Fourth-Order Compact FDMs for Steady and Time-Dependent Nonlinear Convection-Diffusion Equations

Qiwei Feng[a,1,*], Catalin Trenchea[a,1]

[a]*Department of Mathematics, University of Pittsburgh, Pittsburgh, PA 15260 USA.*

Abstract

In this paper, we discuss the steady and time-dependent nonlinear convection-diffusion (advection-diffusion) equations with the Dirichlet boundary condition. For the steady nonlinear equation, we use an iteration method to reformulate the nonlinear equation into its linear counterpart, and derive a fourth-order compact 9-point finite difference method (FDM) to solve the reformulated equation on a uniform Cartesian grid. To increase the accuracy, we modify the FDM to reduce the pollution effect. The linear system of the FDM generates an M-matrix, provided the mesh size h is sufficiently small. For the time-dependent nonlinear equation, we discrete the temporal domain using the Crank-Nicolson (CN), BDF3, BDF4 time stepping methods, and apply a similar iterative method to rewrite the nonlinear equation as the same linear convection-diffusion equation. Then we propose the second-order to fourth-order compact 9-point FDMs with the reduced pollution effects on a uniform Cartesian grid. We prove that all FDMs satisfy the discrete maximum principle for sufficiently small h. Several examples with the variable and time-dependent diffusion coefficients and challenging nonlinear terms (not limited to the Burgers equation) are provided to verify the accuracy and the desired convergence rates in the l_2 and l_∞ norms in space and time. We also compare our second-order CN method with the third-order BDF3 method and the discontinuous Galerkin (DG) method, and the numerical results demonstrate that our FDM with the coarse time step generates the small error. Especially, if the same BDF3 scheme is applied, our error is 1.6% of that obtained from the DG method. The proposed methods can be easily extended to a 3D spatial domain and more general nonlinear convection-diffusion-reaction equations.

Keywords: Fourth-order of consistency, compact 9-point FDM, reduced pollution effect, nonlinear convection-diffusion equation
2020 MSC: 65M06, 65N06, 35G30, 35G31

1. Introduction

The nonlinear convection-diffusion-reaction equations (advection-diffusion-reaction equations, the terms 'convection' and 'advection' are used indiscriminately in numerical analysis [16, p.2]) arise in a wide range of important physical and engineering applications. They serve as the foundation for several well-known models, such as the porous medium equation for the investigation of the fluid transport in porous materials, the Navier-Stokes equation for modeling flows in pipes and channels, and the Fisher-Kolmogorov-Petrovsky-Piskunov equation to describe population growth. To analyze and numerically solve various nonlinear convection-diffusion-reaction equations, we present the following literature review. Additional related works can be found in their references.

In [3], Burman and al. proposed the stabilized Galerkin approximation for the steady nonlinear convection-diffusion-reaction equation $-\epsilon\Delta u + \boldsymbol{v} \cdot \nabla u + r(u) = f$ with the Dirichlet and Neumann boundary conditions, and the nonlinear reaction term $r(u)$, on a square in the numerical examples. In [5], Clain and al. established the second-order to sixth-order FDMs with the uniform Cartesian grids for the steady nonlinear convection-diffusion-reaction equation $-\kappa(u)\Delta u + \boldsymbol{F}(u) \cdot \nabla(u) + r(u)u = f$ with the nonlinear terms of diffusion $\kappa(u)$, convection $F(u)$, and reaction $r(u)$, the Dirichlet, Neumann, and nonlinear Robin conditions in a curved domain. For $u_t - \epsilon\Delta u + \nabla \cdot \boldsymbol{F}(u) = f$ with the nonlinear convective term $\boldsymbol{F}(u)$, the Dirichlet and/or Neumann boundary conditions on a bounded polyhedral domain in 2D and/or 3D, the priori asymptotic error estimate of the discontinuous Galerkin (DG) method was deduced in [7, 8, 9, 23], and [11] proposed the convergence analysis of the combined finite volume-finite element (FV-FE) method. In [7, 8, 9], Dolejší and al. tested the 2D viscous Burgers equation with the Dirichlet boundary condition on a square in the numerical example. Nguyen and al. introduced in [24] the fourth-order DG method for $-\nabla \cdot (\kappa\nabla u) + \nabla \cdot \boldsymbol{F}(u) = f$, and the third-order DG method for $u_t - \nabla \cdot (\kappa\nabla u) + \nabla \cdot \boldsymbol{F}(u) = f$ with the constant diffusion κ, the nonlinear convection $\boldsymbol{F}(u) = (u^2/2, u^2/2)$,

---

and the Dirichlet boundary condition on a rectangle in the numerical experiments. For $u_t - \nabla \cdot (K(u)\nabla u) + \nabla \cdot \boldsymbol{F}(u) = 0$ with the periodic boundary condition and the nonlinear diffusion term $K(u)$, where $K(u)$ is a symmetric and positive definite matrix of the variable u, [27] constructed the error estimate for the semi-discrete local DG method on a rectangular domain, and [28] provided the error analysis for the nonsymmetric DG method using the uniform and nonuniform meshes on a bounded domain in 1D and 2D. The equation $u_t - (u^2)_{xx} - (u^2)_{yy} = 0$ and the 2D strongly degenerate parabolic problem in a square domain were examined in [28]. New central schemes that maintain the high-resolution independent of $O(1/\tau)$ for the general nonlinear diffusion equation $u_t - (P(u, u_x, u_y))_x - (Q(u, u_x, u_y))_x + \nabla \cdot \boldsymbol{F}(u) = 0$ were offered in [18], where τ is the time step of the temporal discretization. In [10], Eymard and al. described the combined FV-FE method for $(\alpha(u))_t - \nabla \cdot (K\nabla u) + \nabla \cdot (vu) + r(u) = f$ with nonlinear $\alpha(u)$ and $r(u)$, the Dirichlet and Neumann boundary conditions, and the nonmatching grids on a bounded domain in 2D and 3D. In [25], Tezduyar and al. built the streamline-upwind/Petrov-Galerkin method for the steady nonlinear convection-diffusion-reaction equation systems. The numerical approximated solutions from [25] are accurate with the minimal numerical dissipation and oscillations. For the systems of the time-dependent nonlinear convection-diffusion-reaction equations, [1] presented a conforming FEM of up to the fourth order using the regular triangular mesh, [6] extended the Runge-Kutta DG method of the purely hyperbolic system to design the local DG method, and [22] extended the von Neumann analysis to prove the nonlinear stability of the high-order DG method. Furthermore, [4] analyzed the large time behavior of the nonlinear FVMs for the linear anisotropic convection-diffusion equations with the Dirichlet and Neumann boundary conditions in a bounded domain.

In this paper, we consider the steady and time-dependent nonlinear convection-diffusion equations in the spatial domain $\Omega = (0, 1)^2$ and the temporal domain $I = [0, 1]$ as follows

$$\begin{cases} -\nabla \cdot (\kappa \nabla u) + \nabla \cdot \boldsymbol{F}(u) = f, & (x, y) \in \Omega, \\ u = g, & (x, y) \in \partial\Omega, \end{cases} \tag{1.1}$$

and

$$\begin{cases} u_t - \nabla \cdot (\kappa \nabla u) + \nabla \cdot \boldsymbol{F}(u) = f, & (x, y) \in \Omega \quad \text{and} \quad t \in I, \\ u = u^0, & (x, y) \in \Omega \quad \text{and} \quad t = 0, \\ u = g, & (x, y) \in \partial\Omega \quad \text{and} \quad t \in I, \end{cases} \tag{1.2}$$

where u is the unknown scalar variable function; κ and f denote the available scalar-valued diffusion coefficient and the source term, respectively; g and u^0 represent the given Dirichlet boundary function of u on Ω and the initial value of u at $t = 0$, respectively. Here $\boldsymbol{F}$ is the vector-valued nonlinear function of u, i.e.,

$$\boldsymbol{F}(u) = (\alpha(u), \beta(u)), \quad \text{with two nonlinear scalar functions } \alpha(u), \beta(u) \text{ of the variable } u.$$

For example, if

$$(\alpha(u), \beta(u)) = (u^2/2, u^2/2), \quad (\sin(u), \cos(u)), \quad (\cos(u), \exp(u)),$$

then

$$\nabla \cdot \boldsymbol{F}(u) = uu_x + uu_y, \quad \cos(u)u_x - \sin(u)u_y, \quad -\sin(u)u_x + \exp(u)u.$$

In this paper, we assume that $\kappa > 0$, and $u, \kappa, f, \alpha, \beta$ are smooth in Ω and I. The rest of the paper is organized as follows:

In Section 2 we consider the steady nonlinear convection-diffusion equation (1.1). We first reformulate (1.1) into a linear convection-diffusion equation in Section 2.1. Then, we provide the explicit expression of the fourth-order compact 9-point FDM for the linear equation in Section 2.3 by using the techniques in Section 2.2. To increase the accuracy of our FDM, we reduce the pollution effect in Section 2.4.

In Section 3 we discuss the time-dependent nonlinear convection-diffusion equation (1.2). We apply the Crank-Nicolson (CN), BDF3, and BDF4 time stepping methods, and rewrite (1.2) as the linear convection-diffusion equation in Section 3.1. The second-to fourth-order compact 9-point FDMs with the reduced pollution effects are proposed in Section 3.2.

In Section 4 we provide various examples with the variable $\kappa(x, y), \kappa(x, y, t)$, and the challenging $\alpha(u), \beta(u)$, and compare our scheme with the discontinuous Galerkin (DG) and the BDF3 methods in [24].

In Section 5 we highlight the main conclusion of this paper.

2. Fourth-order compact 9-point FDMs for the steady nonlinear convection-diffusion equation

In this section we convert the steady nonlinear convection-diffusion equation (1.1) into its linear counterpart by a fixed point method, and derive the fourth-order compact 9-point FDMs.

2.1. Reformulation of the steady nonlinear convection-diffusion equation

Clearly, (1.1) results in

$$-\kappa \Delta u - \kappa_x u_x - \kappa_y u_y + \alpha_u(u)u_x + \beta_u(u)u_y = f.$$

Let

$$a := \frac{\kappa_x - \alpha_u(u)}{\kappa}, \qquad b := \frac{\kappa_y - \beta_u(u)}{\kappa}, \qquad \psi := -\frac{f}{\kappa}. \tag{2.1}$$

Then (1.1) is equivalent to

$$\begin{cases} \Delta u + a u_x + b u_y = \psi, & (x, y) \in \Omega, \\ u = g, & (x, y) \in \partial\Omega. \end{cases} \tag{2.2}$$

In this paper, an iteration method is applied to numerically solve the nonlinear equation (2.2):

$$\begin{cases} \Delta u_{k+1} + a_k(u_{k+1})_x + b_k(u_{k+1})_y = \psi, & (x, y) \in \Omega, \\ u_{k+1} = g, & (x, y) \in \partial\Omega, \\ k := k + 1, \text{ the initial } k = 0, \ u_0 = 0, \end{cases} \tag{2.3}$$

where k denotes the index of the k-th iteration, and

$$a_k := \frac{\kappa_x - \alpha_u(u_k)}{\kappa}, \qquad b_k := \frac{\kappa_y - \beta_u(u_k)}{\kappa}. \tag{2.4}$$

As u_k is computed in the k-th iteration, (2.3) can be considered as the linear problem in the $(k + 1)$-th iteration. Precisely, we rewrite the nonlinear problem (1.1) as the following steady linear convection-diffusion equation:

$$\begin{cases} \Delta u + a u_x + b u_y = \psi, & (x, y) \in \Omega, \\ u = g, & (x, y) \in \partial\Omega, \end{cases} \tag{2.5}$$

where

$$u := u_{k+1}, \qquad a := a_k = \frac{\kappa_x - \alpha_u(u_k)}{\kappa}, \qquad b := b_k = \frac{\kappa_y - \beta_u(u_k)}{\kappa}. \tag{2.6}$$

To solve (2.5) efficiently, we propose the fourth-order compact 9-point FDMs in the following Sections 2.2 to 2.4.

2.2. Techniques to derive the high-order FDMs

Figure 1: The illustration for $\Lambda_M = \Lambda_M^1 \cup \Lambda_M^2$ in (2.7)–(2.8) with $M = 7$.

