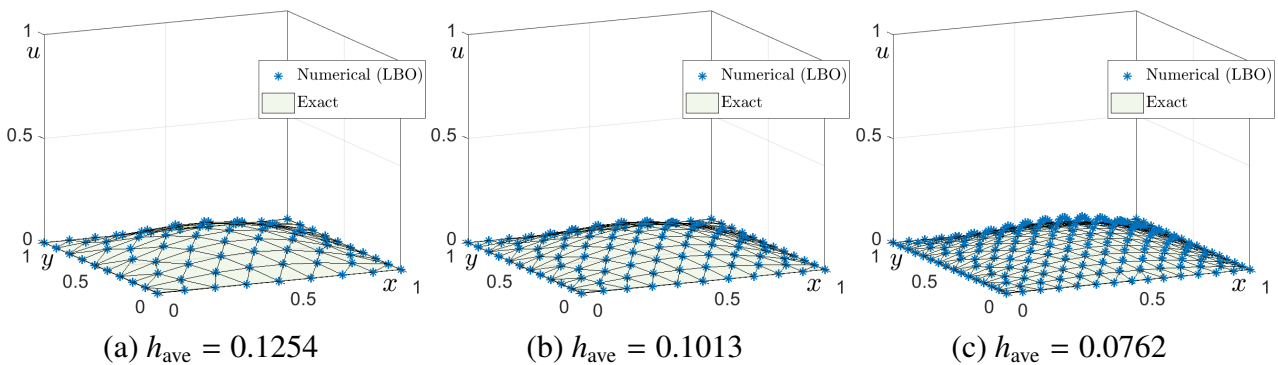


Figure 9. Triangular mesh and numerical solution using the discrete LBO.

Table 5. Maximum-error and l_2 -error for the discrete LBO with respect to different spatial steps.

Spatial step	$h_{\text{ave}} = 0.1254$	rate	$h_{\text{ave}} = 0.1013$	rate	$h_{\text{ave}} = 0.0762$
Maximum-error	5.3441×10^{-3}	2.11	3.4101×10^{-3}	-7.49	2.8734×10^{-2}
l_2 -error	2.3112×10^{-3}	2.01	1.5061×10^{-3}	-7.16	1.1569×10^{-2}

We observed that the error of the discrete LBO is larger than that of the proposed discrete Laplace operator and does not converge as the spatial step becomes smaller.

3.2.2. Diffusion equation on circular domain

In this section, we consider a diffusion equation on a circular domain using the proposed discrete Laplace operator. We use the circular domain of radius 1 and with center at the origin, i.e., $\Omega = \{(x, y) | x^2 + y^2 < 1\}$. We use the zero Dirichlet boundary condition and $u(x, y, 0) = 1 - x^2 - y^2$ as the initial condition.

$$\frac{\partial u(x, y, t)}{\partial t} = \Delta u(x, y, t), \quad (x, y) \in \Omega, \quad t > 0, \quad (3.11)$$

$$\begin{aligned}u(x, y, t) &= 0, \quad (x, y) \in \partial\Omega, \\u(x, y, 0) &= 1 - x^2 - y^2, \quad (x, y) \in \Omega.\end{aligned}$$

Let us consider a benchmark problem using a manufactured solution [28], i.e., we assume that the solution of the benchmark problem is

$$u(x, y, t) = (1 - x^2 - y^2)e^{-t}. \quad (3.12)$$

Then, we consider the following benchmark problem:

$$\begin{aligned}\frac{\partial u(x, y, t)}{\partial t} &= \Delta u(x, y, t) + g(x, y, t), \quad (x, y) \in \Omega, \quad t > 0, \\u(x, y, t) &= 0, \quad (x, y) \in \partial\Omega, \\u(x, y, 0) &= 1 - x^2 - y^2, \quad (x, y) \in \Omega,\end{aligned} \quad (3.13)$$

where $g(x, y, t) = \partial u(x, y, t)/\partial t - \Delta u(x, y, t) = (3 + x^2 + y^2)e^{-t}$ is the source term. To construct the numerical solution, we use explicit Euler method as follows:

$$u_k^{n+1} = u_k^n + \Delta t \left(\tilde{\Delta} u_k + g_k^{n+\frac{1}{2}} \right), \quad (3.14)$$

where $g_k^{n+\frac{1}{2}} = (3 + x_k^2 + y_k^2)e^{-(n+\frac{1}{2})\Delta t}$.

