Appl. Comput. Math., V.22, N.2, 2023, pp.259-274
DOI: 10.30546/1683-6154.22.2.2023.259

ISOTROPIC FINITE DIFFERENCE DISCRETIZATION OF LAPLACIAN OPERATOR

HYUN GEUN LEE[1], SEOKJUN HAM[2], JUNSEOK KIM[2,*]

Abstract. In this paper, we review and investigate isotropic finite difference discretizations of the two-dimensional (2D) and three-dimensional (3D) Laplacian operators. In particular, we propose benchmark functions to quantitatively evaluate the isotropy of the discrete Laplacian operators in 2D and 3D spaces. The benchmark functions have analytic 2D and 3D Laplacian solutions so that we can exactly compute the errors between the numerical and analytic solutions.

Keywords: Isotropic Discretization, Finite Difference Method, Discrete Laplacian Operator, Isotropic Stencil.

AMS Subject Classification: 80M20.

1. Introduction

The Laplacian operator is extensively used in many mathematical modeling equations of various important scientific problems such as biological and physical pattern formations. The Laplacian operators in two-dimensional (2D) and three-dimensional (3D) spaces are respectively given by

$$\begin{aligned}\Delta u(x,y) &= \frac{\partial^2 u(x,y)}{\partial x^2}+\frac{\partial^2 u(x,y)}{\partial y^2},\\ \Delta u(x,y,z) &= \frac{\partial^2 u(x,y,z)}{\partial x^2}+\frac{\partial^2 u(x,y,z)}{\partial y^2}+\frac{\partial^2 u(x,y,z)}{\partial z^2}.\end{aligned}$$

The Laplacian operator has been used in many governing equations. For example, a 2D Poisson equation [7]:

$$\Delta u(x,y) = f(x,y).$$

The diffusion equation [4,6,22,24]:

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

where $u(x,y,t)$ is the density of the diffusing material at location (x,y) and time t. The 3D acoustic wave equation [23]:

$$\frac{\partial^2 p(x,y,z,t)}{\partial t^2} = v^2 \Delta p(x,y,z,t),$$

where $p(x,y,z,t)$ is a scalar wave-field and v is the velocity. The 3D Allen–Cahn (AC) equation [16]:

$$\frac{\partial \phi(x,y,z,t)}{\partial t} = -\frac{\phi^3(x,y,z,t)-\phi(x,y,z,t)}{\epsilon^2}+\Delta\phi(x,y,z,t),$$

[1]Department of Mathematics, Kwangwoon University, Seoul 01897, Republic of Korea
[2]Department of Mathematics, Korea University, Seoul 02841, Republic of Korea
*Corresponding author: e-mail: cfdkim@korea.ac.kr
Manuscript received 25 January 2023.

where $\phi(x, y, z, t)$ is an order parameter and ϵ is an interfacial parameter. Isotropic finite difference methods for phase-field simulations of polycrystalline alloy solidification was applied for phase-field simulation of polycrystalline alloy solidification in [10]. The Cahn–Hilliard (CH) equation [15]:

$$\frac{\partial \phi(x, y, z, t)}{\partial t} = \Delta \left[\phi^3(x, y, z, t) - \phi(x, y, z, t) - \epsilon^2 \Delta \phi(x, y, z, t)\right].$$

The incompressible Naiver–Stokes (NS) equations [9]:

$$\begin{aligned} \frac{\partial \mathbf{u}(x, y, z, t)}{\partial t} + \mathbf{u}(x, y, z, t) \cdot \nabla \mathbf{u}(x, y, z, t) &= -\frac{1}{\rho} \nabla p(x, y, z, t) + \nu \Delta \mathbf{u}(x, y, z, t), \\ \nabla \cdot \mathbf{u}(x, y, z, t) &= 0, \end{aligned}$$

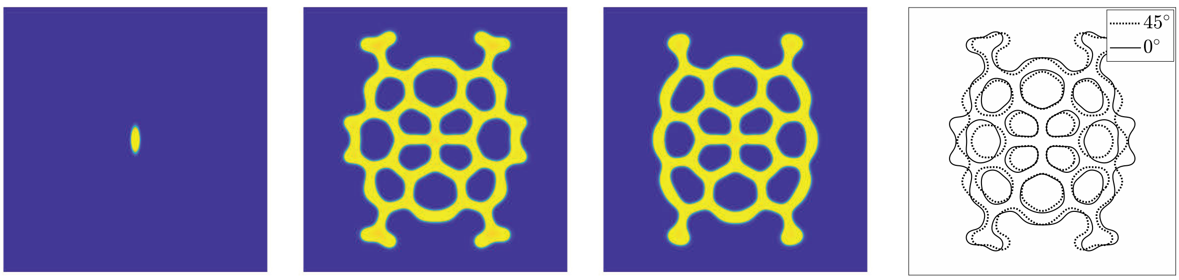
where $\mathbf{u}(x, y, z, t) = (u(x, y, z, t), v(x, y, z, t), w(x, y, z, t))$ is the fluid velocity, $p(x, y, z, t)$ is the pressure field, ρ is density, and ν is kinematic viscosity.

An accurate and parallel method to solve the anisotropic phase-field dendritic crystal growth model was proposed in [21]. The authors used first-order operator splitting method for the second derivative. In [18], the authors researched the dynamics of dislocations block copolymer system carried out by two-dimensional CH model. For the numerical results, the cell dynamics method was used with isotropic Laplacian operator.

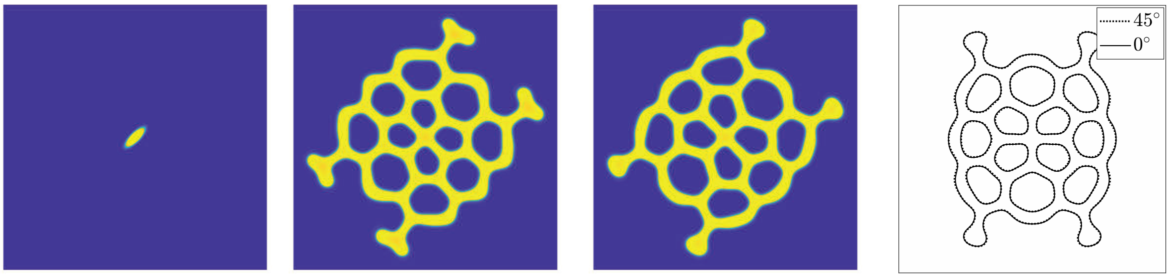
In general, it is difficult and in many cases not known to find closed-form analytic solutions of the above-mentioned equations with nontrivial initial and boundary conditions. Therefore, we need to use numerical methods to approximate the solutions of the equations. The numerical methods are recalled as finite volume method (FVM), finite element method (FEM) [19], spectral methods, and finite difference method (FDM) [13]. The AC and CH equations have been extensively used in modeling and simulating pattern formations. However, if there is anisotropic pattern during evolution of the numerical solutions, then it is difficult to conclude whether the pattern is due to the governing equation or numerical anisotropic problem. Fig.1 shows the numerical results of a tumor growth model, which is an example of a real-world mathematical model problem. In Fig.1, (a) is an elliptical initial condition. (b) and (c) are the computational results from the anisotropic and isotropic schemes, respectively, at later time. (e) is 45 ° rotated elliptical initial condition. (f) and (g) are the computational results from the anisotropic and isotropic schemes, respectively, at later time. (d) and (h) are the overlapped contours of the results ((b) and (f); (c) and (g)) for the anisotropic and isotropic schemes, respectively, after being rotated. As shown in Fig.1, the isotropic Laplacian stencil has superiorities over anisotropic ones when applied to some real life mathematical problems.

The main purpose of this paper is to propose benchmark functions to quantitatively evaluate the isotropy of the discrete Laplacian operators in 2D and 3D spaces. The benchmark functions have analytic 2D and 3D Laplacian solutions so that we can exactly compute the errors between the numerical and analytic solutions.

The paper is organized as follows: In Section 2, 2D and 3D isotropic discretizations of Laplacian operator are briefly described. In Section 3, several computational experiments are presented. In Section 4, conclusions are given.



(a) (b) (c) (d)



(e) (f) (g) (h)

Figure 1. (a) is an elliptical initial condition. (b) and (c) are the computational results from the anisotropic and isotropic schemes, respectively, at later time. (e) is 45 ° rotated elliptical initial condition. (f) and (g) are the computational results from the anisotropic and isotropic schemes, respectively, at the later time. (d) and (h) are the overlapped contours of the results ((b) and (f); (c) and (g)) for the anisotropic and isotropic schemes, respectively, after being rotated. Reprinted from [25] with permission from Hindawi.