In this section, we present the key techniques for deriving the high-order (not limited to the fourth-order) FDMs in 2D, which can be naturally extended to the three-dimensional spatial domain. For $M \in \mathbb{N}_0 := \mathbb{N} \cup \{0\}$, we define (see Fig. 1 for an illustration)

$$\Lambda_M := \{(m, n) \in \mathbb{N}_0^2 \ : \ m + n \leq M\}, \tag{2.7}$$

$$\Lambda_M^2 := \Lambda_M \setminus \Lambda_M^1 \quad \text{with} \quad \Lambda_M^1 := \{(m, n) \in \Lambda_M \ : \ m = 0, 1\}. \tag{2.8}$$

For any smooth function φ, we also define

$$\varphi^{(m,n)} := \frac{\partial^{m+n}\varphi(x, y)}{\partial^m x \partial^n y}, \text{ i.e., the } (m, n)\text{-th order partial derivative of } \varphi(x, y). \tag{2.9}$$

3

$$
\begin{array}{llllll}
u^{(2,0)} & u^{(3,0)} & u^{(4,0)} & u^{(5,0)} & u^{(6,0)} & u^{(7,0)} \\
u^{(2,1)} & u^{(3,1)} & u^{(4,1)} & u^{(5,1)} & u^{(6,1)} \\
u^{(2,2)} & u^{(3,2)} & u^{(4,2)} & u^{(5,2)} \\
u^{(2,3)} & u^{(3,3)} & u^{(4,3)} \\
u^{(2,4)} & u^{(3,4)} \\
u^{(2,5)}
\end{array}
\quad = \quad
\begin{array}{ll}
u^{(0,0)} & u^{(1,0)} \\
u^{(0,1)} & u^{(1,1)} \\
u^{(0,2)} & u^{(1,2)} \\
u^{(0,3)} & u^{(1,3)} \\
u^{(0,4)} & u^{(1,4)} \\
u^{(0,5)} & u^{(1,5)} \\
u^{(0,6)} & u^{(1,6)} \\
u^{(0,7)}
\end{array}
\quad + \quad
\begin{array}{llllll}
a^{(0,0)} & a^{(1,0)} & a^{(2,0)} & a^{(3,0)} & a^{(4,0)} & a^{(5,0)} \\
a^{(0,1)} & a^{(1,1)} & a^{(2,1)} & a^{(3,1)} & a^{(4,1)} \\
a^{(0,2)} & a^{(1,2)} & a^{(2,2)} & a^{(3,2)} \\
a^{(0,3)} & a^{(1,3)} & a^{(2,3)} \\
a^{(0,4)} & a^{(1,4)} \\
a^{(0,5)}
\end{array}
$$

$$
+ \quad
\begin{array}{llllll}
b^{(0,0)} & b^{(1,0)} & b^{(2,0)} & b^{(3,0)} & b^{(4,0)} & b^{(5,0)} \\
b^{(0,1)} & b^{(1,1)} & b^{(2,1)} & b^{(3,1)} & b^{(4,1)} \\
b^{(0,2)} & b^{(1,2)} & b^{(2,2)} & b^{(3,2)} \\
b^{(0,3)} & b^{(1,3)} & b^{(2,3)} \\
b^{(0,4)} & b^{(1,4)} \\
b^{(0,5)}
\end{array}
\quad + \quad
\begin{array}{llllll}
\psi^{(0,0)} & \psi^{(1,0)} & \psi^{(2,0)} & \psi^{(3,0)} & \psi^{(4,0)} & \psi^{(5,0)} \\
\psi^{(0,1)} & \psi^{(1,1)} & \psi^{(2,1)} & \psi^{(3,1)} & \psi^{(4,1)} \\
\psi^{(0,2)} & \psi^{(1,2)} & \psi^{(2,2)} & \psi^{(3,2)} \\
\psi^{(0,3)} & \psi^{(1,3)} & \psi^{(2,3)} \\
\psi^{(0,4)} & \psi^{(1,4)} \\
\psi^{(0,5)}
\end{array}
$$

Figure 2: The illustration for (2.17) with $M = 7$.

The main technique to derive the high-order FDM is the following formula (2.33). Note that (2.33) can be obtained by [14, (2.2)-(2.12)]. For the readers' convenience and to further simplify the corresponding expression in [14, (2.12)], we provide more details in the following relations (2.10)-(2.33). It follows from (2.5) and (2.9) that

$$u^{(2,0)} = -u^{(0,2)} - a^{(0,0)}u^{(1,0)} - b^{(0,0)}u^{(0,1)} + \psi^{(0,0)}, \tag{2.10}$$

where a, b, u are defined in (2.6), and ψ is defined in (2.1). Next, we take the m-th derivative with respect to x of (2.10),

$$u^{(m+2,0)} = -u^{(m,2)} - \sum_{i=0}^{m} \binom{m}{i} a^{(m-i,0)} u^{(i+1,0)} - \sum_{i=0}^{m} \binom{m}{i} b^{(m-i,0)} u^{(i,1)} + \psi^{(m,0)}, \quad m \geq 0. \tag{2.11}$$

We also take the n-th derivative with respect to y of (2.11) to obtain

$$
\begin{aligned}
u^{(m+2,n)} = & -u^{(m,n+2)} - \sum_{j=0}^{n}\sum_{i=0}^{m} \binom{n}{j}\binom{m}{i} a^{(m-i,n-j)} u^{(i+1,j)} \\
& - \sum_{j=0}^{n}\sum_{i=0}^{m} \binom{n}{j}\binom{m}{i} b^{(m-i,n-j)} u^{(i,j+1)} + \psi^{(m,n)}, \quad m, n \geq 0.
\end{aligned}
\tag{2.12}
$$

Then (2.12) implies

$$u^{(m+2,n)} \longrightarrow (u^{(m,n+2)}, u^{(i+1,j)}, u^{(i,j+1)}, \psi^{(m,n)}), \quad \text{with} \quad 0 \leq i \leq m, \quad 0 \leq j \leq n, \quad m, n \geq 0. \tag{2.13}$$

Applying (2.13) recursively $m + 1$ times yields

$$u^{(m+2,n)} \longrightarrow (u^{(0,j_0)}, u^{(1,j_1)}, \psi^{(i,j)}), \tag{2.14}$$

with $0 \leq j_0 \leq m + n + 2$, $0 \leq j_1 \leq m + n + 1$, $m, n \geq 0$, $0 \leq i, j \leq m + n$, and $i + j \leq m + n$ (examples of (2.14) for $u^{(m+2,n)}$ with $m = 0, 1, 2$ are explicitly given in the following identities (2.20)–(2.23)). Namely,

$$u^{(m+2,n)} = \sum_{j=0}^{m+2} \xi_{m+2,n,0,j} u^{(0,j)} + \sum_{j=0}^{m+1} \xi_{m+2,n,1,j} u^{(1,j)} + \sum_{\substack{i,j=0 \\ i+j\leq m+n}}^{m+n} \eta_{m+2,n,i,j} \psi^{(i,j)}, \qquad m, n \geq 0, \tag{2.15}$$

where $\xi_{m+2,n,0,j}, \xi_{m+2,n,1,j}, \eta_{m+2,n,i,j}$ are uniquely determined by the above recursive algorithms (2.12)–(2.14), and only depend on the high-order partial derivatives of a and b. Now, we replace $(m + 2, n)$ by (p, q) in (2.15) to get

$$u^{(p,q)} = \sum_{j=0}^{p} \xi_{p,q,0,j} u^{(0,j)} + \sum_{j=0}^{p-1} \xi_{p,q,1,j} u^{(1,j)} + \sum_{\substack{i,j=0 \\ i+j\leq p+q-2}}^{p+q-2} \eta_{p,q,i,j} \psi^{(i,j)}, \qquad p \geq 2, q \geq 0. \tag{2.16}$$

4

According to the definitions of $\Lambda_M, \Lambda_M^1, \Lambda_M^2$ in (2.7) and (2.8),

$$\mathsf{u}^{(p,q)} = \sum_{(m,n)\in\Lambda_{p+q}^1} \xi_{p,q,m,n}\mathsf{u}^{(m,n)} + \sum_{(m,n)\in\Lambda_{p+q-2}^2} \eta_{p,q,m,n}\psi^{(m,n)}, \qquad (p,q)\in\Lambda_M^2, \tag{2.17}$$

where $\psi, \mathsf{a}, \mathsf{b}, \mathsf{u}$ are defined in (2.1) and (2.6), and $\xi_{p,q,m,n}, \eta_{p,q,m,n}$ are uniquely determined by the high-order partial derivatives of a and b (see Fig. 2 for an illustration). By the derivations (2.12)–(2.16) of (2.17), we can say that $\xi_{p,q,m,n}$ and $\eta_{p,q,m,n}$ are uniquely determined by the multiplications of the high-order partial derivatives of a and b, i.e., $\xi_{p,q,m,n}$ and $\eta_{p,q,m,n}$ in (2.17) belong to

$$S := \mathrm{span}\left\{ \prod_{\substack{i_1,j_1\in\mathbb{N}_0 \\ v_1,w_1\in\mathbb{N}_0}} \mathsf{a}^{(i_1,j_1)}\mathsf{b}^{(v_1,w_1)}, \prod_{\substack{i_2,j_2\in\mathbb{N}_0 \\ v_2,w_2\in\mathbb{N}_0}} \mathsf{a}^{(i_2,j_2)}\mathsf{b}^{(v_2,w_2)}, \ldots, \prod_{\substack{i_k,j_k\in\mathbb{N}_0 \\ v_k,w_k\in\mathbb{N}_0}} \mathsf{a}^{(i_k,j_k)}\mathsf{b}^{(v_k,w_k)} \right\}, \tag{2.18}$$

with $k\in\mathbb{N}$, and the coefficients in $\mathbb{R}$. Precisely,

$$\begin{aligned}
\xi_{p,q,m,n} &= \sum_{k\in\mathbb{N}} c_{p,q,m,n,k} \prod_{i_k,j_k,v_k,w_k\in\mathbb{N}_0} \mathsf{a}^{(i_k,j_k)}\mathsf{b}^{(v_k,w_k)}, \quad \text{with} \quad c_{p,q,m,n,k}\in\mathbb{R}, \\
\eta_{p,q,m,n} &= \sum_{k\in\mathbb{N}} d_{p,q,m,n,k} \prod_{i_k,j_k,v_k,w_k\in\mathbb{N}_0} \mathsf{a}^{(i_k,j_k)}\mathsf{b}^{(v_k,w_k)}, \quad \text{with} \quad d_{p,q,m,n,k}\in\mathbb{R}.
\end{aligned} \tag{2.19}$$

For example, $\mathsf{u}^{(p,q)}$ with $(p,q)\in\Lambda_M^2$, $M\geq 4$, and $p=2,3,4$ in (2.17) can be explicitly given by

$$\mathsf{u}^{(2,q)} = -\mathsf{u}^{(0,q+2)} - \sum_{k=0}^{q}\binom{q}{k}\left[\mathsf{a}^{(0,q-k)}\mathsf{u}^{(1,k)} + \mathsf{b}^{(0,q-k)}\mathsf{u}^{(0,k+1)}\right] + \psi^{(0,q)}, \quad 0\leq q\leq M-2, \tag{2.20}$$

$$\begin{aligned}
\mathsf{u}^{(3,q)} = &- \mathsf{u}^{(1,q+2)} + \sum_{k=0}^{q}\binom{q}{k}[(\mathsf{ab}-\mathsf{b}^{(1,0)})^{(0,q-k)}\mathsf{u}^{(0,k+1)} + \mathsf{a}^{(0,q-k)}\mathsf{u}^{(0,k+2)} \\
&+ (\mathsf{a}^2-\mathsf{a}^{(1,0)})^{(0,q-k)}\mathsf{u}^{(1,k)}] - \sum_{k=0}^{q}\binom{q}{k}\mathsf{b}^{(0,q-k)}\mathsf{u}^{(1,k+1)} \\
&- \sum_{k=0}^{q}\binom{q}{k}\mathsf{a}^{(0,q-k)}\psi^{(0,k)} + \psi^{(1,q)}, \quad 0\leq q\leq M-3,
\end{aligned} \tag{2.21}$$

$$\begin{aligned}
\mathsf{u}^{(4,q)} = &\sum_{k=0}^{q}\binom{q}{k}\left[\xi_{4,0,0,1}^{(0,q-k)}\mathsf{u}^{(0,k+1)} + \xi_{4,0,0,2}^{(0,q-k)}\mathsf{u}^{(0,k+2)} + 2\mathsf{b}^{(0,q-k)}\mathsf{u}^{(0,k+3)} + \xi_{4,0,1,0}^{(0,q-k)}\mathsf{u}^{(1,k)} \right. \\
&\left. + \xi_{4,0,1,1}^{(0,q-k)}\mathsf{u}^{(1,k+1)} + 2\mathsf{a}^{(0,q-k)}\mathsf{u}^{(1,k+2)}\right] + \mathsf{u}^{(0,q+4)} + \sum_{k=0}^{q}\binom{q}{k}\left[(\mathsf{a}^2-2\mathsf{a}^{(1,0)})^{(0,q-k)}\psi^{(0,k)} \right. \\
&\left. - \mathsf{a}^{(0,q-k)}\psi^{(1,k)} - \mathsf{b}^{(0,q-k)}\psi^{(0,k+1)}\right] + \psi^{(2,q)} - \psi^{(0,q+2)}, \quad 0\leq q\leq M-4,
\end{aligned} \tag{2.22}$$

where

$$\begin{aligned}
\xi_{4,0,0,1} &= 2\mathsf{a}^{(1,0)}\mathsf{b} - \mathsf{a}^2\mathsf{b} + \mathsf{b}^{(0,1)}\mathsf{b} + \mathsf{ab}^{(1,0)} + \mathsf{b}^{(0,2)} - \mathsf{b}^{(2,0)}, & \xi_{4,0,0,2} &= 2\mathsf{a}^{(1,0)} - \mathsf{a}^2 + \mathsf{b}^2 + 2\mathsf{b}^{(0,1)}, \\
\xi_{4,0,1,0} &= 3\mathsf{aa}^{(1,0)} - \mathsf{a}^3 + \mathsf{a}^{(0,1)}\mathsf{b} + \mathsf{a}^{(0,2)} - \mathsf{a}^{(2,0)}, & \xi_{4,0,1,1} &= 2\mathsf{ab} + 2\mathsf{a}^{(0,1)} - 2\mathsf{b}^{(1,0)}.
\end{aligned} \tag{2.23}$$