Let us consider three different spatial meshes with $h_{\text{ave}} = 0.5315, 0.2133$ and 0.1020 . The total time is $T = 0.1$ and time step $\Delta t = 0.001h_{\text{ave}}^2$ with total iteration $N_t = T/\Delta t$. Using the proposed discrete Laplace operator, we present the numerical solution at final time T and compare it with the exact solution, Eq (3.12). Figure 10 presents the triangular mesh, exact solution and numerical results, and the absolute error between the results.

3.3. Two-dimensional Allen-Cahn equation on triangular mesh

In this section, we consider the AC Eq (1.1) on plane with triangular mesh. Let Ω be the domain in 2D space and N be the number of nodes in domain. For $k = 1, \dots, N$, let $\mathbf{x}_k = (x_k, y_k)$ be the k -th node in domain. On the computational triangular mesh, let u_k^n be an approximation of $u(x_k, y_k, n\Delta t)$, where Δt is a time step.

3.3.1. Motion by mean curvature on square domain

In Figure 12, we verify the mean curvature flow of the AC equation using following initial condition on $\Omega = (-1, 1) \times (-1, 1)$:

$$u(x, y, 0) = \tanh\left(\frac{R_0 - \sqrt{x^2 + y^2}}{\sqrt{2}\epsilon}\right), \quad (x, y) \in \Omega, \quad (3.15)$$

where $R_0 = 0.4$ is the initial radius with Dirichlet boundary condition $u(x, y, t) = -1$ on $\partial\Omega$. The analytic radius is given as $R(t) = \sqrt{R_0^2 - 2t}$ for time t [29]. The numerical radius is computed using the average of distances between zero level points of the numerical solution and the center point $(0, 0)$. As shown in Figure 12, the numerical radii shrink along with the analytic radius.