2. Isotropic discretization of Laplacian operator

2.1. Two-dimensional space. Let $\Omega = [L_x,\ R_x] \times [L_y,\ R_y]$ be a computational domain and be discretized as $\Omega_h = \{(x_i, y_j)|x_i = L_x + (i-1)h,\ i = 1, \cdots, N_x \text{ and } y_j = L_y + (j-1)h,\ j = 1, \cdots, N_y\}$. Here, N_x and N_y are the positive integers and $h = (R_x - L_x)/(N_x - 1) = (R_y - L_y)/(N_y - 1)$ is the space grid size. By Taylor's theorem in two variables [8], it can be written as

$$\begin{aligned} u(x+a, y+b) &= u(x,y) + \left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y}\right)u(x,y) + \frac{1}{2!}\left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y}\right)^2 u(x,y) \\ &+\frac{1}{3!}\left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y}\right)^3 u(x,y) + \cdots + \frac{1}{(n-1)!}\left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y}\right)^{n-1} u(x,y) \\ &+\frac{1}{n!}\left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y}\right)^n u(x+\theta a, y+\theta b),\ 0<\theta<1. \end{aligned} \tag{1}$$

Let us denote $u_{ij} = u(x_i, y_j)$ for simplicity of the notation. Using (1), we obtain the following equations:

$$u_{i+1,j} = \left(u + hu_x + \frac{h^2}{2}u_{xx} + \frac{h^3}{6}u_{xxx} + \frac{h^4}{24}u_{xxxx}\right)_{ij} + O(h^5), \tag{2}$$

$$u_{i-1,j} = \left(u - hu_x + \frac{h^2}{2}u_{xx} - \frac{h^3}{6}u_{xxx} + \frac{h^4}{24}u_{xxxx}\right)_{ij} + O(h^5), \tag{3}$$

$$u_{i,j+1} = \left(u + hu_y + \frac{h^2}{2}u_{yy} + \frac{h^3}{6}u_{yyy} + \frac{h^4}{24}u_{yyyy}\right)_{ij} + O(h^5), \tag{4}$$

$$u_{i,j-1} = \left(u - hu_y + \frac{h^2}{2}u_{yy} - \frac{h^3}{6}u_{yyy} + \frac{h^4}{24}u_{yyyy}\right)_{ij} + O(h^5), \tag{5}$$

$$\begin{aligned} u_{i+1,j+1} &= \Big(u + hu_x + hu_y + \frac{h^2}{2}u_{xx} + h^2u_{xy} + \frac{h^2}{2}u_{yy} + \frac{h^3}{6}u_{xxx} + \frac{h^3}{2}u_{xxy} + \frac{h^3}{2}u_{xyy} \\ &+\frac{h^3}{6}u_{yyy} + \frac{h^4}{24}u_{xxxx} + \frac{h^4}{6}u_{xxxy} + \frac{h^4}{4}u_{xxyy} + \frac{h^4}{6}u_{xyyy} + \frac{h^4}{24}u_{yyyy}\Big)_{ij} + O(h^5), \end{aligned} \tag{6}$$

$$\begin{aligned} u_{i-1,j+1} &= \Big(u - hu_x + hu_y + \frac{h^2}{2}u_{xx} - h^2u_{xy} + \frac{h^2}{2}u_{yy} - \frac{h^3}{6}u_{xxx} + \frac{h^3}{2}u_{xxy} - \frac{h^3}{2}u_{xyy} \\ &+\frac{h^3}{6}u_{yyy} + \frac{h^4}{24}u_{xxxx} - \frac{h^4}{6}u_{xxxy} + \frac{h^4}{4}u_{xxyy} - \frac{h^4}{6}u_{xyyy} + \frac{h^4}{24}u_{yyyy}\Big)_{ij} + O(h^5), \end{aligned} \tag{7}$$

$$
\begin{aligned}
u_{i+1,j-1} &= \Big(u + hu_x - hu_y + \frac{h^2}{2}u_{xx} - h^2 u_{xy} + \frac{h^2}{2}u_{yy} + \frac{h^3}{6}u_{xxx} - \frac{h^3}{2}u_{xxy} + \frac{h^3}{2}u_{xyy} \qquad (8)\\
&\quad -\frac{h^3}{6}u_{yyy} + \frac{h^4}{24}u_{xxxx} - \frac{h^4}{6}u_{xxxy} + \frac{h^4}{4}u_{xxyy} - \frac{h^4}{6}u_{xyyy} + \frac{h^4}{24}u_{yyyy}\Big)_{ij} + O(h^5),\\
u_{i-1,j-1} &= \Big(u - hu_x - hu_y + \frac{h^2}{2}u_{xx} + h^2 u_{xy} + \frac{h^2}{2}u_{yy} - \frac{h^3}{6}u_{xxx} - \frac{h^3}{2}u_{xxy} - \frac{h^3}{2}u_{xyy} \qquad (9)\\
&\quad -\frac{h^3}{6}u_{yyy} + \frac{h^4}{24}u_{xxxx} + \frac{h^4}{6}u_{xxxy} + \frac{h^4}{4}u_{xxyy} + \frac{h^4}{6}u_{xyyy} + \frac{h^4}{24}u_{yyyy}\Big)_{ij} + O(h^5),
\end{aligned}
$$

where $(a,b) = (h,0)$, $(-h,0)$, $(0,h)$, $(0,-h)$, (h,h), $(-h,h)$, $(h,-h)$, and $(-h,-h)$ are used for (2)–(9), respectively. Summing (2)–(5) results, we obtain

$$
u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} = 4u_{ij} + h^2 (u_{xx} + u_{yy})_{ij} + \frac{h^4}{12}(u_{xxxx} + u_{yyyy})_{ij} + O(h^5). \quad (10)
$$

Summing (6)–(9) results, we have

$$
\begin{aligned}
u_{i+1,j+1} + u_{i-1,j+1} + u_{i+1,j-1} + u_{i-1,j-1} &= 4u_{ij} + 2h^2 (u_{xx} + u_{yy})_{ij}\\
&\quad + \frac{h^4}{6}(u_{xxxx} + 6u_{xxyy} + u_{yyyy})_{ij} + O(h^5). \quad (11)
\end{aligned}
$$

After multiplying weights w and $(1-w)$ to (10) and (11), respectively, then by summing them, we get

$$
\begin{aligned}
\Delta u_{ij} &= (u_{xx} + u_{yy})_{ij} = \frac{1}{(2-w)h^2}\Big(w(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) - 4u_{ij} + (1-w)(u_{i+1,j+1}\\
&\quad + u_{i-1,j+1} + u_{i+1,j-1} + u_{i-1,j-1})\Big) - \frac{h^2}{12}\left(u_{xxxx} + \frac{12(1-w)}{2-w}u_{xxyy} + u_{yyyy}\right)_{ij} + O(h^3)\\
&= \frac{1}{(2-w)h^2}\Big(w(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) - 4u_{ij} + (1-w)(u_{i+1,j+1} + u_{i-1,j+1}\\
&\quad + u_{i+1,j-1} + u_{i-1,j-1})\Big) - \frac{h^2}{12}\left(\Delta^2 u + \frac{8-10w}{2-w}u_{xxyy}\right)_{ij} + O(h^3) \qquad (12)\\
&= \frac{1}{(2-w)h^2}\Big(w(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) - 4u_{ij} + (1-w)(u_{i+1,j+1} + u_{i-1,j+1}\\
&\quad + u_{i+1,j-1} + u_{i-1,j-1})\Big) + O(h^2).
\end{aligned}
$$

We should note that there exists anisotropic term $(8-10w)u_{xxyy}/(2-w)$ in (12) unless $w = 4/5$ in the leading order. In fact, this $w = 4/5$ value is unique to make the scheme be isotropic in the lowest order. For $0 \le w \le 1$, we define discrete Laplacian operator as

$$
\begin{aligned}
\Delta_w u_{ij} &= \frac{1}{(2-w)h^2}\Big(w(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) - 4u_{ij} + (1-w)(u_{i+1,j+1}\\
&\quad + u_{i-1,j+1} + u_{i+1,j-1} + u_{i-1,j-1})\Big). \qquad (13)
\end{aligned}
$$

Therefore, if $w = 1$ in the definition (13), then we have the standard 5-point stencil for the 2D Laplacian operator as follow:

$$
\Delta_1 u_{ij} = \frac{u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{ij}}{h^2}. \quad (14)
$$

If $w = 4/5$ in the definition (13), then we have the 9-point isotropic stencil for the 2D Laplacian operator:

$$\Delta_{\frac{4}{5}} u_{ij} = \frac{4(u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1}) + u_{i-1,j+1} + u_{i+1,j+1} + u_{i+1,j-1} + u_{i-1,j-1} - 20u_{ij}}{6h^2},$$

which is the unique 2D isotropic discretization in the leading order error [10].

Fig.2(a) and (b) represent the numerical stencils for the standard 5-point and 9-point stencils for the 2D Laplacian operator, respectively.

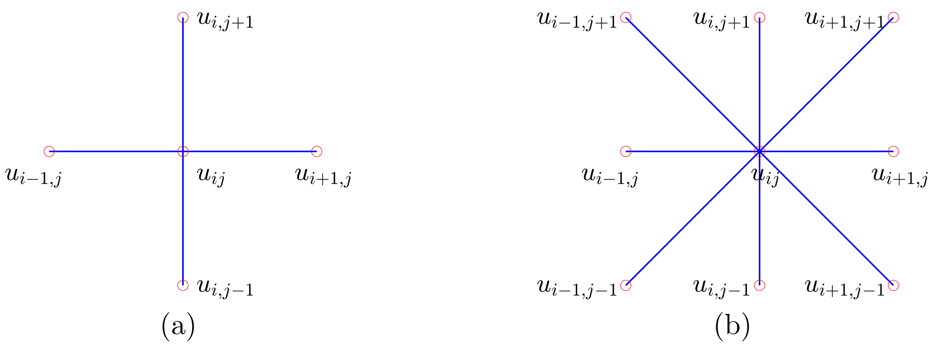


Figure 2. Stencils for the standard 5-point (a) and 9-point (b) stencils for the 2D Laplacian operator.