To derive a fourth-order compact 9-point FDM, we use the uniform Cartesian grid in the spatial domain $\Omega=(0,1)^2$ as follows:

$$x_i := ih, \quad i=0,\ldots,N_1, \quad y_j := jh, \quad j=0,\ldots,N_1, \quad \text{and} \quad h:=1/N_1, \tag{2.24}$$

for some integer $N_1\in\mathbb{N}$. To simplify the presentation, we adapt the following notations throughout the remainder of this paper:

$$\begin{aligned}
\mathsf{a}^{(m,n)} &:= \frac{\partial^{m+n}\mathsf{a}(x_i,y_j)}{\partial^m x\partial^n y}, & \mathsf{b}^{(m,n)} &:= \frac{\partial^{m+n}\mathsf{b}(x_i,y_j)}{\partial^m x\partial^n y}, \\
\mathsf{u}^{(m,n)} &:= \frac{\partial^{m+n}\mathsf{u}(x_i,y_j)}{\partial^m x\partial^n y}, & \psi^{(m,n)} &:= \frac{\partial^{m+n}\psi(x_i,y_j)}{\partial^m x\partial^n y}.
\end{aligned} \tag{2.25}$$

The Taylor approximation at the base point (x_i,y_j) and the definitions of $\Lambda_M, \Lambda_M^1, \Lambda_M^2$ in (2.7) and (2.8) yield

$$\mathsf{u}(x+x_i, y+y_j) = \sum_{(m,n)\in\Lambda_M} \mathsf{u}^{(m,n)}\frac{x^m y^n}{m!n!} + O(h^{M+1}) \tag{2.26}$$

5

$$= \sum_{(m,n)\in\Lambda_M^1} \mathsf{u}^{(m,n)} \frac{x^m y^n}{m!n!} + \sum_{(p,q)\in\Lambda_M^2} \mathsf{u}^{(p,q)} \frac{x^p y^q}{p!q!} + O(h^{M+1}), \tag{2.27}$$

where $x, y \in [-h, h]$. From (2.17), we see that

$$\sum_{(p,q)\in\Lambda_M^2} \mathsf{u}^{(p,q)} \frac{x^p y^q}{p!q!} = \sum_{(p,q)\in\Lambda_M^2} \left[\sum_{(m,n)\in\Lambda_{p+q}^1} \xi_{p,q,m,n} \mathsf{u}^{(m,n)} + \sum_{(m,n)\in\Lambda_{p+q-2}} \eta_{p,q,m,n} \psi^{(m,n)} \right] \frac{x^p y^q}{p!q!}$$

$$= \sum_{\substack{p=2,q=0 \\ p+q\leq M}}^{M} \left[\sum_{n=0}^{p+q} \xi_{p,q,0,n} \mathsf{u}^{(0,n)} + \sum_{n=0}^{p+q-1} \xi_{p,q,1,n} \mathsf{u}^{(1,n)} + \sum_{\substack{m,n=0 \\ m+n\leq p+q-2}}^{p+q-2} \eta_{p,q,m,n} \psi^{(m,n)} \right] \frac{x^p y^q}{p!q!}. \tag{2.28}$$

Clearly,

$$\sum_{\substack{p=2,q=0 \\ p+q\leq M}}^{M} \left[\sum_{n=0}^{p+q} \xi_{p,q,0,n} \mathsf{u}^{(0,n)} \right] \frac{x^p y^q}{p!q!} = \sum_{n=0}^{M} \left[\sum_{\substack{p=2,q=0 \\ n\leq p+q\leq M}}^{M} \xi_{p,q,0,n} \frac{x^p y^q}{p!q!} \right] \mathsf{u}^{(0,n)}, \tag{2.29}$$

$$\sum_{\substack{p=2,q=0 \\ p+q\leq M}}^{M} \left[\sum_{n=0}^{p+q-1} \xi_{p,q,1,n} \mathsf{u}^{(1,n)} \right] \frac{x^p y^q}{p!q!} = \sum_{n=0}^{M-1} \left[\sum_{\substack{p=2,q=0 \\ n+1\leq p+q\leq M}}^{M} \xi_{p,q,1,n} \frac{x^p y^q}{p!q!} \right] \mathsf{u}^{(1,n)}, \tag{2.30}$$

$$\sum_{\substack{p=2,q=0 \\ p+q\leq M}}^{M} \left[\sum_{\substack{m,n=0 \\ m+n\leq p+q-2}}^{p+q-2} \eta_{p,q,m,n} \psi^{(m,n)} \right] \frac{x^p y^q}{p!q!} = \sum_{\substack{m,n=0 \\ m+n\leq M-2}}^{M-2} \left[\sum_{\substack{p=2,q=0 \\ m+n+2\leq p+q\leq M}}^{M} \eta_{p,q,m,n} \frac{x^p y^q}{p!q!} \right] \psi^{(m,n)}. \tag{2.31}$$

Plugging (2.29)–(2.31) into (2.28) gives

$$\sum_{(p,q)\in\Lambda_M^2} \mathsf{u}^{(p,q)} \frac{x^p y^q}{p!q!} = \sum_{(m,n)\in\Lambda_M^1} \left[\sum_{(p,q)\in\Lambda_M^2\setminus\Lambda_{m+n-1}^2} \xi_{p,q,m,n} \frac{x^p y^q}{p!q!} \right] \mathsf{u}^{(m,n)}$$

$$+ \sum_{(m,n)\in\Lambda_{M-2}} \left[\sum_{(p,q)\in\Lambda_M^2\setminus\Lambda_{m+n+1}^2} \eta_{p,q,m,n} \frac{x^p y^q}{p!q!} \right] \psi^{(m,n)}. \tag{2.32}$$

Then, we substitute (2.32) into (2.27),

$$\mathsf{u}(x + x_i, y + y_j) = \sum_{(m,n)\in\Lambda_M^1} \mathsf{u}^{(m,n)} G_{M,m,n}(x,y) + \sum_{(m,n)\in\Lambda_{M-2}} \psi^{(m,n)} H_{M,m,n}(x,y) + O(h^{M+1}), \tag{2.33}$$

where $x, y \in [-h, h]$, and

$$G_{M,m,n}(x,y) := \frac{x^m y^n}{m!n!} + \sum_{(p,q)\in\Lambda_M^2\setminus\Lambda_{m+n-1}^2} \xi_{p,q,m,n} \frac{x^p y^q}{p!q!}, \qquad H_{M,m,n}(x,y) := \sum_{(p,q)\in\Lambda_M^2\setminus\Lambda_{m+n+1}^2} \eta_{p,q,m,n} \frac{x^p y^q}{p!q!}, \tag{2.34}$$

where the expressions of $\xi_{p,q,m,n}$ and $\eta_{p,q,m,n}$ are provided in (2.19). Finally, we use (2.33) to construct the fourth-order compact 9-point FDMs for (2.5) in the following Section 2.3.

2.3. Fourth-order compact 9-point FDMs

Recall that u denotes the exact solution of (2.5), and the grid point (x_i, y_j) with the uniform mesh size h is defined in (2.24). We define u_h as the numerical solution computed by the FDM, $\mathsf{u}_{i,j} = \mathsf{u}(x_i, y_j)$, and $(\mathsf{u}_h)_{i,j}$ is the value of u_h at (x_i, y_j). The fourth-order compact 9-point FDM at the grid point (x_i, y_j) is expressed as

$$h^{-2}\mathcal{L}_h\mathsf{u}_h := h^{-2} \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell}(\mathsf{u}_h)_{i+k,j+\ell} = F_{i,j}, \tag{2.35}$$

with

$$C_{k,\ell} := \sum_{p=0}^{M+1} c_{k,\ell,p} h^p, \quad c_{k,\ell,p} \in S, \quad M \in \mathbb{N}_0, \tag{2.36}$$

6

where S is defined in (2.18). The coefficients $\{c_{k,\ell,p}\}_{k,\ell=-1,0,1,p=0,\dots,M+1}$ and the right-hand side $F_{i,j}$ are determined next via relations (2.37)–(2.50). We restrict each $c_{k,\ell,p} \in S$ such that the high-order partial derivatives $\mathsf{a}^{(m,n)}$ and $\mathsf{b}^{(m,n)}$ do not appear in the denominator, ensuring that each $c_{k,\ell,p}$ is well defined. We say that the set of nine elements $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ is nontrivial if $C_{k,\ell}|_{h=0} \neq 0$ for at least some $k, \ell = -1, 0, 1$.

The algorithm to derive the fourth-order compact 9-point FDMs: We define the linear operator $\mathcal{L}_h \mathsf{u}$

$$h^{-2}\mathcal{L}_h \mathsf{u} := h^{-2} \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell}\mathsf{u}(kh + x_i, \ell h + y_j). \tag{2.37}$$

Plugging (2.33) into (2.37) and replacing M by $M + 1$ generate

$$\begin{aligned}
h^{-2}\mathcal{L}_h \mathsf{u} &= h^{-2} \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} \sum_{(m,n)\in\Lambda_{M+1}^1} \mathsf{u}^{(m,n)} G_{M+1,m,n}(kh, \ell h) \\
&\quad + h^{-2} \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} \sum_{(m,n)\in\Lambda_{M-1}} \psi^{(m,n)} H_{M+1,m,n}(kh, \ell h) + O(h^M), \\
&= h^{-2} \sum_{(m,n)\in\Lambda_{M+1}^1} \mathsf{u}^{(m,n)} I_{M+1,m,n} + h^{-2} \sum_{(m,n)\in\Lambda_{M-1}} \psi^{(m,n)} J_{M-1,m,n} + O(h^M),
\end{aligned} \tag{2.38}$$

where

$$I_{M+1,m,n} := \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} G_{M+1,m,n}(kh, \ell h), \qquad J_{M-1,m,n} := \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} H_{M+1,m,n}(kh, \ell h), \tag{2.39}$$

with $G_{M+1,m,n}(x, y)$ and $H_{M+1,m,n}(x, y)$ defined by (2.34). Owing to (2.35) and (2.38), the local spatial truncation error

$$h^{-2}\mathcal{L}_h(\mathsf{u} - \mathsf{u}_h) = h^{-2} \sum_{(m,n)\in\Lambda_{M+1}^1} \mathsf{u}^{(m,n)} I_{M+1,m,n} + h^{-2} \sum_{(m,n)\in\Lambda_{M-1}} \psi^{(m,n)} J_{M-1,m,n} - F_{i,j} + O(h^M). \tag{2.40}$$

Let

$$F_{i,j} := \text{the terms of } \Big(h^{-2} \sum_{(m,n)\in\Lambda_{M-1}} \psi^{(m,n)} J_{M-1,m,n}\Big) \text{ with degree} \leq M - 1 \text{ in } h, \tag{2.41}$$

where $J_{M-1,m,n}$ is defined in (2.39) with $H_{M+1,m,n}(x, y)$ in (2.34). Then (2.40) leads to

$$h^{-2}\mathcal{L}_h(\mathsf{u} - \mathsf{u}_h) = h^{-2} \sum_{(m,n)\in\Lambda_{M+1}^1} \mathsf{u}^{(m,n)} I_{M+1,m,n} + O(h^M). \tag{2.42}$$

By (2.7), (2.8), and (2.34), we have that smallest degrees of h among the nonzero terms in $G_{M+1,m,n}(kh, \ell h)$ and $H_{M+1,m,n}(kh, \ell h)$ are 0 and 2, respectively. So, (2.36), (2.39), and (2.41) imply that the smallest degree of h among the nonzero terms in both $I_{M+1,m,n}$ and $F_{i,j}$ is 0.

