Journal of Computational Science 56 (2021) 101491

Contents lists available at ScienceDirect

Journal of Computational Science

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

A conservative and stable explicit finite difference scheme for the diffusion equation

Junxiang Yang [a], Chaeyoung Lee [b], Soobin Kwak [b], Yongho Choi [c], Junseok Kim [b,*]

[a] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510275, China
[b] Department of Mathematics, Korea University, Seoul, 02841, Republic of Korea
[c] Department of Mathematics and Big Data, Daegu University, Gyeongsan-si, Gyeongsangbuk-do 38453, Republic of Korea

ARTICLE INFO

Keywords:
Saul'yev-type
Finite difference method
Alternating direction explicit scheme

ABSTRACT

In this study, we present a conservative and stable explicit finite difference scheme for the heat equation. We use Saul'yev-type finite difference scheme and propose a conservative weighted correction step to make the scheme conservative. We can practically use about 100 times larger time step than the fully Euler-type explicit scheme. Computational results demonstrate that the proposed scheme has stable and good conservative properties.

1. Introduction

In this manuscript, we propose a numerical method which is conservative and stable alternating direction explicit (ADE) scheme to solve a diffusion equation. The diffusion equation is

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = \Delta u(\mathbf{x},t), \quad \mathbf{x} \in \Omega, \ t > 0, \tag{1}$$
$$\mathbf{n} \cdot \nabla u(\mathbf{x},t) = 0 \text{ on } \partial\Omega,$$

where $\Omega \subset \mathbb{R}^d$ $(d = 1, 2, 3)$ is a domain and $u(\mathbf{x}, t)$ is the concentration of the substance at point $\mathbf{x}$ and time t. Here, $\mathbf{n}$ is the outer unit normal vector to $\partial\Omega$. There are various ways to solve Eq. (1) using numerical methods such as explicit finite difference method (FDM) [1], implicit FDM, Crank–Nicolson method, FDM [2], lattice Boltzmann method [3] and so on. The explicit FDM can be easily implemented, however, this method may not be stable and depends on the time step size for stability. To eliminate the restriction of time step size, Barakat and Clark [4] proposed an ADE scheme using forward and reverse sweeps, which is an unconditionally stable method. Because the ADE scheme maintains an explicitness and can use a large time step, the scheme has been applied to many fields such as heat flux [5–7], time-dependent problems [8,9], phase change problems [10,11], space-dependent heat source [12], fluid flow and deformation in deformable porous media [13], and high order unconditionally stable ADE scheme [13,14].

A fully implicit method is also used to solve the stability problem that occurs when a fully explicit scheme is used, however, it has the disadvantage of taking a lot of computation time. In [15], an unconditionally stable explicit Saul'yev scheme is applied to the heat equation to reduce computation time and secure stability. Karahan [16] solved the advection–diffusion equation using the Saul'yev scheme in spreadsheets. It showed the accuracy and stability according to the parameters for the proposed model. Pochai [17] proposed a new fourth-order scheme using the unconditionally stable Saul'yev method near boundary conditions to solve the hydrodynamic model. The newly proposed fourth-order scheme guarantees accuracy without significant loss of computational efficiency and improves prediction accuracy compared to conventional methods.

Since the Saul'yev scheme is unconditionally stable explicit [18,19], many recent studies have solved the advection–diffusion equation using the Saul'yev scheme. In [20], it was solved by applying a dimensionless mathematical model of salinity measurement. The actual salinity of the Chaophraya River in Thailand was measured and compared with the approximate value obtained using the model. In [21], the simple-advection–diffusion–reaction equation was solved using the Saul'yev method. It is also suitable for use in real-world problems because its implementation is simple and shows reasonable approximations. Ginzburg [22] proposes a simple and uniform modification of the local mass-conservation solvability condition to preserve mass when using the ADE scheme.

Although various studies related to the ADE scheme have been performed because it is both simple and stable, in general, the ADE scheme does not conserve a mass, which is a disadvantage. Therefore, we propose a conservative and stable ADE scheme for the diffusion equation. By introducing a weighted correction step, the proposed

* 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.jocs.2021.101491
Received 28 April 2021; Received in revised form 6 October 2021; Accepted 27 October 2021
Available online 19 November 2021

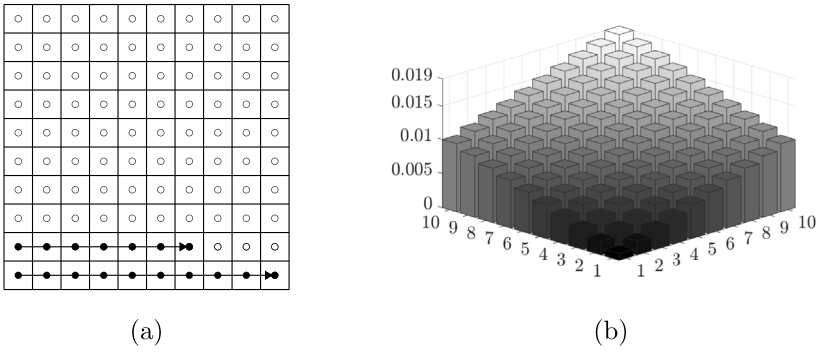


Fig. 1. Schematic of weighting matrix.

scheme overcomes the disadvantage that the ADE scheme is not able to guarantee mass conservation. The outline of this article is organized as follows. We state the proposed numerical solution algorithm in Section 2. We perform various numerical experiments to verify the accuracy, convergence, and basic properties of the diffusion equation in Section 3. In Section 4, we give conclusions.

2. Numerical solution algorithm

We describe the numerical solution algorithm for the diffusion equation in a two-dimensional (2D) space $\Omega = (x_l, x_r) \times (y_l, y_r)$:

$$\frac{\partial u(x,y,t)}{\partial t} = \Delta u(x,y,t). \tag{2}$$

Let $\Omega_h = \{x_i = x_l + (i-0.5)h,\ y_j = y_l + (j-0.5)h |\ 1 \le i \le N_x,\ 1 \le j \le N_y\}$ be the discrete computational domain, where $h = (x_r - x_l)/N_x = (y_r - y_l)/N_y$ is the uniform step size; N_x and N_y are the numbers of the grid points in the x- and y-direction, respectively. Let u_{ij}^n be the numerical approximations of $u(x_i, y_j, n\Delta t)$, where Δt is the time step. We use the following ADE scheme [14] for the diffusion Eq. (2):

$$\text{For } j = 1, 2, \dots, N_y, \quad \text{for } i = 1, 2, \dots, N_x, \tag{3}$$

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

Eq. (4) can be simplified as follows:

$$u_{ij}^* = \frac{r u_{i-1,j}^* + r u_{i+1,j}^n + (1-2r) u_{ij}^n + r u_{i,j-1}^* + r u_{i,j+1}^n}{1+2r}, \tag{5}$$

where $r = \Delta t/h^2$. Note that we use a nested loop in Eq. (3). For each outer loop j, the inner loop i runs from 1 to N_x, see Fig. 1(a).