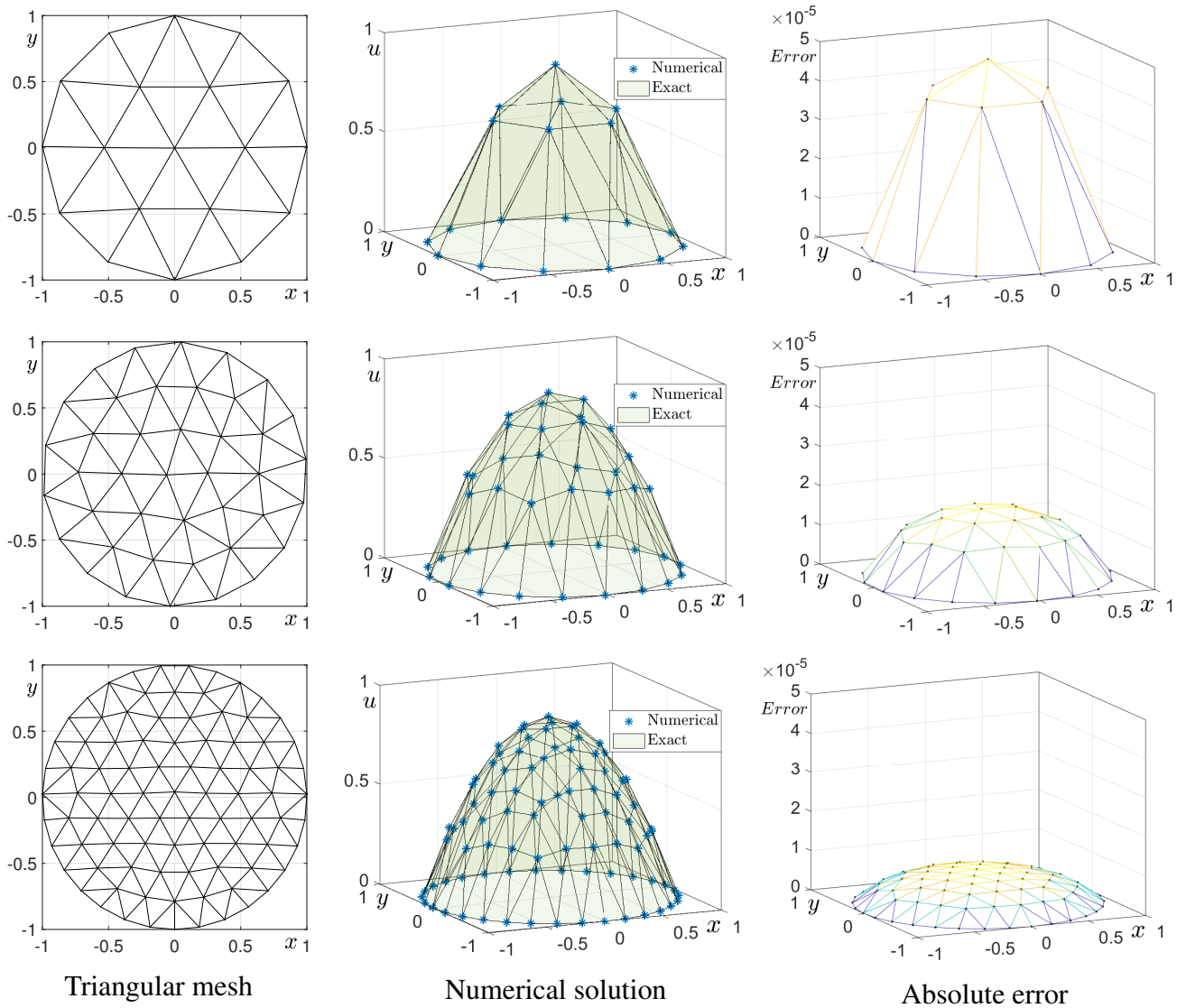


Figure 10. Triangular mesh, numerical and exact solutions, and absolute error, here top to bottom $h_{ave} = 0.5315, 0.3138, 0.2133$, respectively.

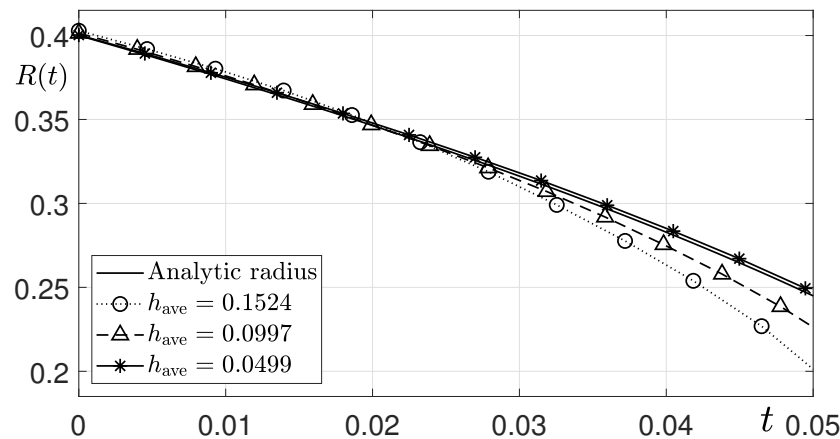


Figure 12. Motion by mean curvature for different spatial step $h_{ave} = 0.1524, 0.0997, 0.0499$.

3.3.2. Dynamics of the AC equation on various domains

In this section, we observe the dynamics of the AC equation on various domains. We consider rectangular domain with circular hole $\Omega_{rectangle} = ((-1.3, 1.3) \times (-1.3, 1.3)) \cap B_2(0.5)^c$ and circular domain with circular hole $\Omega_{circle} = B_1(1.3) \cap B_2(0.5)^c$ to verify the performance of the proposed method according to the shape of the domain, where $B_1(r) = \{(x, y) \mid x^2 + y^2 < r\}$ and $B_2(r) = \{(x, y) \mid x^2 + y^2 \leq r\}$. The initial condition is that all internal node points in the domain are 1 with Dirichlet boundary condition $u(x, y, t) = 1$ in $\partial\Omega_{inner}$ and $u(x, y, t) = -1$ in $\partial\Omega_{outer}$. Here, $\partial\Omega_{inner}$ and $\partial\Omega_{outer}$ are the inner and outer boundaries of the domain, respectively. We use the parameters $h_{ave} \approx 0.2, 0.15, 0.1$, $\epsilon = 0.15$, $\Delta t = 0.001$, and $N_t = T/\Delta t$ to compute Eq (2.2). Here, we use $T = 5.6$, which is a sufficient time to reach an numerical equilibrium state for all parameters. The value of h_{ave} is below each figure. Figures 13 and 14 show the dynamics of the AC equation on $\Omega_{rectangle}$ and Ω_{circle} , respectively, for the given initial conditions.