Now, if $C_{k,\ell}$ in (2.36), $I_{M+1,m,n}$ in (2.39), and $F_{i,j}$ in (2.41) satisfy

$$I_{M+1,m,n} = \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} G_{M+1,m,n}(kh, \ell h) = O(h^{M+2}) \quad \text{for all} \quad (m, n) \in \Lambda_{M+1}^1, \tag{2.43}$$

and

$$C_{0,0}|_{h=0} \neq 0, \qquad F_{i,j}|_{h=0} = \psi, \tag{2.44}$$

where $G_{M+1,m,n}(x, y)$ is defined in (2.34), then (2.42) generates

$$h^{-2}\mathcal{L}_h(\mathsf{u} - \mathsf{u}_h) = O(h^M), \tag{2.45}$$

and $\mathcal{L}_h \mathsf{u}_h$ approximates $\Delta \mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{b}\mathsf{u}_y = \psi$ in (2.5) with the consistency order M at the grid point (x_i, y_j).

By the definitions of $J_{M-1,m,n}$ and $F_{i,j}$ in (2.39) and (2.41), $F_{i,j}$ can be obtained immediately if $\{c_{k,\ell,p}\}|_{k,\ell=-1,0,1}^{p=0,M+1}$ in (2.36) are fixed. So the main task of deriving the high-order FDM is to find $\{c_{k,\ell,p}\}|_{k,\ell=-1,0,1}^{p=0,M+1}$ to satisfy (2.43) and (2.44). Using the symbolic calculation in Maple, the largest M such that the nontrivial $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ solving the corresponding linear system of (2.43) with the variables $\{c_{k,\ell,p}\}|_{k,\ell=-1,0,1}^{p=0,M+1}$ is, $M = 4$. Therefore, the highest consistency order of the compact 9-point FDM based on our technique

using the uniform Cartesian mesh grid for $\Delta \mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{b}\mathsf{u}_y = \psi$, is 4. Furthermore, there exist nine coefficients in $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ satisfying (2.43) with $M = 4$, for any free variables in the following set:

$$\{c_{0,0,0}, c_{0,0,1}, c_{-1,-1,2}, c_{1,1,2}, c_{0,1,3}, c_{1,0,3}, c_{1,0,2}, c_{1,1,3}\} \cup \{c_{k,\ell,p}\}_{k,\ell=-1,0,1}^{p=4,5}. \tag{2.46}$$

The symbolic calculation in Maple also reveals that: if we fix $c_{0,0,0} = -10/3$, $c_{0,0,1} = c_{-1,-1,2} = c_{1,1,2} = c_{0,1,3} = c_{1,0,3} = 0$, $\{c_{k,\ell,p}\}_{k,\ell=-1,0,1,p=4,5} = \{0\}$, $c_{1,0,2} = (\mathsf{a}^2 + \mathsf{a}\mathsf{b} + \mathsf{a}^{(0,1)} + \mathsf{a}^{(1,0)} + \mathsf{b}^{(1,0)} - \mathsf{b}^{(0,1)})/12$, and $c_{1,1,3} = \mathsf{a}^{(0,1)}\mathsf{b}/24$ in (2.46), then there exist the unique $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ and $F_{i,j}$ fulfilling (2.43) and (2.44) with $M = 4$ as follows (every free variable $c_{k,\ell,p}$ in (2.46) is chosen to ensure (2.44) and make the expression of $C_{k,\ell}$ concise):

$$
\begin{aligned}
C_{-1,-1} &= \tfrac{1}{6} - \tfrac{r_1 h}{12} + (r_3 \mathsf{a} - (2\mathsf{a}^{(0,1)} + \mathsf{a}^{(1,0)})\mathsf{b} + \Delta r_1)\tfrac{h^3}{24}, \\
C_{-1,0} &= \tfrac{2}{3} - \tfrac{\mathsf{a}h}{3} + (\mathsf{a}^2 + \mathsf{a}\mathsf{b} + r_2 + r_3)\tfrac{h^2}{12} + (r_2\mathsf{b} - r_3\mathsf{a} - \Delta r_1)\tfrac{h^3}{12}, \\
C_{-1,1} &= \tfrac{1}{6} - \tfrac{r_5 h}{12} - (\mathsf{a}\mathsf{b} + r_4)\tfrac{h^2}{12} + (\mathsf{a}\mathsf{b}^{(1,0)} - r_2\mathsf{b} + \Delta\mathsf{b})\tfrac{h^3}{24}, \\
C_{0,-1} &= \tfrac{2}{3} - \tfrac{\mathsf{b}h}{3} + (\mathsf{a}\mathsf{b} + \mathsf{b}^2 + r_6 + r_7)\tfrac{h^2}{12} + (r_2\mathsf{b} - r_3\mathsf{a} - \Delta r_1)\tfrac{h^3}{12}, \\
C_{0,0} &= -\tfrac{10}{3} - (\mathsf{a}^2 + \mathsf{a}\mathsf{b} + \mathsf{b}^2 + r_4)\tfrac{h^2}{6} + (r_3\mathsf{a} - r_2\mathsf{b} + \Delta r_1)\tfrac{h^3}{12}, \\
C_{0,1} &= \tfrac{2}{3} + \tfrac{\mathsf{b}h}{3} + (\mathsf{a}\mathsf{b} + \mathsf{b}^2 + r_6 + r_7)\tfrac{h^2}{12}, \\
C_{1,-1} &= \tfrac{1}{6} + \tfrac{r_5 h}{12} - (\mathsf{a}\mathsf{b} + r_4)\tfrac{h^2}{12} + (\Delta\mathsf{a} - \mathsf{a}\mathsf{b}^{(0,1)})\tfrac{h^3}{24}, \\
C_{1,0} &= \tfrac{2}{3} + \tfrac{\mathsf{a}h}{3} + (\mathsf{a}^2 + \mathsf{a}\mathsf{b} + r_2 + r_3)\tfrac{h^2}{12}, \\
C_{1,1} &= \tfrac{1}{6} + \tfrac{r_1 h}{12} + \mathsf{a}^{(0,1)}\mathsf{b}\tfrac{h^3}{24},
\end{aligned}
\tag{2.47}
$$

where

$$
\begin{aligned}
r_1 &= \mathsf{a} + \mathsf{b}, & r_2 &= \mathsf{a}^{(0,1)} + \mathsf{a}^{(1,0)}, & r_3 &= \mathsf{b}^{(1,0)} - \mathsf{b}^{(0,1)}, & r_4 &= \mathsf{a}^{(0,1)} + \mathsf{b}^{(1,0)}, \\
r_5 &= \mathsf{a} - \mathsf{b}, & r_6 &= \mathsf{a}^{(0,1)} - \mathsf{a}^{(1,0)}, & r_7 &= \mathsf{b}^{(1,0)} + \mathsf{b}^{(0,1)},
\end{aligned}
\tag{2.48}
$$

and

$$F_{i,j} := \psi - ((\mathsf{a}^{(1,0)} + \mathsf{b}^{(0,1)})\psi - \mathsf{a}\psi^{(1,0)} - \mathsf{b}\psi^{(0,1)} - \Delta\psi)\tfrac{h^2}{12}. \tag{2.49}$$

In particular, if $\mathsf{a} = \mathsf{b}$ (e.g, if $\kappa = $ constant and $\alpha(u) = \beta(u) = u^2/2$, then (2.6) yields the steady viscous Burgers equation), we obtain a more concise expression by the following steps: (2.43) and (2.44) with $\mathsf{a} = \mathsf{b}$, $M = 4$, $c_{0,0,0} = -10/3$, $c_{-1,1,1} = c_{0,0,1} = c_{1,-1,1} = c_{-1,-1,2} = c_{1,1,2} = c_{0,1,3} = c_{1,0,3} = c_{1,1,3} = 0$, $\{c_{k,\ell,p}\}_{k,\ell=-1,0,1,p=4,5} = \{0\}$, and $c_{1,0,2} = (\mathsf{a}^2 + \mathsf{a}^{(1,0)})/6$ result in

$$
\begin{aligned}
C_{-1,-1} &= \tfrac{1}{6} - \tfrac{\mathsf{a}h}{6} + \tfrac{\Delta\mathsf{a}h^3}{12}, & C_{-1,0} &= \tfrac{2}{3} - \tfrac{\mathsf{a}h}{3} + (\mathsf{a}^2 + \mathsf{a}^{(1,0)})\tfrac{h^2}{6} - \tfrac{\Delta\mathsf{a}h^3}{6}, \\
C_{-1,1} &= \tfrac{1}{6} - (\mathsf{a}^2 + \mathsf{a}^{(0,1)} + \mathsf{a}^{(1,0)})\tfrac{h^2}{12} + \tfrac{\Delta\mathsf{a}h^3}{24}, & C_{0,-1} &= \tfrac{2}{3} - \tfrac{\mathsf{a}h}{3} + (\mathsf{a}^2 + \mathsf{a}^{(1,0)})\tfrac{h^2}{6} - \tfrac{\Delta\mathsf{a}h^3}{6}, \\
C_{0,0} &= -\tfrac{10}{3} - (3\mathsf{a}^2 + \mathsf{a}^{(0,1)} + \mathsf{a}^{(1,0)})\tfrac{h^2}{6} + \tfrac{\Delta\mathsf{a}h^3}{6}, & C_{0,1} &= \tfrac{2}{3} + \tfrac{\mathsf{a}h}{3} + (\mathsf{a}^2 + \mathsf{a}^{(1,0)})\tfrac{h^2}{6}, \\
C_{1,-1} &= C_{-1,1}, & C_{1,0} &= \tfrac{2}{3} + \tfrac{\mathsf{a}h}{3} + (\mathsf{a}^2 + \mathsf{a}^{(1,0)})\tfrac{h^2}{6}, & C_{1,1} &= \tfrac{1}{6} + \tfrac{\mathsf{a}h}{6},
\end{aligned}
\tag{2.50}
$$

and $F_{i,j}$ is obtained by replacing b by a in (2.49).

Note that the nine coefficients in $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ satisfying (2.43)–(2.44) with $M = 4$ are not unique. We choose each free variable $c_{k,\ell,p}$ to endow every $C_{k,\ell}$ with the simplest explicit expression in (2.47) and (2.50). The coefficients in (2.47) with $\mathsf{a} = \mathsf{b}$ can also give the left-hand side of a fourth-order FDM for $\Delta\mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{a}\mathsf{u}_y = \psi$, but the corresponding nine coefficients $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ are not of the simplest forms based on our technique in Section 2.2.

Summarizing the above relations (2.35)–(2.50), the following Theorems 2.1 and 2.2 provide the fourth-order compact 9-point FDMs for $\Delta\mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{a}\mathsf{u}_y = \psi$ and $\Delta\mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{b}\mathsf{u}_y = \psi$, respectively.

Theorem 2.1. *Assume $\kappa, u, \alpha = \beta, f$ are all smooth, and $\kappa_x = \kappa_y$ in (1.1), $\mathcal{L}_h u_h$ is defined in (2.35), the nine coefficients $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ are defined in (2.50), the right-hand side $F_{i,j}$ is obtained by replacing b by a in (2.49), and all the functions in $\mathcal{L}_h u_h$ are evaluated at the grid point (x_i, y_j) in (2.24). Then $h^{-2}\mathcal{L}_h u_h$ approximates $\Delta\mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{b}\mathsf{u}_y = \psi$ in (2.5) with the fourth-order of consistency at (x_i, y_j) for the case $\mathsf{a} = \mathsf{b}$, where $\psi, \mathsf{a}, \mathsf{b}, \mathsf{u}$ are defined in (2.1) and (2.6).*

Proof. From the derivations of $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ and $F_{i,j}$ in identities (2.35)–(2.50), this result can be proved directly. For the readers' convenience and to make our FDM more rigorous, we also provide the proof in the standard perspective as follows. From (2.5) with $\mathsf{a} = \mathsf{b}$ and (2.25), we have

$$\Delta\mathsf{u} + (\mathsf{u}^{(1,0)} + \mathsf{u}^{(0,1)})\mathsf{a} = \psi. \tag{2.51}$$

Taking the first-order partial derivatives of (2.51) with respect to x and y yields

$$\Delta u^{(1,0)} + (u^{(2,0)} + u^{(1,1)})a + (u^{(1,0)} + u^{(0,1)})a^{(1,0)} = \psi^{(1,0)}, \tag{2.52}$$

and

$$\Delta u^{(0,1)} + (u^{(1,1)} + u^{(0,2)})a + (u^{(1,0)} + u^{(0,1)})a^{(0,1)} = \psi^{(0,1)}, \tag{2.53}$$

so that (2.52)+(2.53) gives

$$\Delta(u^{(1,0)} + u^{(0,1)}) = -(\Delta u + 2u^{(1,1)})a - (u^{(1,0)} + u^{(0,1)})(a^{(1,0)} + a^{(0,1)}) + \psi^{(1,0)} + \psi^{(0,1)}. \tag{2.54}$$