Here, we use the following Neumann boundary condition:

$$\begin{aligned} &u_{0j}^* = u_{1j}^*,\ u_{N_x+1,j}^* = u_{N_x,j}^n,\ \text{for } j = 1, \dots, N_y, \\ &u_{i0}^* = u_{i1}^*,\ u_{i,N_y+1}^* = u_{i,N_y}^n,\ \text{for } i = 1, \dots, N_x. \end{aligned} \tag{6}$$

In general, $\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u_{ij}^* \neq \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u_{ij}^0$, i.e., it is not conservative. The global truncation error is accumulative, therefore, we propose the following weighted correction step:

$$u_{ij}^{n+1} = u_{ij}^* - w_{ij} \sum_{p=1}^{N_x}\sum_{q=1}^{N_y} (u_{pq}^* - u_{pq}^0), \tag{7}$$

where a weight w_{ij} is defined as

$$w_{ij} = \frac{2(i+j-1)}{N_x N_y (N_x + N_y)}. \tag{8}$$

Thereby, we can compensate the problem that the farther away from the starting points of each loop, the larger the errors are. Fig. 1(b) gives a schematic illustration of the weighting matrix. Subsequently, we have

$$\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u_{ij}^{n+1} = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} \left[u_{ij}^* - w_{ij} \sum_{p=1}^{N_x}\sum_{q=1}^{N_y} (u_{pq}^* - u_{pq}^0) \right]$$

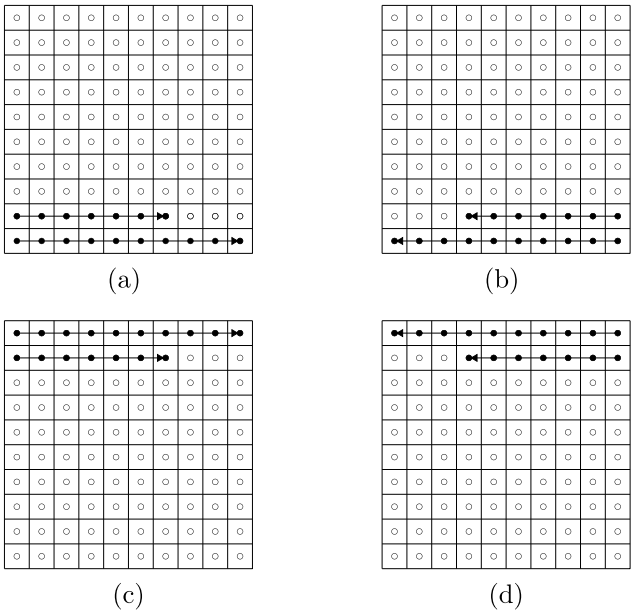


Fig. 2. (a)–(d) are schematic diagrams for Eqs. (3), (9)–(11), respectively.

$$= \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u_{ij}^* - \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} w_{ij} \sum_{p=1}^{N_x}\sum_{q=1}^{N_y} (u_{pq}^* - u_{pq}^0)$$

$$= \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u_{ij}^* - \sum_{p=1}^{N_x}\sum_{q=1}^{N_y} u_{pq}^* + \sum_{p=1}^{N_x}\sum_{q=1}^{N_y} u_{pq}^0 = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u_{ij}^0$$

because $\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} w_{ij} = 1$. That is, the property of mass conservation holds by using the weighted correction step. In a 2D space, we use 4 cases of nested loops including Eq. (3); and the other 3 cases are

$$\text{For } j = 1, 2, \dots, N_y, \quad \text{for } i = N_x, N_x - 1, \dots, 1, \tag{9}$$

$$\text{For } j = N_y, N_y - 1, \dots, 1, \quad \text{for } i = 1, 2, \dots, N_x, \tag{10}$$

$$\text{For } j = N_y, N_y - 1, \dots, 1, \quad \text{for } i = N_x, N_x - 1, \dots, 1. \tag{11}$$

Fig. 2 illustrates the 4 cases of nested loops. We sequentially choose one of four iterative orders for each temporal update.

To perform a comparison study for the mass-conservative property of our proposed method to other methods, let us consider one of the modern methods widely used in practice: there is a multigrid method that is an iterative method for solving partial differential equations [23]. The multigrid method is highly efficient, however, for the iterative solvers, it is expected that the accuracy is in the range of the discretization accuracy because of a stopping criterion of the iterative [24]. Each iteration of the multigrid algorithm is ended when

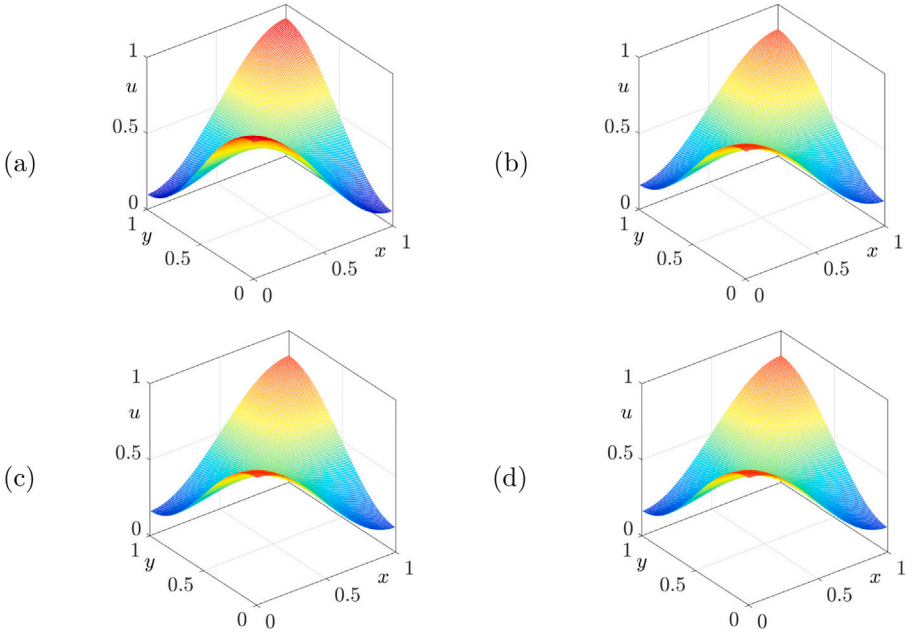


Fig. 3. Numerical results with respect to $\Delta t = 25$ h^2, 2.5 h^2, and 0.25 h^2 at $t = 0.02$ are shown in (a)–(c), respectively. The exact solution at $t = 0.02$ is shown in (d).

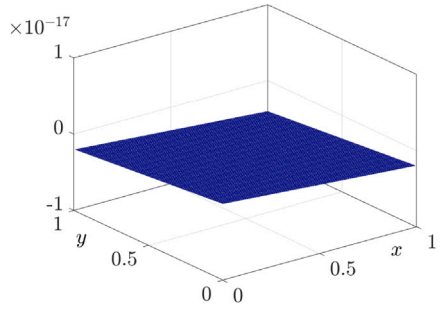


Fig. 4. Correction mass at the first iteration when $\Delta t = 0.25$ h^2.

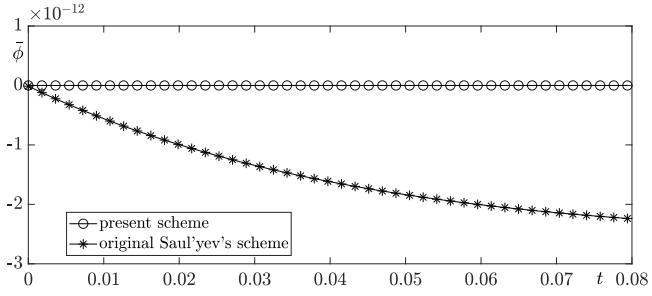


Fig. 5. Temporal evolutions of average concentrations computed by our proposed method and the original Saul'yev's method.

the residual is reduced by a certain value (that is, tolerance) starting with an initial guess [25,26], therefore, loss as much as the tolerance continues to accumulate.

Now, we estimate and compare the computational complexity of both the two methods. The proposed method is the Gauss–Seidel type method, which is commonly used for the relaxation step in the multigrid method. Thus, for the convenience of description, the cost of performing one relaxation sweep is defined as a work unit (WU) [27]. We assume that each number of pre-smoothing and post-smoothing relaxation sweeps is equal to the positive integer ν. In a 2D domain, the computational complexity of the proposed method is estimated to be less than 2 WU because the other total operations are less than one WU. In the case of the multigrid method, the computation cost of a V-cycle with one relaxation sweep on is

$$2\nu\left(1+\frac{1}{2^2}+\frac{1}{2^4}+\cdots+\frac{1}{2^{2N}}\right) < \frac{2\nu}{1-2^{-2}}\text{WU} = \frac{8}{3}\nu \text{ WU}.$$

Generally, ν is larger than 1, and although it depends on the tolerance value, one V-cycle is usually not enough [28,29]. Therefore, the computational complexity of our proposed method is estimated to be substantially lower than that of the multigrid method.