4. Conclusions

In this study, we proposed a novel, effective, simple, and explicit numerical method for solving the diffusion and AC equations on a triangular mesh. In the proposed algorithm, we defined a novel discrete Laplace operator. The proposed discrete Laplace operator has a property that the adjusted net vector of each point using virtual nodes is zero vector on the triangular mesh. This means that the forces acting at each point of the triangular mesh are symmetrically distributed, thus demonstrating the symmetry of the proposed method. To investigate the efficiency and stability of the proposed numerical method, we presented the numerical experiments for solving the diffusion and AC equations in various domains on triangular mesh. We verified the proposed Laplace operator through numerical experiments on general discrete Laplace operators and discrete LBOs. Additionally, we verified the proposed operator division method through numerical experiments involving motion by mean curvature using the AC equation and solving the AC equation in irregular domains. Results indicated that the proposed method can be served in various applications in 2D domain with triangular mesh. In future work, we will solve the AC equation on complex surfaces in 3D space and conduct theoretical analysis on the convergence of

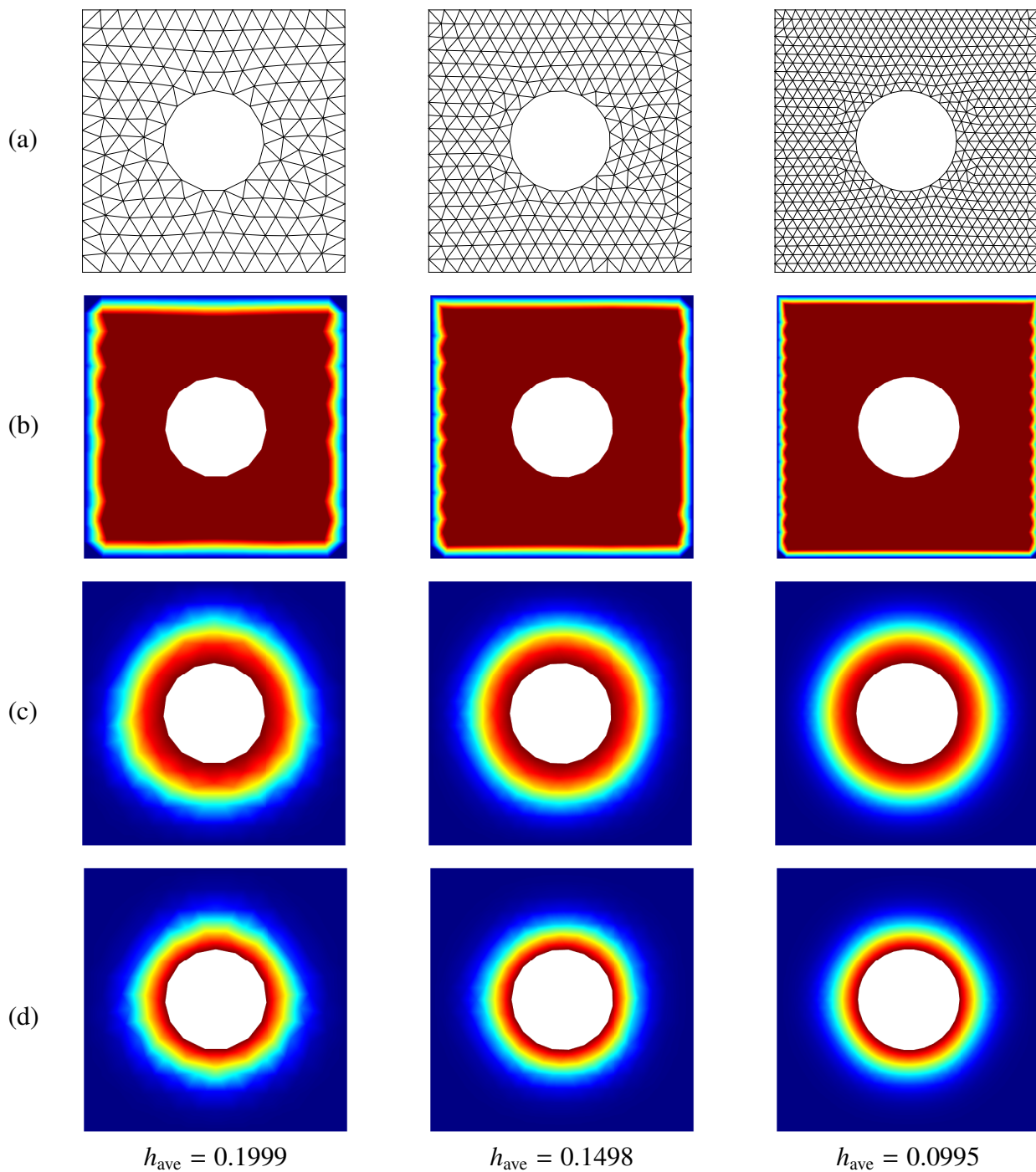


Figure 13. Dynamics of the AC equation on the rectangle domain with circular hole. (a) Triangular mesh, (b) Initial condition, (c) $t = 0.4$, (d) $t = 5.6$.