Now, we take the first-order partial derivatives of (2.52) and (2.53) with respect to x and y, respectively,

$$u^{(4,0)} + u^{(2,2)} + (u^{(3,0)} + u^{(2,1)})a + 2(u^{(2,0)} + u^{(1,1)})a^{(1,0)} + (u^{(1,0)} + u^{(0,1)})a^{(2,0)} = \psi^{(2,0)}, \tag{2.55}$$

$$u^{(2,2)} + u^{(0,4)} + (u^{(1,2)} + u^{(0,3)})a + 2(u^{(1,1)} + u^{(0,2)})a^{(0,1)} + (u^{(1,0)} + u^{(0,1)})a^{(0,2)} = \psi^{(0,2)}, \tag{2.56}$$

hence (2.55)+(2.56) gives

$$\Delta^2 u + (u^{(1,0)} + u^{(0,1)})\Delta a + 2u^{(1,1)}(a^{(0,1)} + a^{(1,0)}) = -a\Delta(u^{(1,0)} + u^{(0,1)}) - 2u^{(2,0)}a^{(1,0)} \\ - 2u^{(0,2)}a^{(0,1)} + \Delta\psi. \tag{2.57}$$

Next, we plug $C_{k,\ell}$ of (2.50) into $\mathcal{L}_h u$ defined in (2.37), and apply (2.26) with $M = 5$. After a direct simplification, we obtain

$$\begin{aligned}
\mathcal{L}_h u &= \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} u(kh + x_i, \ell h + y_j) \\
&= h^2[\Delta u + a(u^{(1,0)} + u^{(0,1)})] + \tfrac{h^4}{12}[a^2\Delta u + 2a^2 u^{(1,1)} + 2a(\Delta u^{(1,0)} + \Delta u^{(0,1)}) \\
&\quad + \Delta^2 u + (u^{(0,1)} + u^{(1,0)})\Delta a + 2(a^{(0,1)} + a^{(1,0)})u^{(1,1)} + (u^{(2,0)} - u^{(0,2)})(a^{(1,0)} - a^{(0,1)})] + O(h^6).
\end{aligned}$$

Using (2.51), (2.54), and (2.57), after algebraic manipulation, we have

$$\begin{aligned}
\mathcal{L}_h u &= h^2\psi + \tfrac{h^4}{12}[a^2\Delta u + 2a^2 u^{(1,1)} - 2a^2\Delta u - 4a^2 u^{(1,1)} - 2a(u^{(1,0)} + u^{(0,1)})(a^{(1,0)} + a^{(0,1)}) \\
&\quad + 2a(\psi^{(1,0)} + \psi^{(0,1)}) - a\Delta(u^{(1,0)} + u^{(0,1)}) - 2u^{(2,0)}a^{(1,0)} - 2u^{(0,2)}a^{(0,1)} \\
&\quad + \Delta\psi + (u^{(2,0)} - u^{(0,2)})(a^{(1,0)} - a^{(0,1)})] + O(h^6), && \text{by (2.51), (2.54), (2.57),} \\
&= h^2\psi + \tfrac{h^4}{12}[-a^2\Delta u - 2a^2 u^{(1,1)} - 2a(u^{(1,0)} + u^{(0,1)})(a^{(1,0)} + a^{(0,1)}) + 2a(\psi^{(1,0)} + \psi^{(0,1)}) \\
&\quad + a^2\Delta u + 2a^2 u^{(1,1)} + a(u^{(1,0)} + u^{(0,1)})(a^{(1,0)} + a^{(0,1)}) - a(\psi^{(1,0)} + \psi^{(0,1)}) \\
&\quad - 2u^{(2,0)}a^{(1,0)} - 2u^{(0,2)}a^{(0,1)} + \Delta\psi + (u^{(2,0)} - u^{(0,2)})(a^{(1,0)} - a^{(0,1)})] + O(h^6), && \text{by (2.54),} \\
&= h^2\psi + \tfrac{h^4}{12}[-a(u^{(1,0)} + u^{(0,1)})(a^{(1,0)} + a^{(0,1)}) + a(\psi^{(1,0)} + \psi^{(0,1)}) \\
&\quad - 2u^{(2,0)}a^{(1,0)} - 2u^{(0,2)}a^{(0,1)} + \Delta\psi + (u^{(2,0)} - u^{(0,2)})(a^{(1,0)} - a^{(0,1)})] + O(h^6), && \text{by simplification,} \\
&= h^2\psi + \tfrac{h^4}{12}[-a(u^{(1,0)} + u^{(0,1)})(a^{(1,0)} + a^{(0,1)}) + a(\psi^{(1,0)} + \psi^{(0,1)}) \\
&\quad - u^{(2,0)}(a^{(1,0)} + a^{(0,1)}) - u^{(0,2)}(a^{(1,0)} + a^{(0,1)}) + \Delta\psi] + O(h^6), && \text{by simplification,} \\
&= h^2\psi + \tfrac{h^4}{12}[-a(u^{(1,0)} + u^{(0,1)})(a^{(1,0)} + a^{(0,1)}) + a(\psi^{(1,0)} + \psi^{(0,1)}) \\
&\quad + (a^{(1,0)} + a^{(0,1)})(a(u^{(1,0)} + u^{(0,1)}) - \psi) + \Delta\psi] + O(h^6), && \text{by (2.51),} \\
&= h^2\psi + \tfrac{h^4}{12}[a(\psi^{(1,0)} + \psi^{(0,1)}) - (a^{(1,0)} + a^{(0,1)})\psi + \Delta\psi] + O(h^6), && \text{by simplification.}
\end{aligned}$$

Finally, (2.35) and (2.49) with $b = a$ complete the argument. □

Theorem 2.2. *Assume $\kappa, u, \alpha, \beta, f$ are all smooth in (1.1), $\mathcal{L}_h u_h$ is defined in (2.35), the nine coefficients $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ are defined in (2.47)–(2.48), the right-hand side $F_{i,j}$ is defined in (2.49), and all the functions in $\mathcal{L}_h u_h$ are evaluated at the grid point (x_i, y_j) in (2.24). Then $h^{-2}\mathcal{L}_h u_h$ approximates $\Delta u + a u_x + b u_y = \psi$ in (2.5) with the fourth-order of consistency at (x_i, y_j), where ψ, a, b, u are defined in (2.1) and (2.6).*

Proof. The proof follows similar arguments as in Theorem 2.1. □

From (2.43)–(2.46), there exist the nine coefficients $\{C_{k,\ell} = \sum_{p=0}^{M+1} c_{k,\ell,p} h^p : k, \ell = -1, 0, 1\}$ fulfilling (2.43) with $M = 4$ for any free variables $c_{k,\ell,p}$ in (2.46). Hence, we use the free variables from (2.46) to reduce the pollution effects in the following Section 2.4.

2.4. Reduction of the pollution effects of $O(h^4)$ and $O(h^5)$

Recall that $M = 4$ is the largest integer such that the nontrivial $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ satisfying (2.43), i.e., the maximum consistency order of the FDM in Theorem 2.2 for $\Delta u + au_x + bu_y = \psi$ in (2.5), is 4. In this section we use the free variables $c_{k,\ell,p}$ in (2.46) to reduce the pollution effects of $O(h^4)$ and $O(h^5)$. Let us choose $M = 6$ in (2.36), (2.39), and (2.41), so that

$$C_{k,\ell} := \sum_{p=0}^{7} c_{k,\ell,p} h^p, \quad c_{k,\ell,p} \in S, \qquad I_{7,m,n} := \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} G_{7,m,n}(kh, \ell h), \tag{2.58}$$

$$F_{i,j} = \text{the terms of } \left(h^{-2} \sum_{(m,n) \in \Lambda_5} \psi^{(m,n)} \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} H_{7,m,n}(kh, \ell h) \right) \text{ with degree} \leq 5 \text{ in } h, \tag{2.59}$$

where $G_{7,m,n}(x, y)$ and $H_{7,m,n}(x, y)$ are defined in (2.34), and S is defined in (2.18). To maintain the fourth-order of consistency, and reduce the truncation errors of $O(h^4)$ and $O(h^5)$, we choose $M = 4$, and replace $I_{5,m,n}$ by $I_{7,m,n}$ and $G_{5,m,n}$ by $G_{7,m,n}$ in (2.43)–(2.44). Using the symbolic computation from Maple, we verify that there exist $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ with $k, \ell = -1, 0, 1$ satisfying

$$I_{7,m,n} = \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} G_{7,m,n}(kh, \ell h) = O(h^6) \quad \text{for all} \quad (m,n) \in \Lambda_5^1, \quad C_{0,0}|_{h=0} \neq 0, \quad F_{i,j}|_{h=0} = \psi, \tag{2.60}$$

for any free variables in the following set

$$\begin{aligned}
&\{c_{-1,-1,p}\}_{p=6,7} \cup \{c_{-1,0,p}\}_{p=5,6,7} \cup \{c_{-1,1,p}\}_{p=4,\dots,7} \cup \{c_{0,-1,p}\}_{p=5,6,7} \\
&\cup \{c_{0,0,p}\}_{p=4,\dots,7} \cup \{c_{0,1,p}\}_{p=3,\dots,7} \cup \{c_{1,-1,p}\}_{p=3,\dots,7} \cup \{c_{1,0,p}\}_{p=2,\dots,7} \cup \{c_{1,1,p}\}_{p=1,\dots,7}.
\end{aligned} \tag{2.61}$$

Furthermore, we also observe that $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ solving (2.60) leads to

$$\begin{aligned}
\sum_{(m,n) \in \Lambda_7^1} u^{(m,n)} I_{7,m,n} &= \sum_{(m,n) \in \Lambda_7^1} u^{(m,n)} \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} G_{7,m,n}(kh, \ell h) \\
&= \frac{h^6}{90} (a^{(0,1)} - b^{(1,0)}) u^{(1,3)} + C_6 h^6 + C_7 h^7 + O(h^8),
\end{aligned} \tag{2.62}$$

where C_6 and C_7 depend on the free variables in (2.61). To reduce the truncation errors of $O(h^4)$ and $O(h^5)$, we consider

$$\begin{aligned}
&\sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} G_{7,m,n}(kh, \ell h) = O(h^8) \quad \text{for all} \quad (m,n) \in \Lambda_7^1 \setminus \{(1,3)\}, \\
&\sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} G_{7,1,3}(kh, \ell h) - \frac{h^6}{90}(a^{(0,1)} - b^{(1,0)}) = O(h^8), \quad C_{0,0}|_{h=0} \neq 0, \quad F_{i,j}|_{h=0} = \psi.
\end{aligned} \tag{2.63}$$

Again using the symbolic calculation from Maple, there exist $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ with $k, \ell = -1, 0, 1$ satisfying (2.63) for any free variables in the following set:

$$\begin{aligned}
&\{c_{-1,0,7}\} \cup \{c_{-1,1,p}\}_{p=6,7} \cup \{c_{0,-1,7}\} \cup \{c_{0,0,p}\}_{p=6,7} \cup \{c_{0,1,p}\}_{p=5,6,7} \cup \{c_{1,-1,p}\}_{p=5,6,7} \\
&\cup \{c_{1,0,p}\}_{p=4,\dots,7} \cup \{c_{1,1,p}\}_{p=2,\dots,7}.
\end{aligned} \tag{2.64}$$

Furthermore, $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ fulfilling (2.63) yields

$$c_{-1,-1,0} = c_{-1,1,0} = c_{1,-1,0} = c_{1,1,0} = \tfrac{1}{6}, \qquad c_{-1,0,0} = c_{1,0,0} = c_{0,-1,0} = c_{0,1,0} = \tfrac{2}{3}, \qquad c_{0,0,0} = -\tfrac{10}{3}. \tag{2.65}$$

Similarly to (2.40)–(2.45), $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ meeting (2.63) implies

$$h^{-2} \mathcal{L}_h(u - u_h) = \frac{h^4}{90}(a^{(0,1)} - b^{(1,0)}) u^{(1,3)} + O(h^6), \tag{2.66}$$

and $\mathcal{L}_h u_h$ approximates $\Delta u + au_x + bu_y = \psi$ with the fourth-order of consistency at (x_i, y_j), where $\mathcal{L}_h u$ is defined in (2.37), $\mathcal{L}_h u_h$ is defined in (2.35), and $F_{i,j}$ is defined in (2.59). In our numerical examples, we set all free variables in (2.64) to zero, so $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ with $k, \ell = -1, 0, 1$ can be uniquely determined by solving (2.63). In summary, we have the FDM with the reduced pollution effects of $O(h^4)$ and $O(h^5)$ in the following theorem:

Theorem 2.3. *Assume $\kappa, u, \alpha, \beta, f$ are all smooth in* (1.1), *$\mathcal{L}_h \mathsf{u}_h$ is defined in* (2.35), *the nine coefficients $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ with $k, \ell = -1, 0, 1$ are uniquely determined by solving* (2.63) *with all free variables being zero in* (2.64), *the right-hand side $F_{i,j}$ is defined in* (2.59), *and all the functions in $\mathcal{L}_h \mathsf{u}_h$ are evaluated at the grid point (x_i, y_j) in* (2.24). *Then $h^{-2}\mathcal{L}_h \mathsf{u}_h$ approximates $\Delta \mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{b}\mathsf{u}_y = \psi$ with the fourth-order of consistency at (x_i, y_j), and the truncation error $\frac{h^4}{90}(\mathsf{a}^{(0,1)} - \mathsf{b}^{(1,0)})\mathsf{u}^{(1,3)} + O(h^6)$.*

Proof. The proof follows directly from (2.58)–(2.66). □

We note that the theoretical results of the discrete maximum principle in [12, 13, 14, 19, 20] imply that, if the nine coefficients $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ satisfy

$$C_{0,0} > 0, \qquad C_{k,\ell} \leq 0, \quad \text{if} \quad (k, \ell) \neq (0, 0), \quad \text{the sign condition;}$$

$$\sum_{k=-1}^{1} \sum_{\ell=-1}^{1} C_{k,\ell} \geq 0, \quad \text{the sum condition;} \tag{2.67}$$

then the corresponding FDM with the Dirichlet boundary condition yields a numerical solution which satisfies the discrete maximum principle, and generates an M-matrix (a real square matrix with the non-positive off-diagonal entries and the positive diagonal entries such that all row sums are non-negative with at least one row sum being positive).