We show that the proposed scheme is unconditionally stable using von Neumann stability analysis. Substituting $u^*_{jk} = \xi^* e^{i(\alpha j+\beta k)h}$ and $u^n_{jk} = \xi^n e^{i(\alpha j+\beta k)h}$ into Eq. (5) yields

$$\begin{aligned}\xi^* e^{i(\alpha j+\beta k)h} = \Big(&r\xi^* e^{i(\alpha(j-1)+\beta k)h} + r\xi^n e^{i(\alpha(j+1)+\beta k)h} + (1-2r)\xi^n e^{i(\alpha j+\beta k)h} \\ &+ r\xi^* e^{i(\alpha j+\beta(k-1))h} + r\xi^n e^{i(\alpha j+\beta(k+1))h}\Big)/(1+2r),\end{aligned} \tag{12}$$

where $i = \sqrt{-1}$ and ξ, α, β are real parameters. Eq. (12) can be written as follows:

$$\begin{aligned}\left|\frac{\xi^*}{\xi^n}\right| &= \left|\frac{1-2r+r(e^{i\alpha h}+e^{i\beta h})}{1+2r-r(e^{-i\alpha h}+e^{-i\beta h})}\right| \\ &= \left|\frac{1-2r+r\left(\cos(\alpha h)+i\sin(\alpha h)+\cos(\beta h)+i\sin(\beta h)\right)}{1+2r-r\left(\cos(\alpha h)-i\sin(\alpha h)+\cos(\beta h)-i\sin(\beta h)\right)}\right|.\end{aligned} \tag{13}$$

To show the unconditional stability of the scheme, we need to show that $|\xi^*/\xi^n| \le 1$ for any r, α, β values. If we square both sides of Eq. (13), then we have

$$\frac{|\xi^*|^2}{|\xi^n|^2} = \frac{(1-2r+r(\cos(\alpha h)+\cos(\beta h)))^2 + r^2(\sin(\alpha h)+\sin(\beta h))^2}{(1+2r-r(\cos(\alpha h)+\cos(\beta h)))^2 + r^2(\sin(\alpha h)+\sin(\beta h))^2}. \tag{14}$$

If we subtract the denominator from the numerator in Eq. (14), we obtain $-8r + 4r(\cos(\alpha h)+\cos(\beta h)) \le 0$, which implies $|\xi^*/\xi^n| \le 1$ for any r, α, β values. The additional mass correction step updates the numerical solution by very small magnitude and has negligible effect in stability of the scheme. The other three loops can be similarly derived. Therefore, the proposed scheme is unconditionally stable.

3. Numerical experiments

In this section, various numerical tests are performed to validate our proposed method. Without specific needs, all simulations are conducted

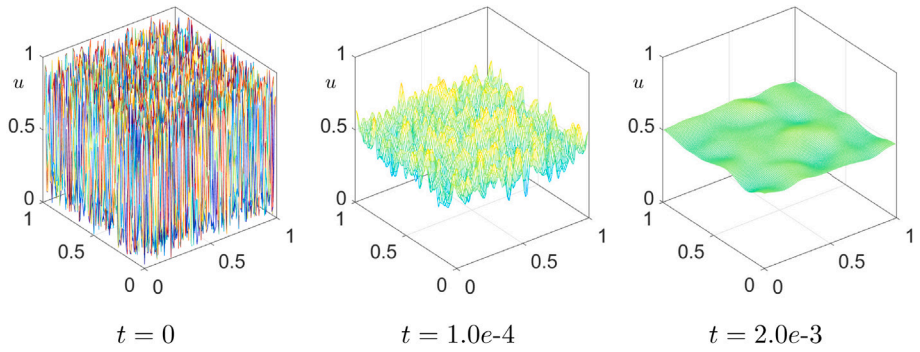


Fig. 6. Snapshots of the numerical results with the random initial condition. The computational moments are shown under each figure.

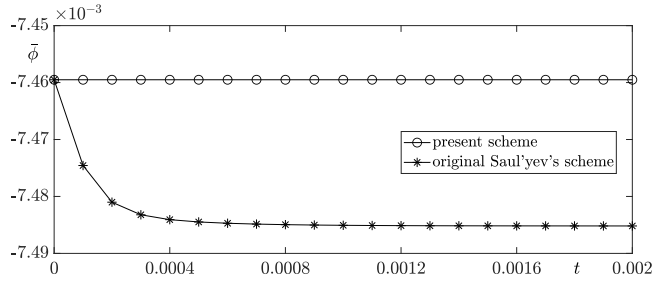


Fig. 7. With a random initial condition, temporal evolutions of average concentrations computed by our proposed method and the original Saul'yev's method.

in the domain $\Omega = (0,1)\times(0,1)$ with the initial condition

$$u(x,y,0) = 0.5 + 0.5\cos(\pi x)\cos(\pi y). \tag{15}$$

Because we use the initial condition (15) and the homogeneous Neumann boundary condition (6), the exact solution is given to be

$$u_{exact}(x,y,t) = 0.5 + 0.5\cos(\pi x)\cos(\pi y)\mathrm{e}^{-2\pi^2 t}.$$

3.1. Comparison with exact solution

First of all, we verify the consistence between the numerical solution and the exact solution. Here, $h = 0.01$ and three different time steps $\Delta t = 0.25h^2,\ 2.5h^2$, and $25h^2$ are used. All simulations are performed until $t = 0.02$. The pointwise error is defined to be $e_{ij}^n = u_{ij}^n - u_{exact}(x_i, y_j, n\Delta t)$. We define the discrete L^2 norm and maximum norm as follows

$$\|e\|_{L^2}^d = \sqrt{h^2\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}(e_{ij}^n)^2}, \qquad \|e\|_{\max}^d = \max_{1\le i\le N_x, 1\le j\le N_y}|e_{ij}^n|. \tag{16}$$

Table 1 lists the discrete L^2 norms and maximum norms at $t = 0.02$. Fig. 3(a)–(c) display the numerical results at $t = 0.02$ with respect to $\Delta t = 25h^2,\ 2.5h^2$, and $0.25h^2$, respectively, and Fig. 3(d) shows the exact profile at $t = 0.02$. For most numerical methods, the increase of time step is accompanied by the loss of accuracy. By comparing Fig. 3(a) ($\Delta t = 25h^2$) and (d) (exact solution), we can find that the difference is obvious. However, it can be observed that the numerical result converges to the exact solution as the time step is refined. Although the proposed scheme can use a relatively large time step to perform the simulation, a small enough time step is still needed when we require computational accuracy.

We examine the correction mass at each iteration. Fig. 4 shows the amount of correction mass in one iteration when $\Delta t = 0.25h^2$ is used. This seems like a very small amount, however, it is highly significant when the ADE scheme is employed in long-time simulations under the assumption of mass conservation, for example.

Table 1
L^2 and maximum norms of numerical error with respect to different time steps.

Time step Δt:	25 h^2	2.5 h^2	0.25 h^2
L^2 norm:	3.36e−2	7.89e−4	1.35e−5
Maximum norm:	7.12e−2	1.60e−3	2.77e−5

Table 2
Temporal L^2 errors and convergence rates at $t = 0.0016$. Here, $h = 0.01$ is fixed.

Time step Δt:	0.5 h^2		0.25 h^2		0.125 h^2
L^2 error:	4.32e−6		1.55e−6		8.60e−7
Rate:		1.48		0.85	

Table 3
Spatial L^2 errors and convergence rates at $t = 0.0016$. Here, $\Delta t = 1.25$e−5 is fixed.