The Laplacian operator, $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2$, is invariant under rotation in two-dimensional space [11]. To show this property, let us consider the following rotated variables (x', y') of (x, y):

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix},$$

where θ is the angle of rotation. Then, $u_{xx} + u_{yy} = (\cos^2\theta + \sin^2\theta)(u_{x'x'} + u_{y'y'}) = u_{x'x'} + u_{y'y'}$. Therefore, Laplacian operator is rotationally invariant. However, the standard 5-point discrete Laplacian operator (14) is not isotropic because there exists anisotropic term $-2u_{xxyy}$ in (12).

We should note that there is a 7-point hexagonal scheme [5] which gives $O(h^4)$ order of accuracy, which is the same with the 9-point classical difference scheme under some smoothness conditions of the boundary functions and conjugation conditions. Furthermore, in [5], the authors showed that 7-point stencil is more efficient in computational time because only 7 nonzero diagonals occur in the coefficient matrix instead of 9 nonzero diagonals of the obtained system of algebraic equations.

2.2. **Three-dimensional space.** Let $\Omega = [L_x,\ R_x] \times [L_y,\ R_y] \times [L_z,\ R_z]$ be a computational domain and be discretized as $\Omega_h = \{(x_i, y_j, z_k) | x_i = L_x + (i-1)h,\ i = 1, \cdots, N_x,\ y_j = L_y + (j-1)h,\ j = 1, \cdots, N_y,$ and $z_k = L_z + (k-1)h,\ k = 1, \cdots, N_z\}$. Here, N_x, N_y, and N_z are the positive integers and $h = (R_x - L_x)/(N_x - 1) = (R_y - L_y)/(N_y - 1) = (R_z - L_z)/(N_z - 1)$ is the space grid size. By Taylor's theorem in three variables [8], it can be written as

$$\begin{aligned} u(x+a, y+b, z+c) &= u(x,y,z) + \left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y} + c\frac{\partial}{\partial z}\right) u(x,y,z) \\ &+ \frac{1}{2!}\left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y} + c\frac{\partial}{\partial z}\right)^2 u(x,y,z) \\ &+ \frac{1}{3!}\left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y} + c\frac{\partial}{\partial z}\right)^3 u(x,y,z) + \cdots + \frac{1}{(n-1)!}\left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y} + c\frac{\partial}{\partial z}\right)^{n-1} u(x,y,z) \\ &+ \frac{1}{n!}\left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y} + c\frac{\partial}{\partial z}\right)^n u(x+\theta a, y+\theta b, z+\theta c),\ 0 < \theta < 1. \end{aligned} \tag{15}$$

Let us denote $u_{ijk} = u(x_i, y_j, z_k)$ for simplicity of the notation. For non-negative values of α, β, and γ, letting a, b, c be one of values $-h, 0, h$ in (15) with $n = 6$, we have

$$\begin{aligned}
&\alpha(u_{i+1,jk} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1}) + \beta(u_{i+1,j,k-1} + u_{i-1,j,k-1} \\
&+u_{i,j+1,k-1} + u_{i,j-1,k-1} + u_{i+1,j+1,k} + u_{i-1,j+1,k} + u_{i+1,j-1,k} + u_{i-1,j-1,k} + u_{i+1,j,k+1} \\
&+u_{i-1,j,k+1} + u_{i,j+1,k+1} + u_{i,j-1,k+1}) + \gamma(u_{i+1,j+1,k-1} + u_{i-1,j+1,k-1} + u_{i+1,j-1,k-1} \\
&+u_{i-1,j-1,k-1} + u_{i+1,j+1,k+1} + u_{i-1,j+1,k+1} + u_{i+1,j-1,k+1} + u_{i-1,j-1,k+1}) \\
&= (6\alpha + 12\beta + 8\gamma)u_{ijk} + h^2(\alpha + 4\beta + 4\gamma)(u_{xx} + u_{yy} + u_{zz})_{ijk} \\
&+\frac{h^4}{12}[(\alpha + 4\beta + 4\gamma)(u_{xxxx} + u_{yyyy} + u_{zzzz}) + (12\beta + 24\gamma)(u_{xxyy} + u_{yyzz} + u_{xxzz})]_{ijk} + O(h^6).
\end{aligned} \tag{16}$$

From (16), we have

$$\begin{aligned}
&\Delta u_{ijk} = (u_{xx} + u_{yy} + u_{zz})_{ijk} \\
&= \frac{1}{h^2(\alpha + 4\beta + 4\gamma)}\Big[\alpha(u_{i+1,jk} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1}) \\
&+\beta(u_{i+1,j,k-1} + u_{i-1,j,k-1} + u_{i,j+1,k-1} + u_{i,j-1,k-1} + u_{i+1,j+1,k} + u_{i-1,j+1,k} + u_{i+1,j-1,k} \\
&+u_{i-1,j-1,k} + u_{i+1,j,k+1} + u_{i-1,j,k+1} + u_{i,j+1,k+1} + u_{i,j-1,k+1}) + \gamma(u_{i+1,j+1,k-1} + u_{i-1,j+1,k-1} \\
&+u_{i+1,j-1,k-1} + u_{i-1,j-1,k-1} + u_{i+1,j+1,k+1} + u_{i-1,j+1,k+1} + u_{i+1,j-1,k+1} + u_{i-1,j-1,k+1}) \\
&-(6\alpha + 12\beta + 8\gamma)u_{ijk}\Big] - \frac{h^4}{12}\Big[u_{xxxx} + u_{yyyy} + u_{zzzz} + \frac{12\beta + 24\gamma}{\alpha + 4\beta + 4\gamma}(u_{xxyy} + u_{yyzz} + u_{xxzz})\Big]_{ijk} \\
&+O(h^6).
\end{aligned}$$

Without loss of generality, we can assume $\alpha + \beta + \gamma = 1$. If we choose $\alpha = 1$, $\beta = 0$, and $\gamma = 0$, we have the 3D 7-point stencil standard discrete Laplacian operator as follow

$$\Delta_S u_{ijk} = \frac{u_{i+1,j,k} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1} - 6u_{ijk}}{h^2}. \tag{17}$$

However, in this case, we have the term related to h^4 as $[u_{xxxx} + u_{yyyy} + u_{zzzz}]_{ijk}$, which is not rotationally invariant. To make the term related to h^4 be a three-dimensional biharmonic term, we require $12\beta + 24\gamma = 2(\alpha + 4\beta + 4\gamma)$, i.e., $3\beta + 9\gamma = 1$. Then,

$$\begin{aligned}
&[u_{xxxx} + u_{yyyy} + u_{zzzz} + \frac{12\beta + 24\gamma}{\alpha + 4\beta + 4\gamma}(u_{xxyy} + u_{yyzz} + u_{xxzz})]_{ijk} \\
&= [u_{xxxx} + u_{yyyy} + u_{zzzz} + 2(u_{xxyy} + u_{yyzz} + u_{xxzz})]_{ijk} = (\Delta^2 u)_{ijk}.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
&(\Delta u)_{ijk} = (u_{xx} + u_{yy} + u_{zz})_{ijk} \\
&= \frac{1}{h^2(\alpha + 4\beta + 4\gamma)}\Big[\alpha(u_{i+1,jk} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1}) \\
&+\beta(u_{i+1,j,k-1} + u_{i-1,j,k-1} + u_{i,j+1,k-1} + u_{i,j-1,k-1} + u_{i+1,j+1,k} + u_{i-1,j+1,k} + u_{i+1,j-1,k} \\
&+u_{i-1,j-1,k} + u_{i+1,j,k+1} + u_{i-1,j,k+1} + u_{i,j+1,k+1} + u_{i,j-1,k+1}) + \gamma(u_{i+1,j+1,k-1} + u_{i-1,j+1,k-1} \\
&+u_{i+1,j-1,k-1} + u_{i-1,j-1,k-1} + u_{i+1,j+1,k+1} + u_{i-1,j+1,k+1} + u_{i+1,j-1,k+1} + u_{i-1,j-1,k+1}) \\
&-(6\alpha + 12\beta + 8\gamma)u_{ijk}\Big] - \frac{h^4}{12}(\Delta^2 u)_{ijk} + O(h^6).
\end{aligned} \tag{18}$$