Proposition 2.4. *All $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ in Theorems 2.1 to 2.3 with $C_{k,\ell}|_{h=0} = c_{k,\ell,0}$ satisfy*

$$c_{\pm 1,\pm 1,0} = c_{\pm 1,\mp 1,0} = \tfrac{1}{6}, \qquad c_{\pm 1,0,0} = c_{0,\pm 1,0} = \tfrac{2}{3}, \qquad c_{0,0,0} = -\tfrac{10}{3}, \qquad \sum_{k=1}^{-1} \sum_{\ell=1}^{-1} c_{k,\ell,0} = 0. \tag{2.68}$$

That is, each FDM in Theorems 2.1 to 2.3 satisfies the discrete maximum principle, and generates an M-matrix, when the mesh size h is sufficiently small.

Proof. The expression (2.68) stems from (2.47), (2.50), and (2.65). By $C_{k,\ell}|_{h=0} = c_{k,\ell,0}$ and (2.35), we see that

$$-h^{-2}\mathcal{L}_h := h^{-2} \sum_{k=-1}^{1} \sum_{\ell=-1}^{1} (-C_{k,\ell})(\mathsf{u}_h)_{i+k,j+\ell} = -F_{i,j}, \tag{2.69}$$

generates an M-matrix, and the numerical solution computed by (2.69) satisfies the discrete maximum principle, if h is sufficiently small. □

Now, we use the fourth-order compact 9-point FDMs in Theorems 2.1 to 2.3, and the iteration method (2.3), to numerically solve (1.1) in the following Algorithm 1. Recall that u denotes the exact solution of (1.1). Now, we define u_h as the numerical solution computed by the FDMs in Theorems 2.1 to 2.3 and the iteration method (2.3).

Algorithm 1. *We use the fourth-order compact 9-point FDMs in Theorems 2.1 to 2.3 to solve* (2.5) *to obtain u_h with the iteration method* (2.3) *after 40 iterations. Then $u_h = \mathsf{u}_h$.*

In Algorithm 1, we consider $u_0 = 0$ in (2.3) as the initial guess in the iteration method, and we observe that 40 iterations are sufficient for convergence. For the time-dependent nonlinear convection-diffusion equation (1.2) in Section 3, we use $u_0^{n+1/2} = u^n$ in (3.5) for the CN method, $u_0^{n+3} = u^{n+2}$ in (3.10) for the BDF3 method, and $u_0^{n+4} = u^{n+3}$ in (3.15) for the BDF4 method, so we only need 20 iterations in Algorithms 2 to 4.

2.5. Approximations of the high-order partial derivatives of $\mathsf{a}^{(m,n)}$, $\mathsf{b}^{(m,n)}$, and $\psi^{(m,n)}$

For the explicit expressions of $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ and $F_{i,j}$ in (2.47), (2.49), and (2.50) of Theorems 2.1 and 2.2, the high-order partial derivatives $\{\mathsf{a}^{(m,n)}, \mathsf{b}^{(m,n)}, \psi^{(m,n)} : m + n \leq 2\}$ are required. Furthermore, the symbolic computation in Maple also reveals that the unique $\{C_{k,\ell}\}_{k,\ell=-1,0,1}$ and $F_{i,j}$ of the FDM in Theorem 2.3 need $\{\mathsf{a}^{(m,n)}, \mathsf{b}^{(m,n)} : m + n \leq 4\}$ and $\{\psi^{(m,n)} : m + n \leq 5\}$. Since κ and f are available in (1.1), the expressions of (2.1) and (2.6) imply that, we only need to approximate

$$\left(\frac{\alpha_u(u)}{\kappa}\right)^{(m,n)} \quad \text{and} \quad \left(\frac{\beta_u(u)}{\kappa}\right)^{(m,n)}, \quad \text{with} \quad m + n \leq 4. \tag{2.70}$$

So we use the following FDMs to evaluate (2.70). For any smooth 1D function $\rho(x) \in C^5(\mathbb{R})$,
The first-order derivatives:

$$\rho_x(x_i) = \tfrac{1}{h}\left[\tfrac{1}{20}\rho(x_{i-2}) - \tfrac{1}{2}\rho(x_{i-1}) - \tfrac{1}{3}\rho(x_i) + \rho(x_{i+1}) - \tfrac{1}{4}\rho(x_{i+2}) + \tfrac{1}{30}\rho(x_{i+3})\right] + O(h^5),$$

11

$$\text{if} \quad 0 \le x_{i\pm 2}, x_{i+3} \le 1,$$

$$\rho_x(x_i) = \frac{1}{h}\left[-\frac{1}{30}\rho(x_{i-3}) + \frac{1}{4}\rho(x_{i-2}) - \rho(x_{i-1}) + \frac{1}{3}\rho(x_i) + \frac{1}{2}\rho(x_{i+1}) - \frac{1}{20}\rho(x_{i+2})\right] + O(h^5),$$

$$\text{if} \quad 0 \le x_{i\pm 2}, x_{i-3} \le 1,$$

$$\rho_x(x_i) = \frac{\pm 1}{h}\left[\frac{1}{5}\rho(x_{i\pm 5}) - \frac{5}{4}\rho(x_{i\pm 4}) + \frac{10}{3}\rho(x_{i\pm 3}) - 5\rho(x_{i\pm 2}) + 5\rho(x_{i\pm 1}) - \frac{137}{60}\rho(x_i)\right] + O(h^5),$$

$$\text{if} \quad 0 \le x_{i\pm 5} \text{ and } x_i \le 1,$$

$$\rho_x(x_i) = \frac{\pm 1}{h}\left[-\frac{1}{20}\rho(x_{i\pm 4}) + \frac{1}{3}\rho(x_{i\pm 3}) - \rho(x_{i\pm 2}) + 2\rho(x_{i\pm 1}) - \frac{13}{12}\rho(x_i) - \frac{1}{5}\rho(x_{i\mp 1})\right] + O(h^5),$$

$$\text{if} \quad 0 \le x_{i\pm 4} \text{ and } x_{i\mp 1} \le 1.$$

The second-order derivatives:

$$\rho_{xx}(x_i) = \frac{1}{h^2}\left[-\frac{1}{12}\rho(x_{i-2}) + \frac{4}{3}\rho(x_{i-1}) - \frac{5}{2}\rho(x_i) + \frac{4}{3}\rho(x_{i+1}) - \frac{1}{12}\rho(x_{i+2})\right] + O(h^4),$$

$$\text{if} \quad 0 \le x_{i\pm 2} \le 1,$$

$$\rho_{xx}(x_i) = \frac{1}{h^2}\left[-\frac{5}{6}\rho(x_{i\pm 5}) + \frac{61}{12}\rho(x_{i\pm 4}) - 13\rho(x_{i\pm 3}) + \frac{107}{6}\rho(x_{i\pm 2}) - \frac{77}{6}\rho(x_{i\pm 1}) + \frac{15}{4}\rho(x_i)\right] + O(h^4),$$

$$\text{if} \quad 0 \le x_{i\pm 5} \text{ and } x_i \le 1,$$

$$\rho_{xx}(x_i) = \frac{1}{h^2}\left[\frac{1}{12}\rho(x_{i\pm 4}) - \frac{1}{2}\rho(x_{i\pm 3}) + \frac{7}{6}\rho(x_{i\pm 2}) - \frac{1}{3}\rho(x_{i\pm 1}) - \frac{5}{4}\rho(x_i) + \frac{5}{6}\rho(x_{i\mp 1})\right] + O(h^4),$$

$$\text{if} \quad 0 \le x_{i\pm 4} \text{ and } x_{i\mp 1} \le 1.$$

The third-order derivatives:

$$\rho_{xxx}(x_i) = \frac{1}{h^3}\left[-\frac{1}{4}\rho(x_{i-2}) - \frac{1}{4}\rho(x_{i-1}) + \frac{5}{2}\rho(x_i) - \frac{7}{2}\rho(x_{i+1}) + \frac{7}{4}\rho(x_{i+2}) - \frac{1}{4}\rho(x_{i+3})\right] + O(h^3),$$

$$\text{if} \quad 0 \le x_{i\pm 2}, x_{i+3} \le 1,$$

$$\rho_{xxx}(x_i) = \frac{1}{h^3}\left[\frac{1}{4}\rho(x_{i-3}) - \frac{7}{4}\rho(x_{i-2}) + \frac{7}{2}\rho(x_{i-1}) - \frac{5}{2}\rho(x_i) + \frac{1}{4}\rho(x_{i+1}) + \frac{1}{4}\rho(x_{i+2})\right] + O(h^3),$$

$$\text{if} \quad 0 \le x_{i\pm 2}, x_{i-3} \le 1,$$

$$\rho_{xxx}(x_i) = \frac{\pm 1}{h^3}\left[\frac{7}{4}\rho(x_{i\pm 5}) - \frac{41}{4}\rho(x_{i\pm 4}) + \frac{49}{2}\rho(x_{i\pm 3}) - \frac{59}{2}\rho(x_{i\pm 2}) + \frac{71}{4}\rho(x_{i\pm 1}) - \frac{17}{4}\rho(x_i)\right] + O(h^3),$$

$$\text{if} \quad 0 \le x_{i\pm 5} \text{ and } x_i \le 1,$$

$$\rho_{xxx}(x_i) = \frac{\pm 1}{h^3}\left[\frac{1}{4}\rho(x_{i\pm 4}) - \frac{7}{4}\rho(x_{i\pm 3}) + \frac{11}{2}\rho(x_{i\pm 2}) - \frac{17}{2}\rho(x_{i\pm 1}) + \frac{25}{4}\rho(x_i) - \frac{7}{4}\rho(x_{i\mp 1})\right] + O(h^3),$$

$$\text{if} \quad 0 \le x_{i\pm 4} \text{ and } x_{i\mp 1} \le 1.$$

The fourth-order derivatives:

$$\rho_{xxxx}(x_i) = \frac{1}{h^4}\left[\rho(x_{i-2}) - 4\rho(x_{i-1}) + 6\rho(x_i) - 4\rho(x_{i+1}) + \rho(x_{i+2})\right] + O(h^2),$$

$$\text{if} \quad 0 \le x_{i\pm 2} \le 1,$$

$$\rho_{xxxx}(x_i) = \frac{1}{h^4}\left[-2\rho(x_{i\pm 5}) + 11\rho(x_{i\pm 4}) - 24\rho(x_{i\pm 3}) + 26\rho(x_{i\pm 2}) - 14\rho(x_{i\pm 1}) + 3\rho(x_i)\right] + O(h^2),$$

$$\text{if} \quad 0 \le x_{i\pm 5} \text{ and } x_i \le 1,$$

$$\rho_{xxxx}(x_i) = \frac{1}{h^4}\left[-\rho(x_{i\pm 4}) + 6\rho(x_{i\pm 3}) - 14\rho(x_{i\pm 2}) + 16\rho(x_{i\pm 1}) - 9\rho(x_i) + 2\rho(x_{i\mp 1})\right] + O(h^2),$$

$$\text{if} \quad 0 \le x_{i\pm 4} \text{ and } x_{i\mp 1} \le 1.$$

The fifth-order derivatives:

$$\rho_{xxxxx}(x_i) = \frac{1}{h^5}\left[-\rho(x_{i-2}) + 5\rho(x_{i-1}) - 10\rho(x_i) + 10\rho(x_{i+1}) - 5\rho(x_{i+2}) + \rho(x_{i+3})\right] + O(h),$$