Mesh size h:	0.1		0.05		0.025
L^2 error:	6.27e−5		1.57e−5		3.97e−6
Rate:		2.00		1.98	

3.2. Convergence tests

To test the temporal accuracy, we fix the mesh size $h = 0.01$ and use different time steps: $\Delta t = 0.5h^2,\ 0.25h^2$, and $0.125h^2$ to perform the computation until $t = 0.0016$. The temporal convergence rate of two successive errors is defined as $\log_2(\|e^{\Delta t}\|_{L^2}^d / \|e^{\frac{\Delta t}{2}}\|_{L^2}^d)$. Table 2 shows the L^2 errors and convergence rates at $t = 0.0016$, we can observe that temporal accuracy of our proposed scheme is between first order and second order. Next, we investigate the spatial accuracy of our method. Here, $\Delta t = 1.25$e−5 is fixed and three different mesh sizes $h = 0.1,\ 0.05$, and 0.025 are considered. The spatial convergence rate of two successive errors is defined as $\log_2(\|e^h\|_{L^2}^d / \|e^{\frac{h}{2}}\|_{L^2}^d)$. The results at $t = 0.0016$ are listed in Table 3. As we can see, our proposed scheme has second-order spatial accuracy.

3.3. Mass conservation

In this subsection, we validate the mass conservation of our proposed method. We define the discrete average concentration to be $\phi = h^2\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u_{ij}$. We use $h = 0.01$ and $\Delta t = 0.25h^2$ in the simulations. In Fig. 5, we plot the temporal evolutions of average concentrations obtained by our proposed scheme and the original Saul'yev's method [19] (without the correction step, Eq. (7)). For a simple representation of Fig. 5, we define $\bar{\phi} = \phi - 0.5$. As we can see, our proposed method obviously preserves the average concentration compared to the original Saul'yev's method.

Now, we practically demonstrate the mass conservation of our proposed method through two non-trivial examples: one is using a random initial condition and the other is an advection flow. First, we consider a random initial condition $u(x,y,0) = \mathrm{rand(x,y)}$, where $\mathrm{rand(x,y)}$ is a

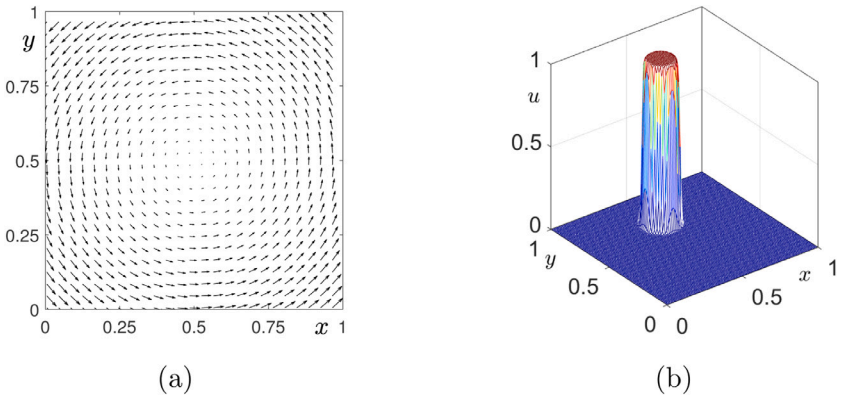


Fig. 8. Background velocity field (a) and initial state of u (b).

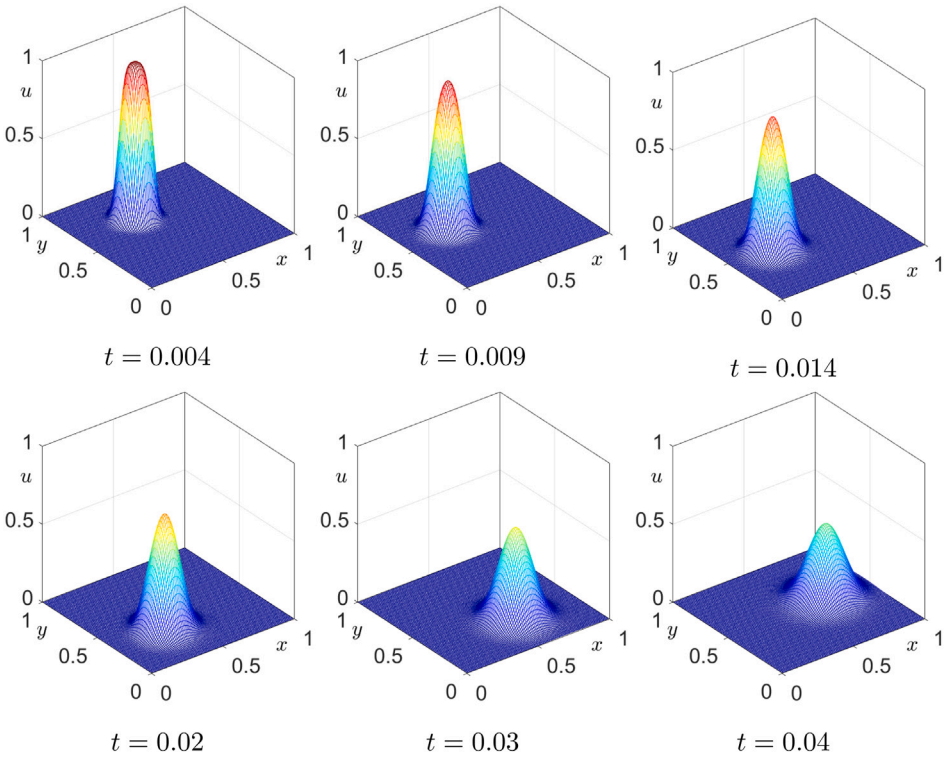


Fig. 9. Snapshots of the diffusion with advection. The computational moments are shown under each figure.

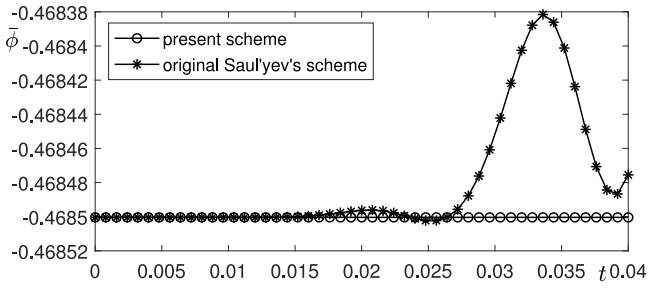


Fig. 10. Temporal evolutions of average concentrations for the diffusion with background flow.

uniformly distributed random numbers between 0 and 1, in this test. We use the same h and Δt with the above test. In Fig. 6, there are snapshots with time when the random initial condition is employed. Fig. 7 shows that the proposed method preserves the average concentration with time in the contrast to the original Saul'yev's method.

Next, we consider an example of advection flow:

$$\frac{\partial u(\mathbf{x},t)}{\partial t} + \nabla \cdot (u(\mathbf{x},t)\mathbf{U}(\mathbf{x})) = D\Delta u(\mathbf{x},t), \tag{17}$$

where $\mathbf{U}(\mathbf{x}) = (U(\mathbf{x}), V(\mathbf{x}))$ is the background velocity field, $\mathbf{x} = (x, y)$, and $D > 0$ is a constant. The velocity components are defined as

$$U(\mathbf{x}) = -120(y - 0.5), \quad V(\mathbf{x}) = 120(x - 0.5).$$

The fully discretization for the advection term in Eq. (17) is

$$\nabla_d \cdot (u\mathbf{U})^n_{ij} = \frac{(uU)^n_{i+1,j} - (uU)^n_{i-1,j}}{2h} + \frac{(uV)^n_{i,j+1} - (uV)^n_{i,j-1}}{2h}.$$

Here, we use $h = 0.01$, $D = 0.1$, $\Delta t = 0.25h^2$. The initial condition is defined as

$$u(x, y, 0) = 0.5 + 0.5 \tanh\left(\frac{0.1 - \sqrt{(x-0.5)^2 + (y-0.7)^2}}{0.004}\right). \tag{18}$$