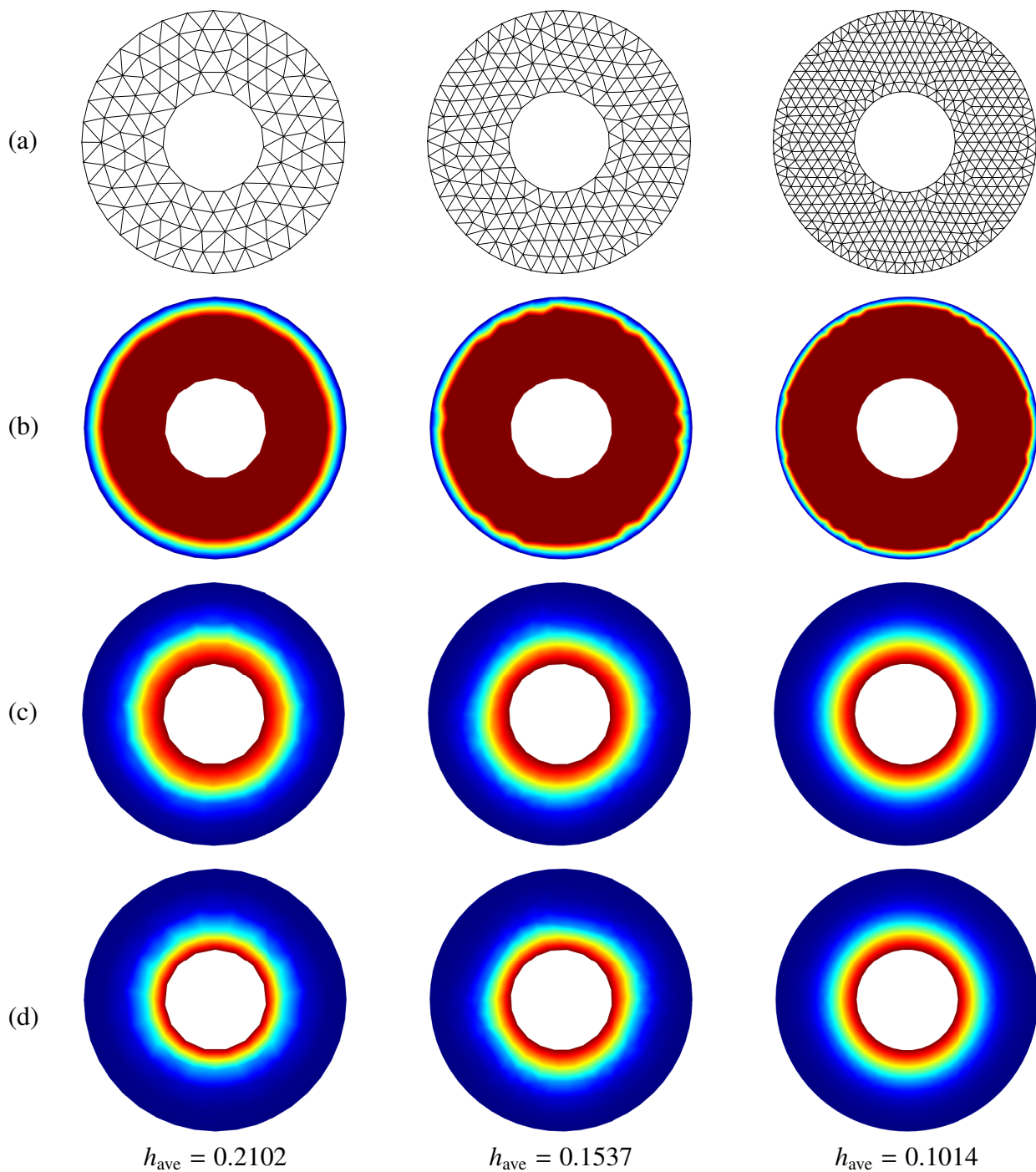


Figure 14. Dynamics of the AC equation on the circular domain with circular hole. (a) Triangular mesh, (b) Initial condition, (c) $t = 0.4$, (d) $t = 5.6$.

the proposed method.

Use of AI tools declaration

The authors have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The corresponding author (J. S. Kim) expresses thanks for the support from the BK21 FOUR program. The authors express their sincere gratitude to the reviewers for providing valuable feedback on this revised version. Their constructive comments have significantly contributed to the improvement of the manuscript.

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. J. W. Choi, H. G. Lee, D. Jeong, J. Kim, An unconditionally gradient stable numerical method for solving the Allen–Cahn equation, *Phys. A*, **388** (2009), 1791–1803. <https://doi.org/10.1016/j.physa.2009.01.026>
2. S. M. Allen, J. W. Cahn, A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening, *Acta Metall.*, **27** (1979), 1085–1095. [https://doi.org/10.1016/0001-6160\(79\)90196-2](https://doi.org/10.1016/0001-6160(79)90196-2)
3. D. Jeong, Y. Li, Y. Choi, C. Lee, J. Yang, J. Kim, A practical adaptive grid method for the Allen–Cahn equation, *Phys. A*, **573** (2021), 125975. <https://doi.org/10.1016/j.physa.2021.125975>
4. X. Xiao, R. He, X. Feng, Unconditionally maximum principle preserving finite element schemes for the surface Allen–Cahn type equations, *Numer. Meth. Part Differ. Equations*, **36** (2020), 418–438. <https://doi.org/10.1002/num.22435>
5. J. Rubinstein, P. Sternberg, Nonlocal reaction–diffusion equations and nucleation, *IMA J. Appl. Math.*, **48** (1992), 249–264. <https://doi.org/10.1093/imamat/48.3.249>
6. 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>
7. B. Xia, Y. Li, Z. Li, Second-order unconditionally stable direct methods for Allen–Cahn and conservative Allen–Cahn equations on Surfaces, *Mathematics*, **8** (2020), 1486. <https://doi.org/10.3390/math8091486>
8. 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** (2021), 797–810. <https://doi.org/10.4208/nmtma.OA-2020-0155>