$$\text{if} \quad 0 \le x_{i\pm 2}, x_{i+3} \le 1,$$

$$\rho_{xxxxx}(x_i) = \frac{1}{h^5}\left[-\rho(x_{i-3}) + 5\rho(x_{i-2}) - 10\rho(x_{i-1}) + 10\rho(x_i) - 5\rho(x_{i+1}) + \rho(x_{i+2})\right] + O(h),$$

$$\text{if} \quad 0 \le x_{i\pm 2}, x_{i-3} \le 1,$$

$$\rho_{xxxxx}(x_i) = \frac{\pm 1}{h^5}\left[\rho(x_{i\pm 5}) - 5\rho(x_{i\pm 4}) + 10\rho(x_{i\pm 3}) - 10\rho(x_{i\pm 2}) + 5\rho(x_{i\pm 1}) - \rho(x_i)\right] + O(h),$$

$$\text{if} \quad 0 \le x_{i\pm 5} \text{ and } x_i \le 1,$$

$$\rho_{xxxxx}(x_i) = \frac{\pm 1}{h^5}\left[\rho(x_{i\pm 4}) - 5\rho(x_{i\pm 3}) + 10\rho(x_{i\pm 2}) - 10\rho(x_{i\pm 1}) + 5\rho(x_i) - \rho(x_{i\mp 1})\right] + O(h),$$

$$\text{if} \quad 0 \le x_{i\pm 4} \text{ and } x_{i\mp 1} \le 1.$$

Similarly, for any smooth 2D function $\rho(x,y) \in C^5(\mathbb{R}^2)$, we can evaluate $\rho_x(x_i,y_j), \rho_{xx}(x_i,y_j), \rho_{xxx}(x_i,y_j), \rho_{xxxx}(x_i,y_j), \rho_{xxxxx}(x_i,y_j), \rho_y(x_i,y_j), \rho_{yy}(x_i,y_j), \rho_{yyy}(x_i,y_j), \rho_{yyyy}(x_i,y_j), \rho_{yyyyy}(x_i,y_j)$. Then we can evaluate the high-order mixed derivatives as follows:

$$\rho_{xy} = (\rho_x)_y, \quad \rho_{xxy} = (\rho_{xx})_y, \quad \rho_{xyy} = (\rho_x)_{yy}, \quad \rho_{xxxy} = (\rho_{xxx})_y, \quad \rho_{xxyy} = (\rho_{xx})_{yy},$$

$$\rho_{xyyy} = (\rho_x)_{yyy}, \quad \rho_{xxxxy} = (\rho_{xxxx})_y, \quad \rho_{xxxyy} = (\rho_{xxx})_{yy}, \quad \rho_{xxyyy} = (\rho_{xx})_{yyy}, \quad \rho_{xyyyy} = (\rho_x)_{yyyy}.$$

Note that the fifth-order partial derivatives are not required in this section, but they are necessary for the FDMs in Section 3.

3. Second- to fourth-order compact 9-point FDMs for the time-dependent nonlinear convection-diffusion equation

Similarly to Section 2, we rewrite the time-dependent nonlinear convection-diffusion equation (1.2) as the linear problem via a fixed point method in Section 3.1, and the compact 9-point FDMs are constructed in Section 3.2 to solve the reformulated linear problem.

3.1. Reformulation of the time-dependent nonlinear convection-diffusion equation

(1.2) leads to

$$u_t - \kappa\Delta u - \kappa_x u_x - \kappa_y u_y + \alpha_u(u)u_x + \beta_u(u)u_y = f. \tag{3.1}$$

Recall that temporal domain $I = [0,1]$, and u is the exact solution of (1.2) and (3.1). Here, we define that

$$u^n := u|_{t=t_n}, \qquad t_n := n\tau, \qquad n = 0,\ldots,N_2, \qquad \tau := 1/N_2, \qquad N_2 \in \mathbb{N}. \tag{3.2}$$

The second-order Crank-Nicolson method: (1.2) and (3.1) with the CN method in [2] imply

$$\begin{cases} \dfrac{u^{n+1/2} - u^n}{\tau/2} - \kappa^{n+1/2}\Delta u^{n+1/2} + [\alpha_u(u^{n+1/2}) - \kappa_x^{n+1/2}]u_x^{n+1/2} + [\beta_u(u^{n+1/2}) - \kappa_y^{n+1/2}]u_y^{n+1/2} = f^{n+1/2}, \\ u^{n+1} = 2u^{n+1/2} - u^n, \quad \text{with the initial } n = 0, \text{ the given } u^0 \in \Omega, \ u^{n+1/2} \in \partial\Omega. \end{cases} \tag{3.3}$$

The first identity in (3.3) yields

$$\Delta u^{n+1/2} + \frac{\kappa_x^{n+1/2} - \alpha_u(u^{n+1/2})}{\kappa^{n+1/2}}u_x^{n+1/2} + \frac{\kappa_y^{n+1/2} - \beta_u(u^{n+1/2})}{\kappa^{n+1/2}}u_y^{n+1/2} - \frac{2u^{n+1/2}}{\tau\kappa^{n+1/2}}$$
$$= \frac{-1}{\kappa^{n+1/2}}\left[f^{n+1/2} + \frac{2}{\tau}u^n\right], \quad \text{with the initial } n = 0, \text{ the given } u^0 \in \Omega, \ u^{n+1/2} \in \partial\Omega. \tag{3.4}$$

We denote the solution at $t = (n + 1/2)\tau$ in the k-iteration by $u_k^{n+1/2}$. Then we use the following iteration method to rewrite (3.3) as the linear convection-diffusion equation:

$$\begin{cases} \Delta u_{k+1}^{n+1/2} + \dfrac{\kappa_x^{n+1/2} - \alpha_u(u_k^{n+1/2})}{\kappa^{n+1/2}}(u_{k+1}^{n+1/2})_x + \dfrac{\kappa_y^{n+1/2} - \beta_u(u_k^{n+1/2})}{\kappa^{n+1/2}}(u_{k+1}^{n+1/2})_y - \dfrac{2u_{k+1}^{n+1/2}}{\tau\kappa^{n+1/2}} \\ = \dfrac{-1}{\kappa^{n+1/2}}\left[f^{n+1/2} + \dfrac{2}{\tau}u^n\right], \\ k := k + 1, \text{ the initial } k = n = 0, \text{ the given } u^0 \in \Omega, \ u_{k+1}^{n+1/2} \in \partial\Omega, \ u_0^{n+1/2} = u^n \text{ in } \Omega, \\ u^{n+1} = 2u^{n+1/2} - u^n, \end{cases} \tag{3.5}$$

where u^n with $n \geq 1$ is calculated at $t = n\tau$ by the same iteration method. Let

$$\tau := rh, \qquad \mathsf{u} := u_{k+1}^{n+1/2}, \qquad \mathsf{a} := \frac{\kappa_x^{n+1/2} - \alpha_u(u_k^{n+1/2})}{\kappa^{n+1/2}}, \qquad \mathsf{b} := \frac{\kappa_y^{n+1/2} - \beta_u(u_k^{n+1/2})}{\kappa^{n+1/2}},$$
$$\mathsf{c} := -\frac{2}{r\kappa^{n+1/2}}, \qquad \mathsf{d} := \frac{\mathsf{c}}{h}, \qquad \varphi := -\frac{f^{n+1/2}}{\kappa^{n+1/2}}, \qquad \phi := -\frac{2u^n}{r\kappa^{n+1/2}}, \qquad \psi := \varphi + \frac{\phi}{h}, \tag{3.6}$$

where r is a positive constant. Then (1.2) and (3.5) indicate

$$\Delta\mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{b}\mathsf{u}_y + \mathsf{d}\mathsf{u} = \psi \text{ in } \Omega \quad \text{and} \quad \mathsf{u} = g \text{ on } \partial\Omega. \tag{3.7}$$

The third-order backward difference formula: (1.2) and (3.1) with the BDF3 method in [15, p.366] give

$$\frac{11u^{n+3} - 18u^{n+2} + 9u^{n+1} - 2u^n}{6\tau} - \kappa^{n+3}\Delta u^{n+3} + (\alpha_u(u^{n+3}) - \kappa_x^{n+3})u_x^{n+3}$$
$$+ (\beta_u(u^{n+3}) - \kappa_y^{n+3})u_y^{n+3} = f^{n+3}, \quad \text{with the given } u^0 \in \Omega \text{ and } u^{n+3} \in \partial\Omega. \tag{3.8}$$

13

After a direct calculation, we rewrite

$$
\Delta u^{n+3} + \frac{\kappa_x^{n+3} - \alpha_u(u^{n+3})}{\kappa^{n+3}} u_x^{n+3} + \frac{\kappa_y^{n+3} - \beta_u(u^{n+3})}{\kappa^{n+3}} u_y^{n+3} - \frac{11 u^{n+3}}{6\tau\kappa^{n+3}}
$$
$$
= \frac{-1}{\kappa^{n+3}} \left[f^{n+3} + \frac{3}{\tau} u^{n+2} - \frac{3}{2\tau} u^{n+1} + \frac{1}{3\tau} u^{n} \right], \quad \text{with the given } u^0 \in \Omega \text{ and } u^{n+3} \in \partial\Omega. \tag{3.9}
$$

Similarly to (3.5), the above nonlinear equation (3.9) is solved by the following linear convection-diffusion equation, via the iteration method:

$$
\begin{cases}
\Delta u_{k+1}^{n+3} + \dfrac{\kappa_x^{n+3} - \alpha_u(u_k^{n+3})}{\kappa^{n+3}} (u_{k+1}^{n+3})_x + \dfrac{\kappa_y^{n+3} - \beta_u(u_k^{n+3})}{\kappa^{n+3}} (u_{k+1}^{n+3})_y - \dfrac{11 u_{k+1}^{n+3}}{6\tau\kappa^{n+3}} \\[2mm]
= \dfrac{-1}{\kappa^{n+3}} \left[f^{n+3} + \dfrac{3}{\tau} u^{n+2} - \dfrac{3}{2\tau} u^{n+1} + \dfrac{1}{3\tau} u^{n} \right], \\[2mm]
k := k + 1, \text{ the initial } k = n = 0, \text{ the given } u^0 \in \Omega, \ u_{k+1}^{n+3} \in \partial\Omega, \text{ and } u_0^{n+3} = u^{n+2} \text{ in } \Omega,
\end{cases} \tag{3.10}
$$

where u_k^{n+3} with $n \geq 0$ is computed in the k-iteration at $t = (n + 3)\tau$; u^n, u^{n+1}, and u^{n+2} with $n \geq 3$ are calculated at $t = n\tau, (n + 1)\tau$, and $(n + 2)\tau$ by the same iteration method; u^1 and u^2 are computed by the CN method. Let

$$
\tau := rh, \qquad u := u_{k+1}^{n+3}, \qquad a := \frac{\kappa_x^{n+3} - \alpha_u(u_k^{n+3})}{\kappa^{n+3}}, \qquad b := \frac{\kappa_y^{n+3} - \beta_u(u_k^{n+3})}{\kappa^{n+3}}, \qquad c := -\frac{11}{6r\kappa^{n+3}},
$$
$$
d := \frac{c}{h}, \qquad \psi := \varphi + \frac{\phi}{h}, \qquad \varphi := -\frac{f^{n+3}}{\kappa^{n+3}}, \qquad \phi := \frac{-1}{\kappa^{n+3}} \left[\frac{3}{r} u^{n+2} - \frac{3}{2r} u^{n+1} + \frac{1}{3r} u^{n} \right], \tag{3.11}
$$

where r is a positive constant. Then

$$
\Delta u + a u_x + b u_y + d u = \psi \text{ in } \Omega \quad \text{and} \quad u = g \text{ on } \partial\Omega. \tag{3.12}
$$

The fourth-order backward difference formula: (1.2) and (3.1) with the BDF4 method in [15, p.366] generate

$$
\frac{25 u^{n+4} - 48 u^{n+3} + 36 u^{n+2} - 16 u^{n+1} + 3 u^{n}}{12\tau} - \kappa^{n+4} \Delta u^{n+4} + (\alpha_u(u^{n+4}) - \kappa_x^{n+4}) u_x^{n+4}
$$
$$
+ (\beta_u(u^{n+4}) - \kappa_y^{n+4}) u_y^{n+4} = f^{n+4}, \quad \text{with the given } u^0 \in \Omega \text{ and } u^{n+4} \in \partial\Omega. \tag{3.13}
$$

Then,

$$
\Delta u^{n+4} + \frac{\kappa_x^{n+4} - \alpha_u(u^{n+4})}{\kappa^{n+4}} u_x^{n+4} + \frac{\kappa_y^{n+4} - \beta_u(u^{n+4})}{\kappa^{n+4}} u_y^{n+4} - \frac{25 u^{n+4}}{12\tau\kappa^{n+4}}
$$
$$
= \frac{-1}{\kappa^{n+4}} \left[f^{n+4} + \frac{4}{\tau} u^{n+3} - \frac{3}{\tau} u^{n+2} + \frac{4}{3\tau} u^{n+1} - \frac{1}{4\tau} u^{n} \right], \quad \text{with given } u^0 \in \Omega \text{ and } u^{n+4} \in \partial\Omega. \tag{3.14}
$$