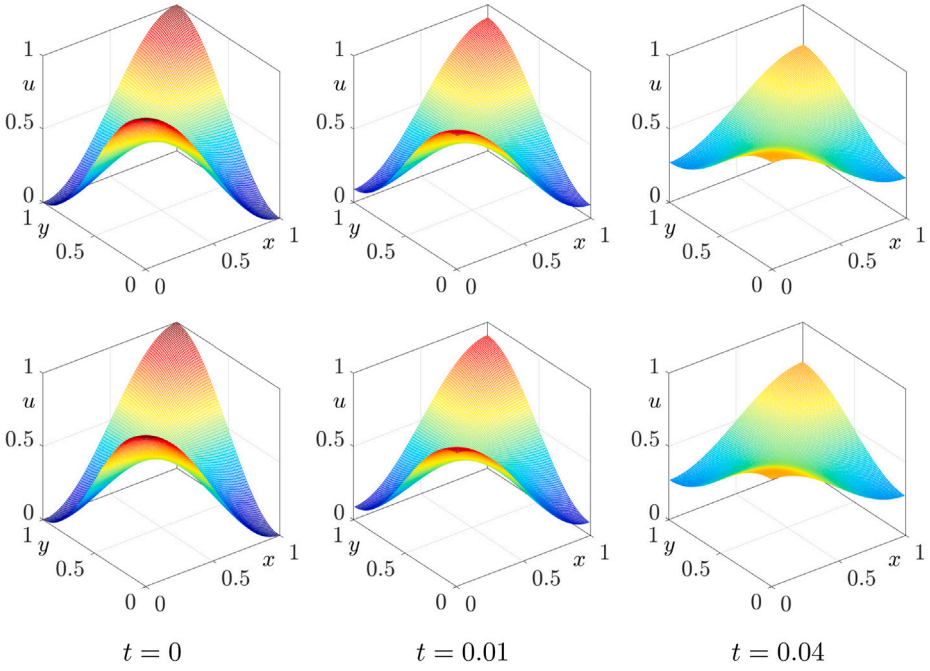


Fig. 11. Snapshots computed by the fully explicit scheme (top row) and our proposed scheme (bottom row).

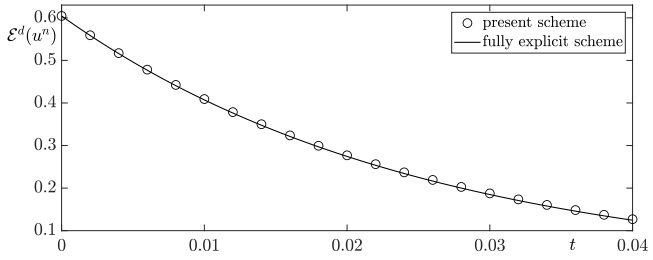


Fig. 12. Temporal evolutions of total energy obtained by our proposed scheme and the fully explicit scheme.

Fig. 8(a) and (b) display the velocity field and initial state, respectively. The snapshots at different moments are shown in Fig. 9. In Fig. 10, we also plot the temporal evolutions of average concentrations with respect to the original Saul'yev's and present methods. As the flow moves, diffusion occurs gradually and the proposed scheme preserves the concentration. On the contrary, mass conservation is not strictly satisfied when the original scheme is used.

3.4. Comparison with the fully explicit scheme

By von Neumann stability analysis [31], we know that the restriction of the time step of the fully explicit scheme for the diffusion equation is of $O(h^2)$. To show the efficiency of the proposed scheme, we investigate the possible maximum time steps $\Delta t_{\max}$ which allow the stable computation with respect to various mesh sizes. The increasingly finer mesh sizes $h = 1/25,\ 1/50,\ 1/100$, and $1/200$ are considered. All computations are performed until the numerical equilibrium state is reached, i.e., $\|u^n - u^{n-1}\|^d_{L^2} \le 10^{-5}$. Table 4 lists the values of $\Delta t_{\max}$ with respect to different mesh sizes. We can find that our proposed scheme can use approximately 100 times larger time steps.

Next, we consider the evolutions computed by our proposed scheme and the fully explicit scheme. Here, $h = 0.01$ is used. For the fully explicit scheme, we use $\Delta t = 0.25h^2$. For our proposed scheme, we use $\Delta t = 2.5h^2$. The snapshots at specific computational moments are shown in Fig. 11. Although the time step for the proposed scheme is ten times larger than the time step adopted for the fully explicit scheme, we can find that the numerical results are very similar. The energy dissipation is a typical property for the diffuse model. Here, we define the discrete total energy at nth time level to be

$$\mathcal{E}^d(u^n) = h^2 \sum_{i=1}^{N_x-1} \sum_{j=1}^{N_y-1} \frac{1}{2} \left[\frac{(u^n_{i+1,j} - u^n_{ij})^2}{h^2} + \frac{(u^n_{i,j+1} - u^n_{ij})^2}{h^2} \right].$$

Table 4
Values of maximum time step $\Delta t_{\max}$ guaranteeing stable computation.

Mesh size h:	1/25	1/50	1/100	1/200
Present scheme:	3.30e−2	1.66e−2	8.30e−3	4.00e−3
Fully explicit scheme:	4.00e−4	1.00e−4	2.50e−5	6.25e−6

Fig. 12 shows the temporal evolutions of discrete total energy with respect to the proposed scheme and fully explicit scheme. It can be observed that the energy evolutions are very similar even if a larger time step is used. On the premise of similar dynamics, the proposed scheme obviously saves the computational costs.

3.5. Energy evolutions with various time steps

The energy dissipation is a basic property of the diffuse equation. To test the discrete energy dissipation law, we use various time steps $\Delta t = 100\delta t,\ 50\delta t,\ 10\delta t$, and δt, where $\delta t = 0.25h^2$ and $h = 0.01$. The results at $t = 0.2$ obtained by various time steps are illustrated in Fig. 13. By using the proposed scheme, we find that the numerical solution does not blow up even if a relatively large time step is used. However, it is clear that the difference of solution increases when we increase the time step. With the refinement of the time step, it can be observed that the solution converges. This result indicates that the stability does not ensure accuracy. In general, a smaller time step is still necessary if one wants to obtain an accurate result. The evolutions of energy curves with respect to different time steps are plotted in Fig. 14. We find that the energy curves are non-increasing with respect to various time steps. This indicates the solutions computed by the proposed scheme satisfy the energy-dissipation law even if larger time steps are used.

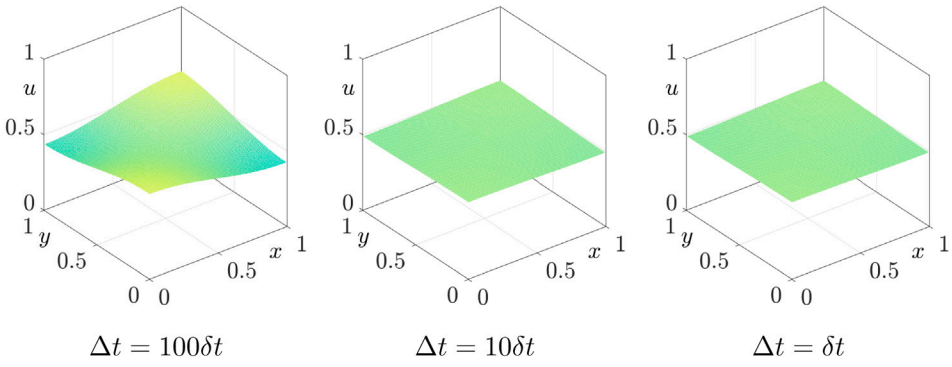


Fig. 13. Snapshots at $t = 0.2$ computed by various times steps.

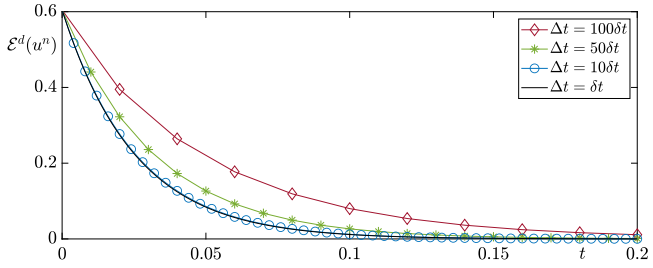


Fig. 14. Temporal evolutions of energy curves with respect to different time steps.