From (18), let us define 3D isotropic discrete Laplacian operator as

$$\begin{aligned}
&(\Delta_\beta u)_{ijk} = (u_{xx} + u_{yy} + u_{zz})_{ijk} \\
&= \frac{1}{h^2(\alpha + 4\beta + 4\gamma)}\Big[\alpha(u_{i+1,jk} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1}) \\
&+\beta(u_{i+1,j,k-1} + u_{i-1,j,k-1} + u_{i,j+1,k-1} + u_{i,j-1,k-1} + u_{i+1,j+1,k} + u_{i-1,j+1,k} + u_{i+1,j-1,k}
\end{aligned} \tag{19}$$

$$+u_{i-1,j-1,k} + u_{i+1,j,k+1} + u_{i-1,j,k+1} + u_{i,j+1,k+1} + u_{i,j-1,k+1}) + \gamma(u_{i+1,j+1,k-1} + u_{i-1,j+1,k-1}$$
$$+u_{i+1,j-1,k-1} + u_{i-1,j-1,k-1} + u_{i+1,j+1,k+1} + u_{i-1,j+1,k+1} + u_{i+1,j-1,k+1} + u_{i-1,j-1,k+1})$$
$$-(6\alpha + 12\beta + 8\gamma)u_{ijk}\Big],$$

where $0 \leq \beta \leq 1/3$, $\gamma = (1-3\beta)/9$, and $\alpha = 1-\beta-\gamma$. We note that there are infinitely many 3D isotropic 27-point stencil discrete Laplacian operators. The Laplacian operator, $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2 + \partial^2/\partial z^2$, is invariant under rotation in three-dimensional space [10]. To show this property, let us consider the following rotated variables (x', y', z'):

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix},$$

where R is a three-dimensional rotation matrix and is given as

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}.$$

The rotational matrix R is an orthogonal matrix, i.e., $RR^T = R^T R = I$ and

$$\sum_{k=1}^{3} r_{ik} r_{jk} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

Then, using chain rule, we have

$$\begin{aligned} u_{xx} + u_{yy} + u_{zz} &= 2(r_{11}r_{21} + r_{12}r_{22} + r_{13}r_{23})u_{x'y'} + 2(r_{21}r_{31} + r_{12}r_{32} + r_{23}r_{33})u_{y'z'} \\ &\quad +2(r_{31}r_{11} + r_{32}r_{12} + r_{33}r_{13})u_{z'x'} + (r_{11}^2 + r_{12}^2 + r_{13}^2)u_{x'x'} \\ &\quad +(r_{21}^2 + r_{22}^2 + r_{23}^2)u_{y'y'} + (r_{31}^2 + r_{32}^2 + r_{33}^2)u_{z'z'} \\ &= u_{x'x'} + u_{y'y'} + u_{z'z'}, \end{aligned}$$

where we have used (20). Therefore, Laplacian operator is rotationally invariant. If we choose $\alpha = 20/27$, $\beta = 2/9$, and $\gamma = 1/27$, then (19) becomes

$$(\Delta_{\frac{2}{9}} u)_{ijk} = \frac{1}{48h^2}\Big[20(u_{i+1,jk} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1}) - 200u_{ijk}$$
$$+6(u_{i+1,j,k-1} + u_{i-1,j,k-1} + u_{i,j+1,k-1} + u_{i,j-1,k-1} + u_{i+1,j+1,k} + u_{i-1,j+1,k} + u_{i+1,j-1,k}$$
$$+u_{i-1,j-1,k} + u_{i+1,j,k+1} + u_{i-1,j,k+1} + u_{i,j+1,k+1} + u_{i,j-1,k+1}) + u_{i+1,j+1,k-1} + u_{i-1,j+1,k-1}$$
$$+u_{i+1,j-1,k-1} + u_{i-1,j-1,k-1} + u_{i+1,j+1,k+1} + u_{i-1,j+1,k+1} + u_{i+1,j-1,k+1} + u_{i-1,j-1,k+1}\Big],$$

which was introduced in [14]. In [16], the authors used $\alpha = 7/9$, $\beta = 1/6$, and $\gamma = 1/18$ for the 3D isotropic discretization for the Laplacian operator:

$$(\Delta_{\frac{1}{6}} u)_{ijk} = \frac{1}{30h^2}\Big[14(u_{i+1,jk} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1}) - 128u_{ijk}$$
$$+3(u_{i+1,j,k-1} + u_{i-1,j,k-1} + u_{i,j+1,k-1} + u_{i,j-1,k-1} + u_{i+1,j+1,k} + u_{i-1,j+1,k} + u_{i+1,j-1,k}$$
$$+u_{i-1,j-1,k} + u_{i+1,j,k+1} + u_{i-1,j,k+1} + u_{i,j+1,k+1} + u_{i,j-1,k+1}) + u_{i+1,j+1,k-1} + u_{i-1,j+1,k-1}$$
$$+u_{i+1,j-1,k-1} + u_{i-1,j-1,k-1} + u_{i+1,j+1,k+1} + u_{i-1,j+1,k+1} + u_{i+1,j-1,k+1} + u_{i-1,j-1,k+1}\Big].$$

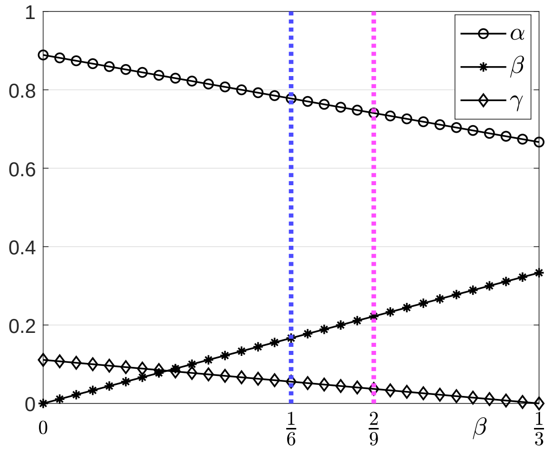


Figure 3. $0 \leq \beta \leq 1/3$, $\gamma = (1-3\beta)/9$, and $\alpha = 1-\beta-\gamma$.

Fig.3 shows the values of α, β, and γ for 3D isotropic discretization against β on $0 \leq \beta \leq 1/3$ with the vertical lines of $\beta = 1/6$ and $\beta = 2/9$.

3. Numerical experiments

3.1. **Two-dimensional space.** Let us consider the following function on domain $\Omega = [-1.5,\ 1.5]\times[-1.5,\ 1.5]$:

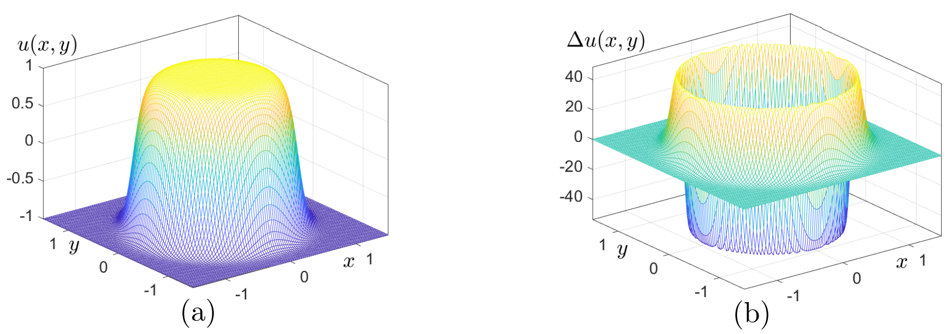
$$u(x,y) = \tanh\left(\frac{1-x^2-y^2}{\epsilon}\right), \tag{21}$$

where ϵ is a parameter related to the thickness of interfacial transition layer. Here, $\epsilon = 0.25$ is used. Then, its Laplacian function is given as

$$\Delta u(x,y) = -\left[\frac{8}{\epsilon^2}(x^2+y^2)\tanh\left(\frac{1-x^2-y^2}{\epsilon}\right)+\frac{4}{\epsilon}\right]\operatorname{sech}^2\left(\frac{1-x^2-y^2}{\epsilon}\right).$$

For $\epsilon = 0.25$, Fig.4(a) and (b) show $u(x,y)$ and $\Delta u(x,y)$, respectively. Fig.4(c) and (d) are mesh plots of $\Delta_1 u(x,y) - \Delta u(x,y)$ and $\Delta_{\frac{4}{5}} u(x,y) - \Delta u(x,y)$, respectively. Fig.4(e) and (f) are filled contours of $\Delta_1 u(x,y) - \Delta u(x,y)$ and $\Delta_{\frac{4}{5}} u(x,y) - \Delta u(x,y)$, respectively. The grid size is 101×101.

As shown in Fig.4(c), in the case of $\Delta_1 u(x,y)$, the errors are large in the directions of $0°$, $90°$, $180°$, and $270°$. As shown in Fig.4(d), in the case of $\Delta_{\frac{4}{5}} u(x,y)$, the errors are radially uniformly distributed. Because the given function $u(x,y)$ in (21) is radially symmetric, its Laplacian should be radially symmetric.