9. V. Joshi, R. K. Jaiman, An adaptive variational procedure for the conservative and positivity preserving Allen–Cahn phase-field model, *J. Comput. Phys.*, **366** (2018), 478–504. <https://doi.org/10.1016/j.jcp.2018.04.022>
10. S. Kwak, J. Yang, J. Kim, A conservative Allen–Cahn equation with a curvature-dependent Lagrange multiplier, *Appl. Math. Lett.*, **126** (2022), 107838. <https://doi.org/10.1016/j.aml.2021.107838>
11. Q. Hong, Y. Gong, J. Zhao, Q. Wang, Arbitrarily high order structure-preserving algorithms for the Allen–Cahn model with a nonlocal constraint, *Appl. Numer. Math.*, **170** (2021), 321–339. <https://doi.org/10.1016/j.apnum.2021.08.002>
12. Z. Chai, D. Sun, H. Wang, B. Shi, A comparative study of local and nonlocal Allen–Cahn equations with mass conservation, *Int. J. Heat Mass Transf.*, **122** (2018), 631–642. <https://doi.org/10.1016/j.ijheatmasstransfer.2018.02.013>
13. M. Inc, A. Yusuf, A. I. Aliyu, D. Baleanu, Time-fractional Cahn–Allen and time-fractional Klein–Gordon equations: Lie symmetry analysis, explicit solutions and convergence analysis, *Phys. A*, **493** (2018), 94–106. <https://doi.org/10.1016/j.physa.2017.10.010>
14. J. Kim, H. Lee, Unconditionally energy stable second-order numerical scheme for the Allen–Cahn equation with a high-order polynomial free energy, *Adv. Differ. Equations*, **2021** (2021), 1–13. <https://doi.org/10.1186/s13662-021-03571-x>
15. H. Bhatt, J. Joshi, I. Argyros, Fourier spectral high-order time-stepping method for numerical simulation of the multi-dimensional Allen–Cahn equations, *Symmetry*, **13** (2021), 245. <https://doi.org/10.3390/sym13020245>
16. Z. Sun, Y. Gao, High order multiquadric trigonometric quasi-interpolation method for solving time-dependent partial differential equations, *Numer. Algorithms*, (2022), 1–21. <https://doi.org/10.1007/s11075-022-01486-6>
17. Z. Sun, L. Ling, A kernel-based meshless conservative Galerkin method for solving Hamiltonian wave equations, *SIAM J. Sci. Comput.*, **44** (2022), A2789–A2807. <https://doi.org/10.1137/21M1436981>
18. M. Wardetzky, S. Mathur, F. Kälberer, E. Grinspun, Discrete Laplace operators: No free lunch, *Eurographics Symp. Geom. Process.*, (2007), 33–37.
19. 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>
20. T. Caissard, D. Coeurjolly, J. O. Lachaud, T. Roussillon, Laplace-beltrami operator on digital surfaces, *J. Math. Imaging Vis.*, **61** (2019), 359–379. <https://doi.org/10.1007/s10851-018-0839-4>
21. S. P. Thampi, S. Ansumali, R. Adhikari, S. Succi, Isotropic discrete Laplacian operators from lattice hydrodynamics, *J. Comput. Phys.*, **234** (2013), 1–7. <https://doi.org/10.1016/j.jcp.2012.07.037>
22. B. J. McCartin, Eigenstructure of the discrete Laplacian on the equilateral triangle: the Dirichlet & Neumann problems, *Appl. Math. Sci.*, **4** (2010), 2633–2646.
23. V. Ganzha, R. Liska, M. Shashkov, C. Zenger, Support operator method for Laplace equation on unstructured triangular grid, *Selcuk J. Appl. Math.*, **3** (2002), 21–48.

24. S. Yoon, J. Park, J. Wang, C. Lee, J. Kim, Numerical simulation of dendritic pattern formation in an isotropic crystal growth model on curved surfaces, *Symmetry*, **12** (2020), 1155. <https://doi.org/10.3390/sym12071155>
25. L. Paquet, R. Korikache, The complete discretization of the dual mixed method for the heat diffusion equation in a polygonal domain, *Math. Comput. Simul.*, **186** (2021), 145–160. <https://doi.org/10.1016/j.matcom.2020.09.023>
26. J. W. Cahn, J. E. Hilliard, Free energy of a nonuniform system. I. Interfacial free energy, *J. Chem. Phys.*, **28** (1958), 258–267. <https://doi.org/10.1063/1.1744102>
27. W. Tian, Y. Chen, Z. Meng, H. Jia, An adaptive finite element method based on superconvergent cluster recovery for the Cahn–Hilliard equation, *Electron. Res. Arch.*, **31** (2023), 1323–1343. <https://doi.org/10.3934/era.2023068>
28. Y. Hwang, C. Lee, S. Kwak, Y. Choi, S. Ham, S. Kang, et al., Benchmark problems for the numerical schemes of the phase-field equations, *Discrete Dyn. Nat. Soc.*, **2022** (2022), 1–10. <https://doi.org/10.1155/2022/2751592>
29. D. Jeong, J. Kim, An explicit hybrid finite difference scheme for the Allen–Cahn equation, *J. Comput. Appl. Math.*, **340** (2018), 247–255. <https://doi.org/10.1016/j.cam.2018.02.026>



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)