3.6. Evolution with a sharp initial value

We consider the evolution of diffusion equation with the following sharp initial condition

$$u(x,y,0) = \begin{cases} 1 & \text{if } 0.4 < x < 0.8 \text{ and } 0.6 < y < 0.75, \\ 1 & \text{if } 0.65 < x < 0.8 \text{ and } 0.25 < y < 0.75, \\ 1 & \text{if } 0.2 < x < 0.8 \text{ and } 0.2 < y < 0.35, \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

The computational domain is $\Omega = (0,1) \times (0,1)$. We use $h = 0.01$ and $\Delta t = 0.25h^2$. Fig. 15 shows the snapshots of evolution. It can be observed that the initially sharp state diffuses with time evolution. The numerical results indicate that our proposed scheme works well for the diffusion equation with sharp initial condition. For the diffusion equation, the diffusion is accompanied by the energy dissipation. To verify this physically meaningful phenomenon, we plot the energy curve in Fig. 16, the numerical result indicates that the energy dissipation law is satisfied.

3.7. Comparison between the alternating direction scheme and one direction scheme

The proposed method is stable ADE scheme which uses four cases of the nested loop, Eqs. (3) and (9)–(11). Now, we compare the numerical results obtained from the proposed and one direction schemes. For the test, $h = 0.01$ and $\Delta t = 0.25h^2$ are used, and the initial condition shown in Fig. 17(a) is given as

$$u(x,y) = 0.5 + 0.5 \tanh\left(\frac{0.3 - \sqrt{(x-0.5)^2 + (y-0.5)^2}}{0.02}\right). \tag{20}$$

Fig. 17(b) and (c) show the snapshots at $t = 0.02$ and $t = 0.04$ using our proposed scheme, respectively.

Fig. 18(a)–(d) show the differences between the numerical results of the proposed scheme using four directions alternatively and each direction scheme using Eqs. (3) and (9)–(11) at $t = 0.04$, respectively. From these results, we can confirm that if the numerical solutions are computed only in one direction, then the result is biased. Therefore, we should employ all directions in the proposed scheme.

3.8. Application for smoothing images

As a useful example of applying our proposed scheme, a noise smoothing method for images is introduced. We adapt an image in [30] with 5% salt-and-pepper noise as shown in Fig. 19(a). We modify the diffusion Eq. (1) with a fidelity term as follows:

$$\frac{\partial u}{\partial t} = \Delta u(\mathbf{x},t) + \lambda\left(f(\mathbf{x}) - u(\mathbf{x},t)\right), \tag{21}$$

where λ is a positive fidelity parameter, $f(\mathbf{x})$ is the input image data and the Neumann boundary condition (6) is used. We discretize Eq. (21) in the same manner as Eqs. (3) and (4) and solve the fidelity term implicitly. Then, for example,

For $j = 1, 2, \ldots, N_y$, for $i = 1, 2, \ldots, N_x$,

$$u^*_{ij} = \frac{ru^*_{i-1,j} + ru^n_{i+1,j} + (1-2r)u^n_{ij} + ru^*_{i,j-1} + ru^n_{i,j+1} + \lambda\Delta t f_{ij}}{1 + 2r + \lambda\Delta t}.$$

In this test, we use $h = 1$, $\Delta t = 0.05$, and $\lambda = 1$. Fig. 19(b) shows the computational result at time $t = 6$. It is observed that the serious noises of the input image are smoothed using the modified diffusion Eq. (21).

4. Conclusions

In this study, we presented a conservative and stable explicit finite difference scheme for the heat equation using the Saul'yev-type finite difference scheme. We proposed a conservative weighted correction step to make the scheme conservative. We conducted several numerical experiments to validate the performance of the proposed scheme. From a comparison study with the fully explicit scheme, we observed that the proposed conservative scheme can use two orders of magnitude larger time steps. The accuracy tests indicated that the present scheme achieved the first-order accuracy in time and second-order accuracy in space. To show the good conservative property, we performed the simulations with the cosine-shaped initial state, random initial state, and diffusion with background flow, the results indicated that the proposed scheme preserved the concentration better than the original Saul'yev's scheme. Moreover, the simulations with cosine-shaped and sharp initial conditions indicated that the solution dissipated the total energy even if larger time steps were adopted. The conventional Saul'yev-type finite difference scheme is simple but is not conservative. In this paper, we proposed a correction step for the scheme to be conservative. We found it is important to apply all the iterative directions; otherwise, we may have a biased result as shown in Section 3.7. In the future works, the proposed scheme will be extended to simulate various energy-dissipative and mass-conserved partial differential equations, such as the Cahn–Hilliard equation [32,33], Ohta–Kawasaki model [34], conserved Allen–Cahn equation [35], conservative convection–diffusion equation [36], phase-field crystal equation [37], and hydrodynamically coupled phase-field models [38,39], etc.

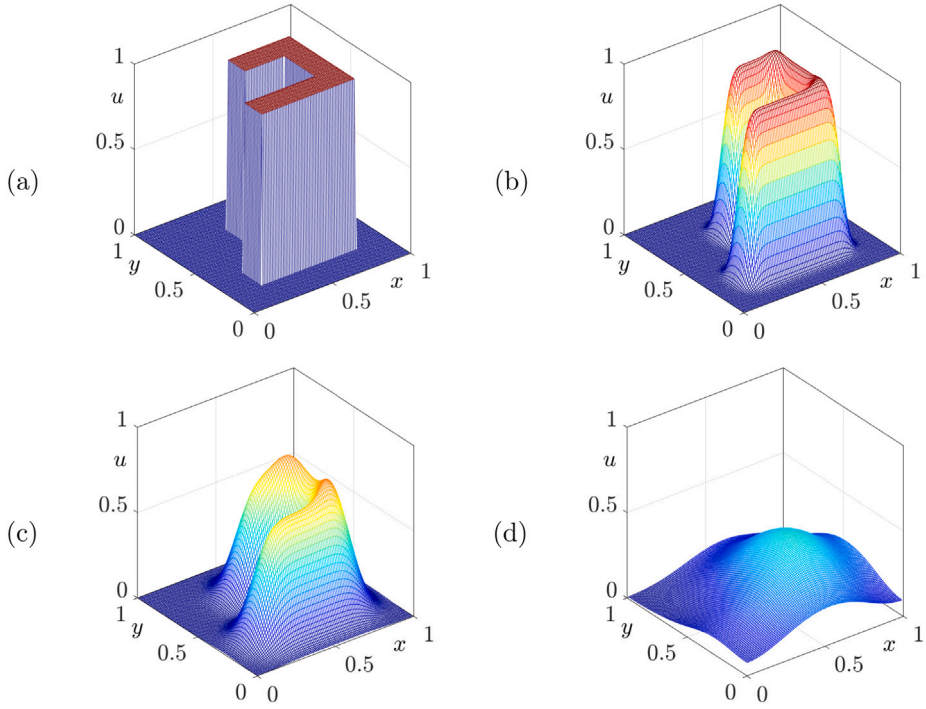


Fig. 15. (a)–(d) are the results of the diffusion equation with sharp initial condition at $t = 0$, 6.0e−4, 3.0e−3, and 2.0e−2, respectively.