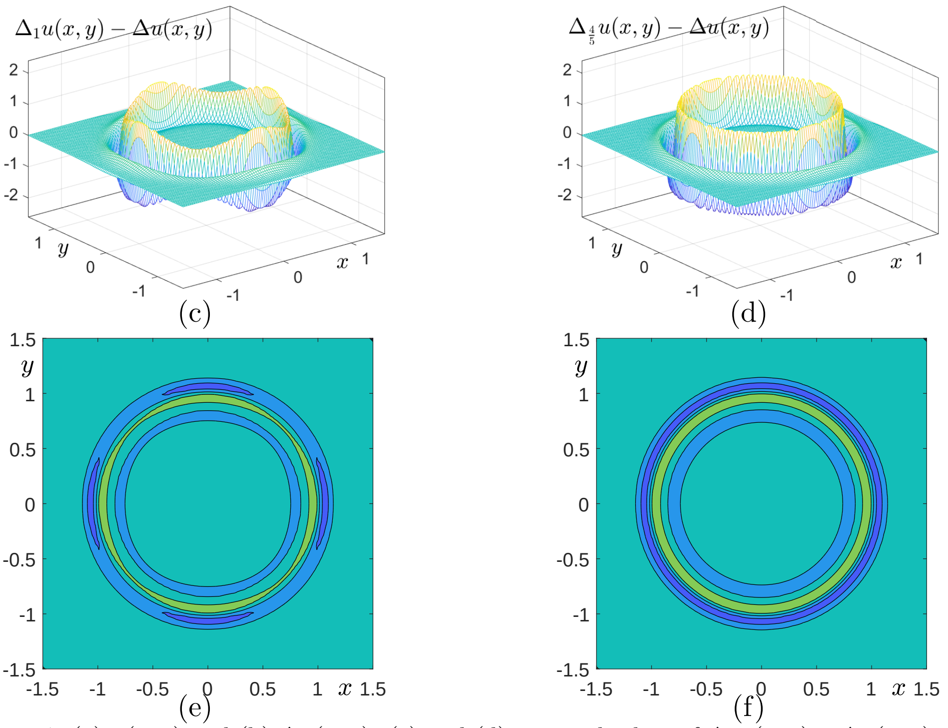


Figure 4. (a) $u(x, y)$ and (b) $\Delta u(x, y)$. (c) and (d) are mesh plots of $\Delta_1 u(x, y) - \Delta u(x, y)$ and $\Delta_{\frac{4}{5}} u(x, y) - \Delta u(x, y)$, respectively. (e) and (f) are filled contours of $\Delta_1 u(x, y) - \Delta u(x, y)$ and $\Delta_{\frac{4}{5}} u(x, y) - \Delta u(x, y)$, respectively.

Fig.5(a) and (b) are graph plots of $\left(\sqrt{x_i^2 + y_j^2},\ \Delta_1 u_{ij} - \Delta u(x_i, y_j)\right)$ and $\left(\sqrt{x_i^2 + y_j^2},\ \Delta_{\frac{4}{5}} u_{ij} - \Delta u(x_i, y_j)\right)$, respectively. We can observe that the errors of the standard 5-point and isotropic stencils are not radially symmetric and radially symmetric, respectively.

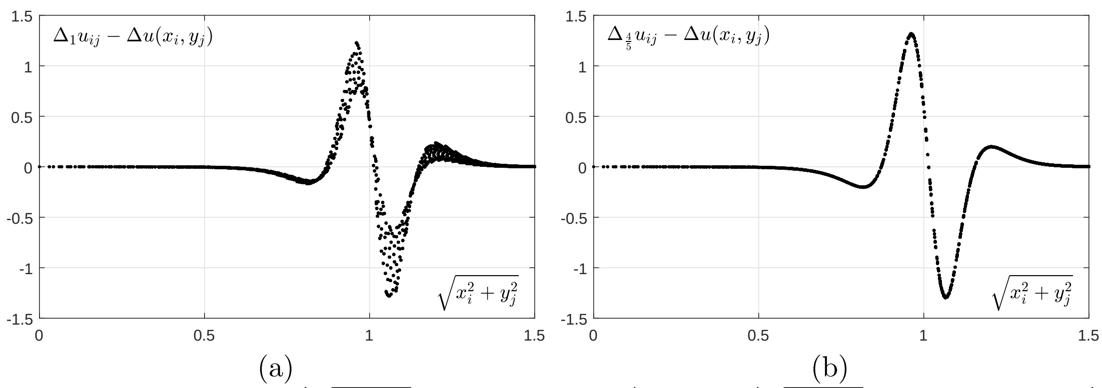


Figure 5. Graph plots of (a) $\left(\sqrt{x_i^2 + y_j^2},\ \Delta_1 u_{ij} - \Delta u(x_i, y_j)\right)$ and (b) $\left(\sqrt{x_i^2 + y_j^2},\ \Delta_{\frac{4}{5}} u_{ij} - \Delta u(x_i, y_j)\right)$.

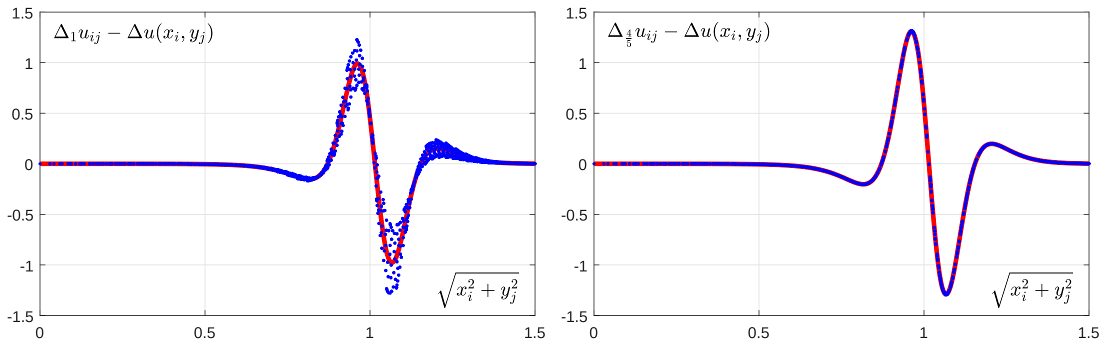
To quantitatively measure the radial symmetry of the numerical Laplacian operator, let us consider a cubic spline least squares approximation to the errors. Let us define the error between the numerical approximation and analytic solution at position (x_i, y_j) as $e_{w,ij} = \Delta_w u_{ij} - \Delta u(x_i, y_j)$. We uniformly partition interval $[0,\ 1.5]$ into N_r subintervals, which implies $(N_r + 1)$ node points, i.e., $r_i = 0.15i/N_r$ for $i = 0, 1, \ldots, N_r$. In this study, we use $N_r = 50$, which is a sufficiently large enough number. Let $f(r)$ be a discrete function which is defined on the $(N_r + 1)$ node points. Our goal is to construct the discrete function $f(r)$ which best fits the given data in the sense of least squares approximation when we use a cubic spine based on the function $f(r)$. To construct the values $f(r_i)$ for $i = 0, 1, \ldots, N_r$, we use the nonlinear curve-fitting routine **lsqcurvefit** in MATLAB R2022a, which is a nonlinear least-squares optimization function [17]. That is, we compute the optimal discrete function values $f(r_i)$ for $i = 0, 1, \ldots, N_r$, which minimize the following cost function:

$$\mathcal{E}(f(r_0), \ldots, f(r_{N_r})) = \frac{1}{2} \sum_{\sqrt{x_i^2+y_j^2} \le 1.5} \left[e_{w,ij} - S\left(\sqrt{x_i^2 + y_j^2}\right)\right]^2,$$

where $S(r)$ is the cubic spline interpolant using the discrete function values $f(r_i)$ for $i = 0, 1, \ldots, N_r$. The specific usage of the nonlinear curve-fitting routine **lsqcurvefit** is as follows:

$$[f(r_0), \ldots, f(r_{N_r})] = \textbf{lsqcurvefit}(\text{'fun'},\ [f^0(r_0), \ldots, f^0(r_{N_r})],\ \mathbf{R},\ \mathbf{E}),$$

where $[f(r_0), \ldots, f(r_{N_r})]$ are the optimized discrete function values, 'fun' is the cubic spline interpolant, $[f^0(r_0), \ldots, f^0(r_{N_r})]$ are the initial guess of the function $f(r)$, $\mathbf{R} = \{R | R = \sqrt{x_i^2 + y_j^2} \le 1.5\}$ is the set of radii, and corresponding error values $\mathbf{E} = \{e_{w,ij} | \sqrt{x_i^2 + y_j^2} \le 1.5\}$. We compute the discrete l_2-norm of the difference between the errors and best fitting function. The discrete l_2-norms of the numerical results in Figs.6(a) and (b) are 0.0775 and 0.0017, respectively. Table 1 lists errors of the two different Laplacian operators, Δ_1 and $\Delta_{4/5}$, for various ϵ values. Fig.7 shows the difference data in Table 1 as a graph for ϵ. We can clearly observe that the isotropic discretization of the Laplacian operator is superior to the standard discretization.



(a) (b)

Figure 6. Graph plots of (a) $\left(\sqrt{x_i^2 + y_j^2},\ \Delta_1 u_{ij} - \Delta u(x_i, y_j)\right)$ and (b) $\left(\sqrt{x_i^2 + y_j^2},\ \Delta_{\frac{4}{5}} u_{ij} - \Delta u(x_i, y_j)\right)$ with corresponding cubic spline interpolants (solid curves) using $\epsilon = 0.25$.

Table 1. Discrete l_2-norm of the difference between the errors and best fitting function for two different Laplacian operator cases for each various $\epsilon = 0.1$, 0.15, 0.2, 0.25, 0.3, and 0.35 on two-dimensional space.

case	$\epsilon = 0.1$	$\epsilon = 0.15$	$\epsilon = 0.2$	$\epsilon = 0.25$	$\epsilon = 0.3$	$\epsilon = 0.35$
Δ_1	1.6172	0.4259	0.1631	0.0775	0.0423	0.0255
$\Delta_{4/5}$	0.4642	0.0438	0.0062	0.0017	0.0006	0.0003

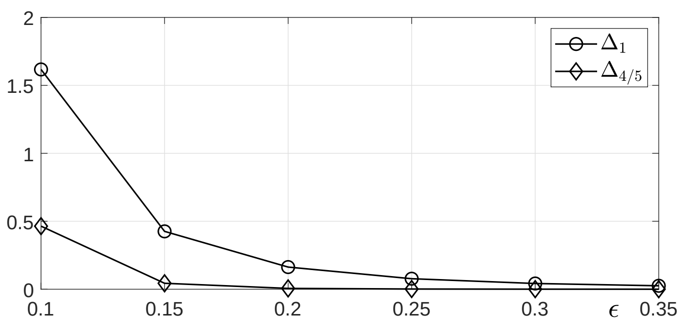


Figure 7. Discrete l_2-error for Δ_1 and $\Delta_{4/5}$ on two-dimensional space.

3.2. **Three-dimensional space.** Let us consider the following function on domain $\Omega = [-1.5,\ 1.5] \times [-1.5,\ 1.5] \times [-1.5,\ 1.5]$:

$$u(x, y, z) = \tanh\left(\frac{1 - x^2 - y^2 - z^2}{\epsilon}\right). \tag{22}$$

Then, its Laplacian function is given as

$$\Delta u(x, y, z) = -\left[\frac{8}{\epsilon^2}(x^2 + y^2 + z^2)\tanh\left(\frac{1 - x^2 - y^2 - z^2}{\epsilon}\right) + \frac{6}{\epsilon}\right]\operatorname{sech}^2\left(\frac{1 - x^2 - y^2 - z^2}{\epsilon}\right). \tag{23}$$

Because the given function $u(x, y, z)$ in (22) is spherically symmetric, its Laplacian should be spherically symmetric, i.e., (23). For $\epsilon = 0.25$, Figs.8(a), (b), and (c) are isosurfaces of $u(x, y, z)$ at level zero, $\Delta_S u(x, y, z) - \Delta u(x, y, z)$ at level 2.7, and $\Delta_{\frac{2}{9}} u(x, y, z) - \Delta u(x, y, z)$ at level 2.7, respectively. The grid size is 51×51. We can observe that the standard 7-point stencil generates the anisotropic result as shown in Fig.8(b). In the case of isotropic discretization, $(\Delta_{\frac{2}{9}} u)_{ijk}$, we have the good isotropic result as shown in Fig.8(c).

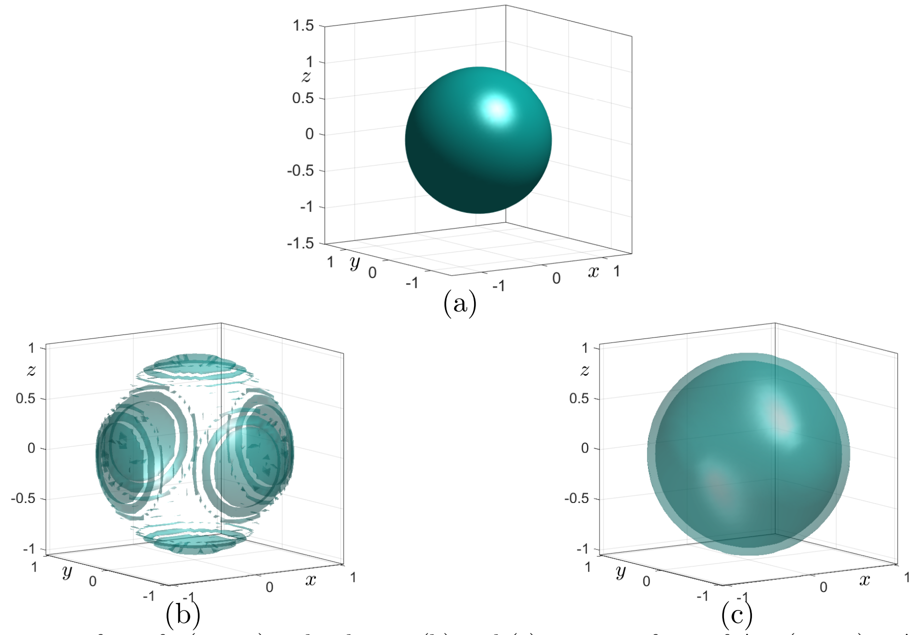


Figure 8. (a) is isosurface of $u(x, y, z)$ at level zero; (b) and (c) are isosurfaces of $\Delta_S u(x, y, z) - \Delta u(x, y, z)$ and $\Delta_{\frac{2}{9}} u(x, y, z) - \Delta u(x, y, z)$, respectively, at level 2.7.

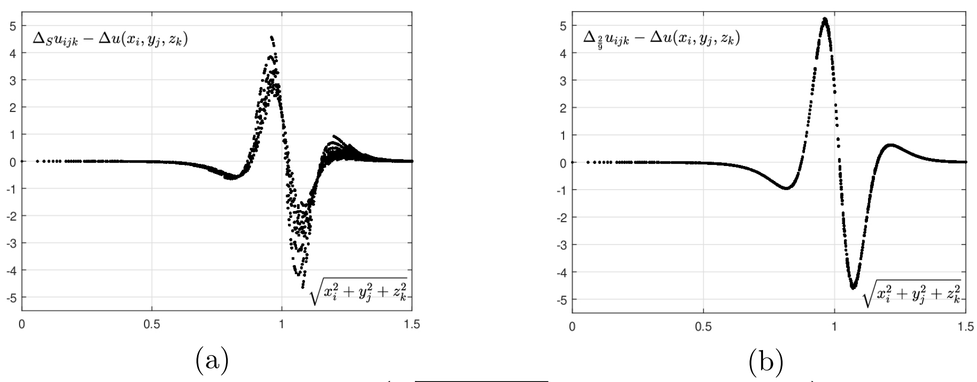


Figure 9. Graph plots of (a) $\left(\sqrt{x_i^2 + y_j^2 + z_k^2}, \Delta_S u_{ijk} - \Delta u(x_i, y_j, z_k)\right)$ and (b) $\left(\sqrt{x_i^2 + y_j^2 + z_k^2}, \Delta_{\frac{2}{9}} u_{ijk} - \Delta u(x_i, y_j, z_k)\right)$.

Fig.9(a) and (b) are graph plots of $\left(\sqrt{x_i^2+y_j^2+z_k^2}, \Delta_S u_{ijk} - \Delta u(x_i,y_j,z_k)\right)$ and $\left(\sqrt{x_i^2+y_j^2+z_k^2},\right.$ $\left.\Delta_{\frac{2}{9}} u_{ijk} - \Delta u(x_i,y_j,z_k)\right)$, respectively.

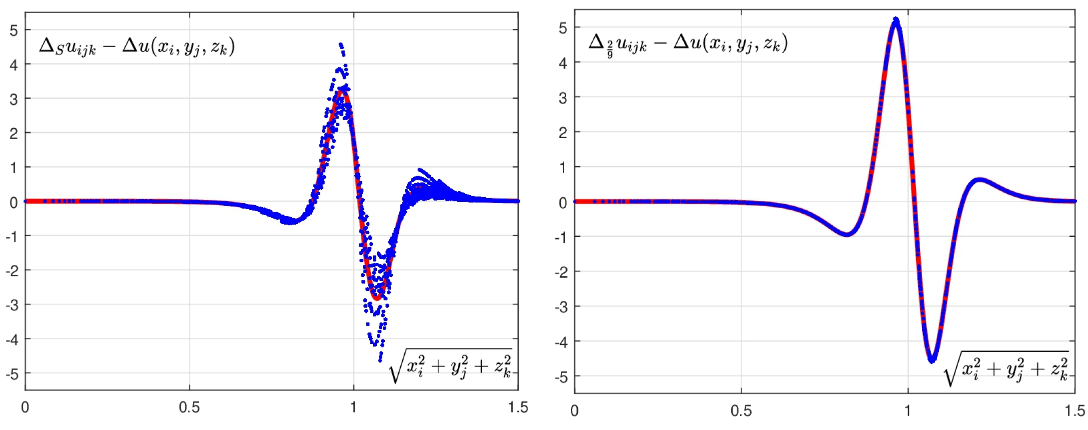
To quantitatively measure the spherical symmetry of the 3D numerical Laplacian operator, let us consider a cubic spline least squares approximation to the errors. Let us define the error between the numerical approximation and analytic solution at position (x_i, y_j, z_k) as $e_{\beta,ijk} = \Delta_\beta u_{ijk} - \Delta u(x_i,y_j,z_k)$. Let $f(r)$ be a discrete function which is defined on the (N_r+1) node points. Our goal is to construct the discrete function $f(r)$ which best fits the given data in the sense of least squares approximation when we use a cubic spine based on the function $f(r)$. That is, we compute the optimal discrete function values $f(r_i)$ for $i = 0, 1, \ldots, N_r$, which minimize the following cost function:

$$\mathcal{E}(f(r_0),\ldots,f(r_{N_r})) = \frac{1}{2} \sum_{\sqrt{x_i^2+y_j^2+z_k^2}\le 1.5} \left[e_{\beta,ijk} - S\left(\sqrt{x_i^2+y_j^2+z_k^2}\right)\right]^2,$$

where $S(r)$ is the cubic spline interpolant using the discrete function values $f(r_i)$ for $i = 0, 1, \ldots, N_r$. The specific usage of the nonlinear curve-fitting routine **lsqcurvefit** is as follows:

$$[f(r_0),\ldots,f(r_{N_r})] = \textbf{lsqcurvefit}(\text{‘fun’},\ [f^0(r_0),\ldots,f^0(r_{N_r})],\ \mathbf{R},\ \mathbf{E}),$$

where $[f(r_0),\ldots,f(r_{N_r})]$ are the optimized discrete function values, ‘fun’ is the cubic spline interpolant, $[f^0(r_0),\ldots,f^0(r_{N_r})]$ are the initial guess of the function $f(r)$, $\mathbf{R} = \{R | R = \sqrt{x_i^2+y_j^2+z_k^2} \le 1.5\}$ is the set of radii, and corresponding error values $\mathbf{E} = \{e_{\beta,ijk} | \sqrt{x_i^2+y_j^2+z_k^2} \le 1.5\}$. We compute the discrete l_2-norm of the difference between the errors and best fitting function $f(r)$. The discrete l_2-norms of the numerical results in Figs.10(a) and (b) are 0.2913 and 0.0176, respectively. Table 2 lists errors of the two different Laplacian operators, Δ_S and $\Delta_{2/9}$, for various ϵ values. Fig.7 shows the difference data in Table 2 as a graph for ϵ. We can clearly observe that the isotropic discretization of the Laplacian operator is superior to the standard discretization.



(a) (b)

Figure 10. Graph plots of (a) $\left(\sqrt{x_i^2+y_j^2+z_k^2}, \Delta_S u_{ijk} - \Delta u(x_i,y_j,z_k)\right)$ and (b) $\left(\sqrt{x_i^2+y_j^2+z_k^2}, \Delta_{\frac{2}{9}} u_{ijk} - \Delta u(x_i,y_j,z_k)\right)$ with corresponding cubic spline interpolants (solid curves).

Table 2. Discrete l_2-norm of the difference between the errors and best fitting function for two different Laplacian operator cases for each various $\epsilon = 0.1, 0.15, 0.2, 0.25, 0.3$, and 0.35 on three-dimensional space. Here, Δ_S is the 3D 7-point standard discrete Laplacian operator (17)

case	$\epsilon = 0.1$	$\epsilon = 0.15$	$\epsilon = 0.2$	$\epsilon = 0.25$	$\epsilon = 0.3$	$\epsilon = 0.35$
Δ_S	4.8169	1.4653	0.5913	0.2913	0.1634	0.1006
$\Delta_{2/9}$	1.3692	0.2211	0.0531	0.0176	0.0072	0.0034

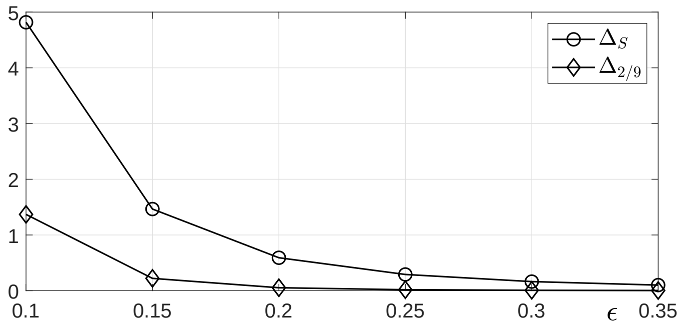


Figure 11. Discrete l_2-error for Δ_S and $\Delta_{2/9}$ on three-dimensional space.

Fig.12 shows the discrete l_2-norm of the difference between the errors and best fitting function against $0 \leq \beta \leq 1/3$ when we use (22). In Fig.12, when we choose $\alpha = 8/9$, $\beta = 0$, and $\gamma = 1/9$, the error is the largest. However, it makes (19) become the smallest 15-point isotropic stencil for the 3D discrete Laplacian operator as follows:

$$
\begin{aligned}
(\Delta_0 u)_{ijk} \;=\; & \frac{1}{12h^2}\Big[8(u_{i+1,jk} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1}) + u_{i+1,j+1,k-1} \\
& + u_{i-1,j+1,k-1} + u_{i+1,j-1,k-1} + u_{i-1,j-1,k-1} + u_{i+1,j+1,k+1} + u_{i-1,j+1,k+1} \\
& + u_{i+1,j-1,k+1} + u_{i-1,j-1,k+1} - 56u_{ijk}\Big],
\end{aligned}
$$

which has a computational advantage. This scheme was studied in [20] as the 14-point averaging operator. Conversely, we can observe that the error is minimum when $\beta = 1/3$, which implies $\gamma = 0$. If we take $\alpha = 2/3$, $\beta = 1/3$, and $\gamma = 0$, then (19) becomes

$$
\begin{aligned}
(\Delta_{\frac{1}{3}} u)_{ijk} \;=\; & \frac{1}{7h^2}\Big[2(u_{i+1,jk} + u_{i-1,jk} + u_{i,j+1,k} + u_{i,j-1,k} + u_{ij,k+1} + u_{ij,k-1}) + u_{i+1,j,k-1} \\
& + u_{i-1,j,k-1} + u_{i,j+1,k-1} + u_{i,j-1,k-1} + u_{i+1,j+1,k} + u_{i-1,j+1,k} + u_{i+1,j-1,k} \\
& + u_{i-1,j-1,k} + u_{i+1,j,k+1} + u_{i-1,j,k+1} + u_{i,j+1,k+1} + u_{i,j-1,k+1} - 24u_{ijk}\Big],
\end{aligned}
$$

which is a 19-point isotropic stencil for the 3D discrete Laplacian operator. Hence, Δ_0 using 15-point is isotropic and has a computational advantage, and $\Delta_{\frac{1}{3}}$ using 19-point is more accurate than Δ_0 and is an isotropic Laplacian operator that is more efficient than using 27-point.

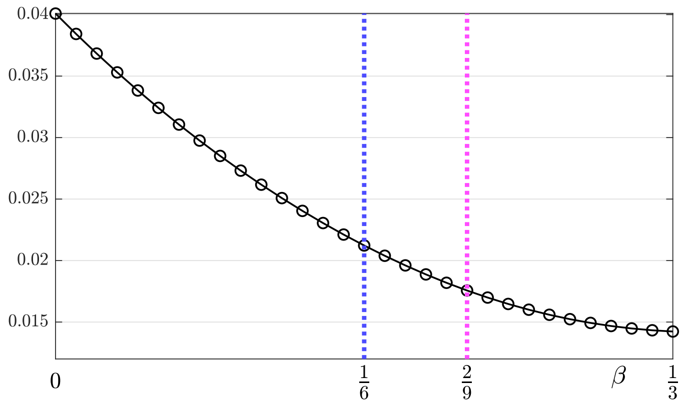


Figure 12. Discrete l_2-norm of the difference between the errors and best fitting function against β.

In this study, we focused on the numerical analysis and computational tests. Theoretical results for the discrete isotropic Laplacian operator were given in [10, 12]. Fig.13, reprinted from [25], shows the numerical results of a tumor growth model in 3D space, which is an application of real-life model problem. In Fig.13, (a) is an ellipsoid initial condition. (b) and (c) are the computational results from the anisotropic and isotropic schemes, respectively, at latter time. (e) is $\pi/4$ rotated ellipsoid initial condition. (f) and (g) are the computational results from the anisotropic and isotropic schemes, respectively, at the later time. (d) and (h) are the overlapped isosurfaces of the results ((b) and (f); (c) and (g)) for the anisotropic and isotropic schemes, respectively, after being rotated.