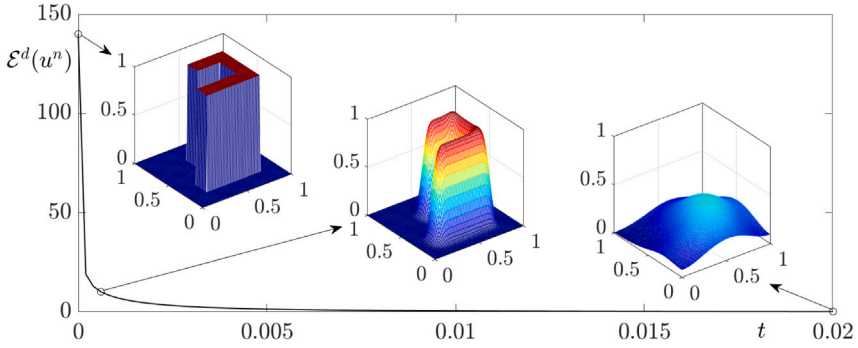


Fig. 16. Temporal evolution of energy curve. Here, the insets are the snapshots at specific moments.

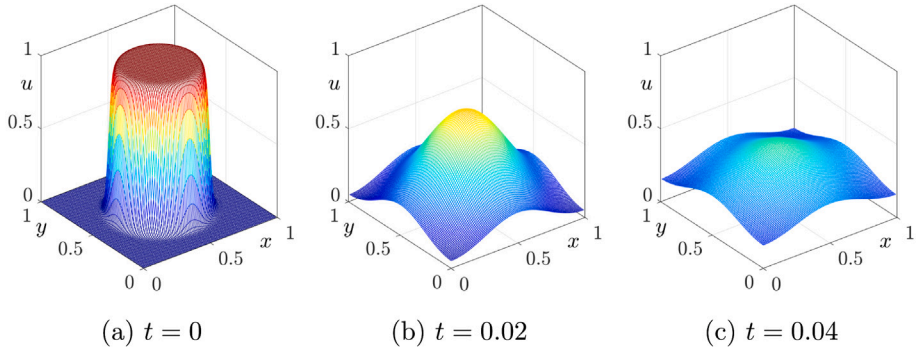


Fig. 17. (a)–(c) are snapshots with the initial condition (20) using the proposed scheme. Each time is mentioned under each figure.

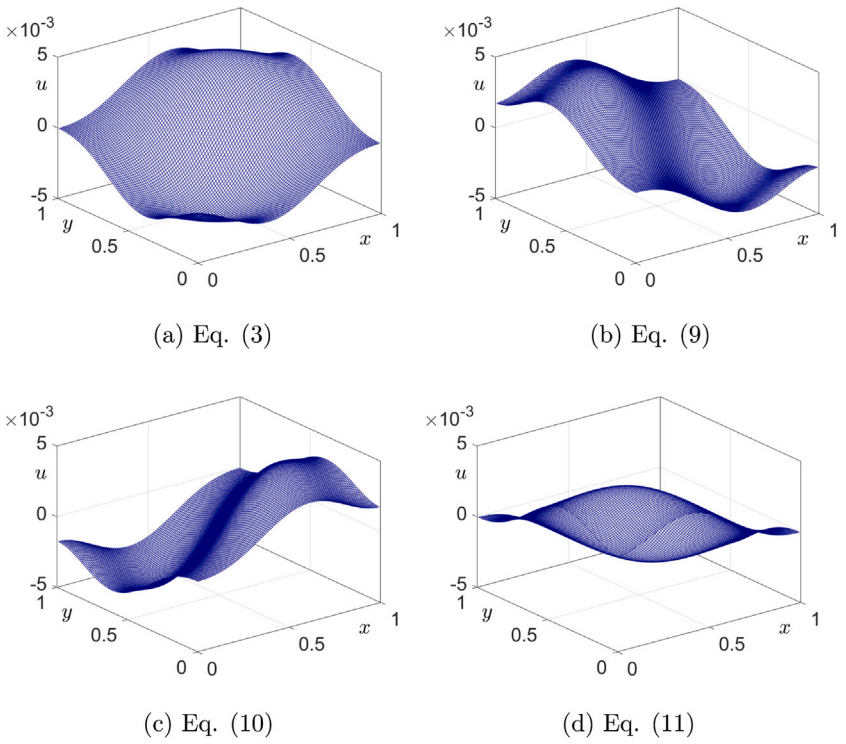


Fig. 18. (a)–(d) are differences between the numerical results of the proposed scheme and each direction scheme using Eqs. (3) and (9)–(11) at $t = 0.04$, respectively.

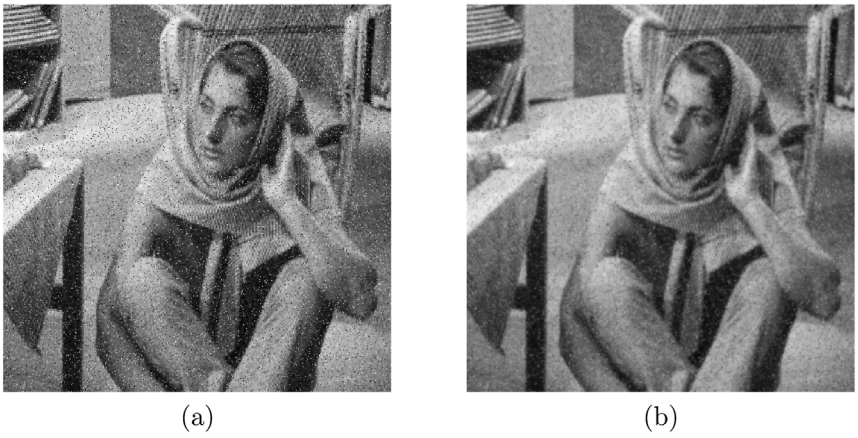


Fig. 19. (a) Input image f with salt-and-pepper noise, which is adapted from [30] with permission of Hindawi, and (b) image smoothed by the proposed scheme.

CRediT authorship contribution statement

Junxiang Yang: Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Chaeyoung Lee:** Formal analysis, Investigation, Software, Validation, Visualization, Writing – original draft, Writing – review and editing. **Soobin Kwak:** Investigation, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Yongho Choi:** Formal analysis, Funding acquisition, Investigation, Validation, Visualization, Writing – original draft, Writing – review & editing. **Junseok Kim:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

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

Acknowledgments

The author (Y. Choi) was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (NRF-2020R1C1C1A0101153712). The corresponding author (J.S. Kim) was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Korea (NRF-2019R1A2C1003053). The authors are grateful to the reviewers for the constructive and helpful comments on the revision of this article.

References

[1] H. Zhang, J. Yan, X. Qian, S. Song, Numerical analysis and applications of explicit high order maximum principle preserving integrating factor Runge–Kutta schemes for Allen–Cahn equation, Appl. Numer. Math. 161 (2021) 372–390.

[2] J. Hötzer, A. Reiter, H. Hierl, P. Steinmetz, M. Selzer, B. Nestler, The parallel multi-physics phase-field framework Pace3D, J. Comput. Sci. 26 (2018) 1–12.

[3] N. Takada, J. Matsumoto, S. Matsumoto, K. Kurihara, Phase-field model-based simulation of two-phase fluid motion on partially wetted and textured solid surface, J. Comput. Sci. 17 (2016) 315–324.

[4] H.Z. Barakat, J.A. Clark, On the solution of the diffusion equations by numerical methods, J. Heat Transfer 88 (4) (1966) 421–427.

[5] Z.Z. He, J. Liu, An efficient parallel numerical modeling of bioheat transfer in realistic tissue structure, Int. J. Heat Mass Transfer 95 (2016) 843–852.

[6] M.J. Huntul, D. Lesnic, Reconstruction of the timewise conductivity using a linear combination of heat flux measurements, J. King Saud Univ. Sci. 32 (1) (2020) 928–933.

[7] M.J. Huntul, Identification of the timewise thermal conductivity in a 2D heat equation from local heat flux conditions, Inverse Probl. Sci. Eng. 29 (7) (2020) 903–919.

[8] M.J. Huntul, Recovering the timewise reaction coefficient for a two-dimensional free boundary problem, Eurasian J. Math. Comput. Appl. 7 (4) (2019) 66–85.

[9] M.J. Huntul, D. Lesnic, Determination of a time-dependent free boundary in a two-dimensional parabolic problem, Int. J. Appl. Comput. Math. 5 (4) (2019) 1–15.