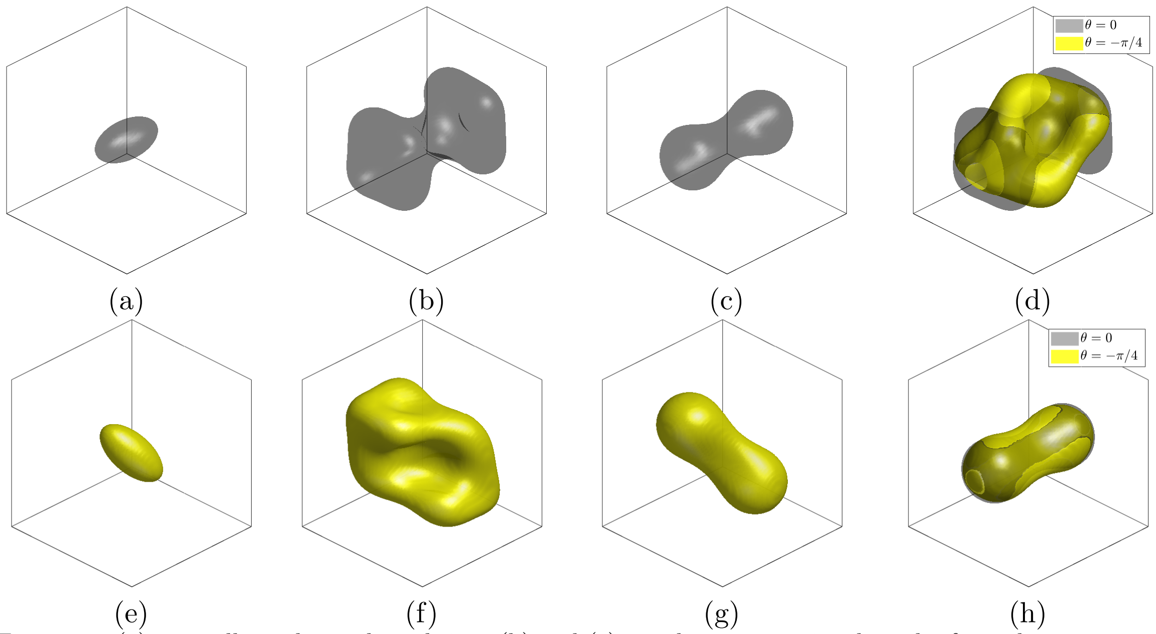


Figure 13. (a) is an ellipsoid initial condition. (b) and (c) are the computational results from the anisotropic and isotropic schemes, respectively, at later time. (e) is 45 ° rotated ellipsoid initial condition. (f) and (g) are the computational results from the anisotropic and isotropic schemes, respectively, at the later time. (d) and (h) are the overlapped isosurfaces of the results ((b) and (f); (c) and (g)) for the anisotropic and isotropic schemes, respectively, after being rotated. Reprinted from [25] with permission from Hindawi.

4. Conclusion

The main conclusion from the numerical experiments is that it is highly recommended to use isotropic stencils for the 2D and 3D Laplacian operators with the extra cost for extended stencils. In this study, we proposed benchmark functions to quantitatively evaluate the isotropy of the discrete Laplacian operators in 2D and 3D spaces using a cubic spline interpolant. There is only one isotropic 2D stencil to the lowest order, while there are many isotropic 3D stencils. Among them, there is the 3D 19-point isotropic stencil which is accurate and fast because of the small number of points compared with full 27-point isotropic stencils. There is the smallest 3D 15-point isotropic stencil which is the fastest because of the smallest number of points with a slightly larger error. As a future work, it would be interesting to develop isotropic 2D Laplacian discrete operators in triangular or hexagonal grids [1–3, 5]. Because, in triangular or hexagonal grids, we may use fewer grid points to solve the isotropic Laplacian operator, and better computational efficiency can be expected.

5. Acknowledgment

H.G. Lee was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1A2C1011708). The corresponding author (J.S. Kim) was supported by the National Research Foundation (NRF), Korea, under project BK21 FOUR. The authors are grateful to the reviewers whose valuable suggestions and comments significantly improved the quality of this paper.

References

[1] Buranay, S.C., Arshad, N. Hexagonal grid approximation of the solution of the heat equation on special polygons, *Adv. Differ. Equ.*, V.2020, N.1, 2020, pp.1-24.

[2] Buranay, S.C., Arshad, N., Matan, A.H. Hexagonal grid computation of the derivatives of the solution to the heat equation by using fourth-order accurate two-stage implicit methods, *Fractal Fract.*, V.5, N.4, 2021, pp.1-34.

[3] Buranay, S.C., Matan, A.H., Arshad, N. Two stage implicit method on hexagonal grids for approximating the first derivatives of the solution to the heat equation, *Fractal Fract.*, V.5, N.1, 2021, pp.1-26.

[4] Chamarthi, A.S., Nishikawa, H., Komurasaki, K. First order hyperbolic approach for anisotropic diffusion equation, *J. Comput. Phys.*, V.396, N.1, 2019, pp.243-263.

[5] Dosiyev, A.A., Celiker, E. Approximation on the hexagonal grid of the Dirichlet problem for Laplace's equation, *Bound. Value Probl.*, V.2014, N.73, 2014, pp.1-19.

[6] Fila, M., Macková, P., Takahashi, J., Yanagida, E. Anisotropic and isotropic persistent singularities of solutions of the fast diffusion equation, *Differ. Integral Equ.*, V., 35 N.11/12, 2022, pp.729-748.

[7] Fukuchi, T. Higher order difference numerical analyses of a 2D Poisson equation by the interpolation finite difference method and calculation error evaluation, *AIP Adv.*, V.10, N.12, 2020, pp.1-28.

[8] Gavete, L., Ureña, F., Benito, J.J., García, A., Ureña, M., Salete E. Solving second order non-linear elliptic partial differential equations using generalized finite difference method, *J. Comput. Appl. Math.*, V.318, 2017, pp.378-387.

[9] Jabbari, M., McDonough, J., Mitsoulis, E, Hattel, J.H. Application of a projection method for simulating flow of a shear-thinning fluid, *Fluids*, V.4, N.3, 2019, pp.1-16.

[10] Ji, K., Tabrizi, A.M., Karma, A. Isotropic finite-difference approximations for phase-field simulations of polycrystalline alloy solidification, *J. Comput. Phys.*, V.457, N.15, 2022, pp.1-24

[11] Kamgar-Parsi, B., Rosenfeld, A. Optimally isotropic Laplacian operator, *IEEE Trans. Image Process.*, V.8, N.10, 1999, pp.1467-1472.

[12] Kantorovich, L.V., Krylov, V.I. *Approximate Methods of Higher Analysis*, New York, Interscience, 1958, 681 p.

[13] Khankishiyev, Z.F. Solution of one problem for a loaded differential equation by the method of finite differences, *Appl. Comput. Math.*, V.21, N.2, 2022, pp.147-157.

[14] Kumar, A. Isotropic finite-differences, *J. Comput. Phys.*, V.201, N.1, 2004, pp.109-118.

[15] Lee, S., Shin, J. Energy stable compact scheme for Cahn–Hilliard equation with periodic boundary condition, *Comput. Math. Appl.*, V.77, N.1, 2019, pp.189-198.

[16] Long, J., Luo, C., Yu, Q., Li, Y. An unconditional stable compact fourth-order finite difference scheme for three dimensional Allen–Cahn equation, *Comput. Math. Appl.*, V.77, N.4, 2019, pp.1042-1054.

[17] López, C.P. *Optimization techniques via the optimization toolbox*, Berkeley, CA, 2014, 85-108 p.

[18] Pezzutti, A.D., Vega, D.A., Villar, M.A. Dynamics of dislocations in a two-dimensional block copolymer system with hexagonal symmetry, *Philos. Trans. R. Soc. A-Math. Phys. Eng. Sci.*, V.369, N.1935, 2011, pp.335-350.

[19] Shallal, M.A., Taqi, A.H. Jabbar, H.N., Rezazadeh, H., Jumaa, B.F., Korkmaz, A., Bekir, A. A numerical technique of the time fractional gas dynamics equation using finite element approach with cubic Hermit element, *Appl. Comput. Math.*, V.21, N.3, 2022, pp.269-278.

[20] Volkov, E.A. Application of a 14-point averaging operator in the grid method, *Comput. Math. Math. Phys.*, V.50 N.12, 2010, pp.2023-2032.

[21] Wang, Y., Xiao, X., Feng, X. An accurate and parallel method with post-processing boundedness control for solving the anisotropic phase-field dendritic crystal growth model, *Commun. Nonlinear Sci. Numer. Simul.*, V.115, 2022, pp.106717.

[22] Weaver, A.T., Mirouze, I. On the diffusion equation and its application to isotropic and anisotropic correlation modelling in variational assimilation, *Q. J. R. Meteorol. Soc.*, V.139, N.670, 2013, pp.242-260.

[23] Xu, S., Liu, Y. 3D acoustic wave modeling with a time-space-domain temporal high-order finite-difference scheme, *J. Geophys. Eng.*, V.15, N.5, 2018, pp.1963-1976.

[24] Xu, Y., Yuan, J. Anisotropic diffusion equation with a new diffusion coefficient for image denoising, *Pattern Anal. Appl.*, V.20, N.2, 2017, pp.579-586.

[25] Yoon, S., Lee, H.G., Li, Y., Lee, C., Park, J., Kim, S., Kim, J. Benchmark problems for the numerical discretization of the Cahn–Hilliard equation with a source term, *Discrete Dyn. Nat. Soc.*, V.2021, 2021, pp.1-11.

Hyun Geun Lee - received her Ph.D. degree in Applied Mathematics from Korea University in 2012. Currently, she is an Associate Professor of Mathematics at Kwangwoon University. Her research interests are in developing accurate and efficient numerical methods.

Seokjun Ham - is a graduate student pursuing a M.Sc. degree at Department of Mathematics, Korea University, Seoul, South Korea. He received B.Sc. degree and M.Sc. degree in Applied Mathematics from the Department of Mathematics, Kangwon National University, Chuncheon, South Korea, in 2021. His research interests include numerical analysis and scientific computing.

Junseok Kim - received his Ph.D. in Applied Mathematics from the University of Minnesota, USA in 2002. He also received his B.Sc. degree from the Department of Mathematics Education, Korea University in 1995. He joined the faculty of Korea University, in 2008, where he is currently a full Professor at the Department of Mathematics. His research interests are in computational finance and computational fluid dynamic.