[10] T. Mauder, P. Charvat, J. Stetina, L. Klimes, Assessment of basic approaches to numerical modeling of phase change problems-accuracy, efficiency, and parallel decomposition, J. Heat Transfer 139 (8) (2017) 084502.

[11] H. Saad, H.G. Asker, An unconditionally stable finite-difference method for the solution of multi-dimensional transport equation, Ain Shams Eng. J. 12 (1) (2021) 807–820.

[12] M.S. Hussein, D. Lesnic, B.T. Johansson, A. Hazanee, Identification of a multi-dimensional space-dependent heat source from boundary data, Appl. Math. Model. 54 (2018) 202–220.

[13] S.H. Prassetyo, M. Gutierrez, Explicit high-order ADE solutions for fluid flow in the coupled Biot equations, in: Poromechanics VI, 2017, pp. 215–222.

[14] S.H. Prassetyo, M. Gutierrez, High-order ADE scheme for solving the fluid diffusion equation in non-uniform grids and its application in coupled hydro-mechanical simulation, Int. J. Numer. Anal. Methods Geomech. 42 (16) (2018) 1976–2000.

[15] J. Sun, Numerical schemes for the forward–backward heat equation, Int. J. Comput. Math. 87 (3) (2010) 552–564.

[16] H. Karahan, Unconditional stable explicit finite difference technique for the advection-diffusion equation using spreadsheets, Adv. Eng. Softw. 38 (2) (2007) 80–86.

[17] N. Pochai, Unconditional stable numerical techniques for a water-quality model in a non-uniform flow stream, Adv. Differential Equations 2017 (1) (2017) 1–13.

[18] S.A. Chin, Understanding Saul'yev-type unconditionally stable schemes from exponential splitting, Numer. Methods Partial Differential Equations 30 (6) (2014) 1961–1983.

[19] V.K. Saul'yev, Integration of Equations of Parabolic Type Equation by the Method of Nets, Pergamon Press, New York, 1964.

[20] N. Camcoon, N.Z. Pochai, A non-dimensional mathematical model of salinity measurement in the Chaophraya River using a new fourth order finite difference method with the Saulyev technique, Eng. Lett. 28 (4) (2020) 978–984.

[21] P. Samalerk, N. Pochai, A saulyev explicit scheme for an one-dimensional advection-diffusion-reaction equation in an opened uniform flow stream, Thai J. Math. 18 (2) (2020) 677–683.

[22] I. Ginzburg, Steady-state two-relaxation-time lattice Boltzmann formulation for transport and flow, closed with the compact multi-reflection boundary and interface-conjugate schemes, J. Comput. Sci. (2020) 101215.

[23] L. Gasparini, J.R. Rodrigues, D.A. Augusto, L.M. Carvalho, C. Conopoima, P. Goldfeld, et al., Hybrid parallel iterative sparse linear solver framework for reservoir geomechanical and flow simulation, J. Comput. Sci. 51 (2021) 101330.

[24] U. Trottenberg, C.W. Oosterlee, A. Schüller, Multigrid, Academic Press, San Diego, 2001.

[25] S. Bauer, M. Huber, S. Ghelichkhan, M. Mohr, U. Rüde, B. Wohlmuth, Large-scale simulation of mantle convection based on a new matrix-free approach, J. Comput. Sci. 31 (2019) 60–76.

[26] R.D. Falgout, M. Lecouvez, C.S. Woodward, A parallel-in-time algorithm for variable step multistep methods, J. Comput. Sci. 37 (2019) 101029.

[27] W.L. Briggs, V.E. Henson, S.F. McCormick, A Multigrid Tutorial, Society for Industrial and Applied Mathematics, 2000.

[28] C. Lee, D. Jeong, J. Yang, J. Kim, Nonlinear multigrid implementation for the two-dimensional Cahn–Hilliard equation, Mathematics 8 (1) (2020) 97.

[29] F. De Oliveira, S.R. Franco, V. Pinto, The effect of multigrid parameters in a 3D heat diffusion equation, Appl. Mech. Rev. 23 (1) (2018) 213–221.

[30] W.C.D.O. Casaca, M. Boaventura, A decomposition and noise removal method combining diffusion equation and wave atoms for textured images, Math. Probl. Eng. 2010 (2010) 764639.

[31] D.J. Duffy, Finite Difference Methods in Financial Engineering: A Partial Differential Equation Approach, Wiley, UK, 2006.

[32] Y. Yan, W. Chen, C. Wang, S.M. Wise, A second-order energy stable BDF numerical scheme for the Cahn–Hilliard equation, Commun. Comput. Phys. 23 (2) (2018) 572–602.

[33] K. Cheng, W. Feng, C. Wang, S.M. Wise, An energy stable fourth order finite difference scheme for the Cahn–Hilliard equation, J. Comput. Appl. Math. 362 (2019) 574–595.

[34] J. Yang, J. Kim, An energy stable second-order accurate scheme for microphase separation of periodic diblock copolymers, East Asian J. Appl. Math. 11 (2) (2021) 234–254.

[35] J. Kim, S. Lee, Y. Choi, A conservative Allen–Cahn equation with a space–time dependent Lagrange multiplier, Internat. J. Engrg. Sci. 84 (2014) 11–17.

[36] B. Wongsaijai, N. Sukantamala, K. Poochinapan, A mass-conservative higher-order ADI method for solving unsteady convection–diffusion equations, Adv. Differential Equations 2020 (1) (2020) 1–24.

[37] S. Pei, Y. Hou, B. You, A linearly second-order energy stable scheme for the phase field crystal model, Appl. Numer. Math. 140 (2019) 134–164.

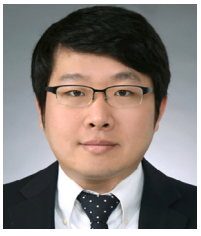
[38] W. Chen, W. Feng, Y. Liu, C. Wang, S.M. Wise, A second order energy stable scheme for the Cahn–Hilliard–Hele–Shaw equations, Discrete Contin. Dyn. Syst. Ser. B 24 (1) (2019) 149–182.

[39] J. Yang, J. Kim, Numerical study of the ternary Cahn–Hilliard fluids by using an efficient modified scalar auxiliary variable approach, Commun. Nonlinear Sci. Numer. Simul. 102 (2021) 105923.

Junxiang Yang is a postdoctoral researcher at Sun Yat-sen University. He received the M.S. and Ph.D. degrees in Applied Mathematics from Korea University, Korea, in 2019 and 2021, respectively. And he received B.S. degree in Naval Architecture and Ocean Engineering from Chongqing Jiaotong University in 2017. His research interests include computational fluid dynamics, mathematical modeling, and scientific computing.

Chaeyoung Lee is a postdoctoral researcher at Korea University. She received the M.S. and Ph.D. degrees in Applied Mathematics from Korea University, Korea, in 2014 and 2021, respectively. And she received B.S. degree in Mathematics from Korea University in 2012. Her research interests include numerical analysis, Mathematical modeling, computational finance, and scientific computing.

Soobin Kwak received Bachelor of Science at Daegu University, South Korea in 2020. Currently, she is a graduate student pursuing a Master's degree at Department of Mathematics, Korea University in South Korea. Her research interests focus on finite difference method for partial difference equations.



Yongho Choi has been an assistant professor at Daegu University since 2019. He received his Ph.D. in 2018 after an integrated master's and doctoral course in Applied Mathematics from Korea University. He also received his B.S. degree from the Department of Mathematics, Kyonggi University, Korea in 2012. His research interests include PDE on surfaces, pattern formation, image analysis, and parallel computation.

Junseok Kim received his Ph.D. in Applied Mathematics from the University of Minnesota, U.S.A. in 2002. He also received his B.S. degree from the Department of Mathematics Education, Korea University, Korea in 1995. He joined the faculty of Korea University, Korea in 2008 where he is currently an full professor at the Department of Mathematics. His research interests are in computational finance and computational fluid dynamics.