Similarly to (3.10), we linearize (3.14) as

$$
\begin{cases}
\Delta u_{k+1}^{n+4} + \dfrac{\kappa_x^{n+4} - \alpha_u(u_k^{n+4})}{\kappa^{n+4}} (u_{k+1}^{n+4})_x + \dfrac{\kappa_y^{n+4} - \beta_u(u_k^{n+4})}{\kappa^{n+4}} (u_{k+1}^{n+4})_y - \dfrac{25 u_{k+1}^{n+4}}{12\tau\kappa^{n+4}} \\[2mm]
= \dfrac{-1}{\kappa^{n+4}} \left[f^{n+4} + \dfrac{4}{\tau} u^{n+3} - \dfrac{3}{\tau} u^{n+2} + \dfrac{4}{3\tau} u^{n+1} - \dfrac{1}{4\tau} u^{n} \right], \\[2mm]
k := k + 1, \text{ the initial } k = n = 0, \text{ the given } u^0 \in \Omega, \ u_{k+1}^{n+4} \in \partial\Omega, \text{ and } u_0^{n+4} = u^{n+3} \text{ in } \Omega,
\end{cases} \tag{3.15}
$$

where u^1 and u^2 are computed by the CN method, and u^3 is computed by the BDF3 method. Let

$$
\tau := rh, \qquad u := u_{k+1}^{n+4}, \qquad a := \frac{\kappa_x^{n+4} - \alpha_u(u_k^{n+4})}{\kappa^{n+4}}, \qquad b := \frac{\kappa_y^{n+4} - \beta_u(u_k^{n+4})}{\kappa^{n+4}}, \qquad c := -\frac{25}{12r\kappa^{n+4}},
$$
$$
d := \frac{c}{h}, \qquad \psi := \varphi + \frac{\phi}{h}, \qquad \varphi := -\frac{f^{n+4}}{\kappa^{n+4}}, \qquad \phi := -\frac{1}{\kappa^{n+4}} \left[\frac{4}{r} u^{n+3} - \frac{3}{r} u^{n+2} + \frac{4}{3r} u^{n+1} - \frac{1}{4r} u^{n} \right], \tag{3.16}
$$

where r is a positive constant. Then

$$
\Delta u + a u_x + b u_y + d u = \psi \text{ in } \Omega \quad \text{and} \quad u = g \text{ on } \partial\Omega. \tag{3.17}
$$

So, the CN, BDF3, and BDF4 methods with the iteration methods (3.5), (3.10), and (3.15) yield the same linear convection-diffusion equation in (3.7), (3.12), and (3.17).

Next, we propose the fourth-order compact 9-point FDM for the linear convection-diffusion equation in the following Section 3.2.

14

3.2. Fourth-order compact 9-point FDM

Since the CN, BDF3, and BDF4 methods yield the same linear convection-diffusion equation (3.7), (3.12), and (3.17), in this section we construct the fourth-order compact 9-point FDM for

$$\Delta u + a u_x + b u_y + d u = \psi, \tag{3.18}$$

where a, b, d, ψ are defined in (3.6), (3.11), and (3.16). Similarly to (2.10)–(2.17),

$$u^{(p,q)} = \sum_{(m,n)\in\Lambda^1_{p+q}} \xi_{p,q,m,n} u^{(m,n)} + \sum_{(m,n)\in\Lambda_{p+q-2}} \eta_{p,q,m,n}\psi^{(m,n)}, \qquad (p,q) \in \Lambda^2_M, \tag{3.19}$$

where $\xi_{p,q,m,n}$ and $\eta_{p,q,m,n}$ are uniquely determined by the high-order partial derivatives of a, b, and d. Similarly to (2.18)–(2.34),

$$u(x + x_i, y + y_j) = \sum_{(m,n)\in\Lambda^1_M} u^{(m,n)} G_{M,m,n}(x,y) + \sum_{(m,n)\in\Lambda_{M-2}} \psi^{(m,n)} H_{M,m,n}(x,y) + O(h^{M+1}), \tag{3.20}$$

where $x, y \in [-h, h]$, and

$$G_{M,m,n}(x,y) := \frac{x^m y^n}{m!n!} + \sum_{(p,q)\in\Lambda^2_M \setminus \Lambda^2_{m+n-1}} \xi_{p,q,m,n} \frac{x^p y^q}{p!q!}, \quad H_{M,m,n}(x,y) := \sum_{(p,q)\in\Lambda^2_M \setminus \Lambda^2_{m+n+1}} \eta_{p,q,m,n} \frac{x^p y^q}{p!q!}, \tag{3.21}$$

each $\xi_{p,q,m,n}, \eta_{p,q,m,n}$ in (3.19) and (3.21) belongs to

$$S := \text{span}\left\{ \prod_{i_1,j_1,v_1,w_1,r_1,s_1\in\mathbb{N}_0} a^{(i_1,j_1)} b^{(v_1,w_1)} d^{(r_1,s_1)}, \dots, \prod_{i_k,j_k,v_k,w_k,r_k,s_k\in\mathbb{N}_0} a^{(i_k,j_k)} b^{(v_k,w_k)} d^{(r_k,s_k)} \right\}, \tag{3.22}$$

with $k \in \mathbb{N}$, and the coefficients in $\mathbb{R}$. Similarly to (2.35)–(2.37), we define the linear operator $\mathcal{L}_h u$ and the stencil $\mathcal{L}_h u_h$ of the FDM as follows:

$$h^{-2}\mathcal{L}_h u := h^{-2} \sum_{k,\ell=-1}^{1} C_{k,\ell} u(kh + x_i, \ell h + y_j), \quad h^{-2}\mathcal{L}_h u_h := h^{-2} \sum_{k,\ell=-1}^{1} C_{k,\ell}(u_h)_{i+k,j+\ell} = F_{i,j}, \tag{3.23}$$

$$C_{k,\ell} := \sum_{p=0}^{7} c_{k,\ell,p} h^p, \quad c_{k,\ell,p} \in \tilde{S}, \tag{3.24}$$

where

$$\tilde{S} := \text{span}\left\{ \prod_{i_1,j_1,v_1,w_1,r_1,s_1\in\mathbb{N}_0} a^{(i_1,j_1)} b^{(v_1,w_1)} c^{(r_1,s_1)}, \dots, \prod_{i_k,j_k,v_k,w_k,r_k,s_k\in\mathbb{N}_0} a^{(i_k,j_k)} b^{(v_k,w_k)} c^{(r_k,s_k)} \right\}, \tag{3.25}$$

a, b, and c are defined in (3.6), (3.11), and (3.16). We restrict each $c_{k,\ell,p} \in \tilde{S}$ in (3.24) such that the high-order partial derivatives $a^{(m,n)}$, $b^{(m,n)}$, and $c^{(m,n)}$ do not appear in the denominator, ensuring that each $c_{k,\ell,p}$ is well defined. Similarly to (2.59),

$$F_{i,j} = \text{the terms of } \left(h^{-2} \sum_{(m,n)\in\Lambda_5} \psi^{(m,n)} \sum_{k,\ell=-1}^{1} C_{k,\ell} H_{7,m,n}(kh, \ell h) \right) \text{ with degree} \le 5 \text{ in } h, \tag{3.26}$$

where $H_{7,m,n}(x, y)$ is defined in (3.21), and ψ is defined in (3.6), (3.11), and (3.16). Similarly to (2.35)–(2.45) and (2.58)–(2.65), by the symbolic calculation in Maple, there exist $\{c_{k,\ell,p}\}_{k,\ell=-1,0,1}^{p=0,7}$ with $c_{k,\ell,p} \in \tilde{S}$ satisfying

$$\sum_{(m,n)\in\Lambda^1_7} u^{(m,n)} \sum_{k,\ell=-1}^{1} C_{k,\ell} G_{7,m,n}(kh, \ell h) - \frac{h^6}{90}(a^{(0,1)} - b^{(1,0)})u^{(1,3)} - h^7 \zeta = O(h^8),$$

$$c_{0,0,0} = -\frac{10}{3}, \qquad F_{i,j}|_{h=0} = \psi, \qquad c_{-1,-1,0} = c_{-1,1,0} = c_{1,-1,0} = c_{1,1,0} = \frac{1}{6}, \tag{3.27}$$

$$c_{-1,0,0} = c_{1,0,0} = c_{0,-1,0} = c_{0,1,0} = \frac{2}{3},$$

for any free variables in the following set:

$$\{c_{-1,0,7}\} \cup \{c_{-1,1,p}\}_{p=6,7} \cup \{c_{0,-1,7}\} \cup \{c_{0,0,p}\}_{p=6,7}$$

$$\cup \{c_{0,1,p}\}_{p=5,6,7} \cup \{c_{1,-1,p}\}_{p=5,6,7} \cup \{c_{1,0,p}\}_{p=4,\dots,7} \cup \{c_{1,1,p}\}_{p=1,\dots,7}, \tag{3.28}$$

where

$$
\begin{aligned}
\zeta :=& \tfrac{1}{37800}(10\mathsf{a}(6\mathsf{bc} - 21(\mathsf{a}^{(0,1)} - \mathsf{b}^{(1,0)}) - 8\mathsf{c}^{(0,1)}) - 210(\mathsf{a}^{(0,1)} - \mathsf{b}^{(1,0)} + \mathsf{c}^{(1,0)})\mathsf{b} \\
& - (49\mathsf{a}^{(0,1)} + 91\mathsf{b}^{(1,0)})\mathsf{c} + 2520c_{1,1,1}(\mathsf{a}^{(0,1)} - \mathsf{b}^{(1,0)}))u^{(1,3)} - \tfrac{1}{7560}(\mathsf{ac} + 14\mathsf{c}^{(1,0)})u^{(1,4)} \\
& + \tfrac{1}{540}(\mathsf{bc} - \mathsf{c}^{(0,1)})u^{(0,5)}.
\end{aligned}
\tag{3.29}
$$

Similarly to (2.66), $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ with $k, \ell = -1, 0, 1$ satisfying (3.27) implies

$$
h^{-2}\mathcal{L}_h(\mathsf{u} - \mathsf{u}_h) = \tfrac{h^4}{90}(\mathsf{a}^{(0,1)} - \mathsf{b}^{(1,0)})u^{(1,3)} + h^5\zeta + O(h^6),
\tag{3.30}
$$

and $\mathcal{L}_h\mathsf{u}_h$ approximates $\Delta\mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{b}\mathsf{u}_y + \mathsf{d}\mathsf{u} = \psi$ in (3.18) with the fourth-order of consistency at (x_i, y_j) in (2.24), where $\mathcal{L}_h\mathsf{u}$ and $\mathcal{L}_h\mathsf{u}_h$ are defined in (3.23), and $F_{i,j}$ is defined in (3.26). In our numerical examples, we set all free variables in (3.28) to zero, so $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ with $k, \ell = -1, 0, 1$ can be uniquely determined by solving (3.27).

In summary, we have the fourth-order compact 9-point FDM with the reduced pollution effects of $O(h^4)$ and $O(h^5)$ in the following theorem:

Theorem 3.1. *Assume $\kappa, u, \alpha, \beta, f$ are all smooth in (1.2), $\mathcal{L}_h\mathsf{u}_h$ is defined in (3.23), the nine coefficients $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ with $k, \ell = -1, 0, 1$ are uniquely determined by solving (3.27) with variables being zero in (3.28), the right-hand side $F_{i,j}$ is defined in (3.26), and all the functions in $\mathcal{L}_h\mathsf{u}_h$ are evaluated at the grid point (x_i, y_j) in (2.24). Then $h^{-2}\mathcal{L}_h\mathsf{u}_h$ approximates $\Delta\mathsf{u} + \mathsf{a}\mathsf{u}_x + \mathsf{b}\mathsf{u}_y + \mathsf{d}\mathsf{u} = \psi$ in (3.18) with the fourth-order of consistency at (x_i, y_j), and the truncation error $\tfrac{h^4}{90}(\mathsf{a}^{(0,1)} - \mathsf{b}^{(1,0)})u^{(1,3)} + h^5\lambda + O(h^6)$, where $\lambda = \zeta$ with $c_{1,1,1} = 0$ in (3.29), and $\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}, \psi$ are defined in (3.6), (3.11), and (3.16).*

Proof. The proof follows similar arguments as in Theorem 2.3. $\qquad\square$

Proposition 3.2. *The nine coefficients $C_{k,\ell} = \sum_{p=0}^{7} c_{k,\ell,p} h^p$ with $k, \ell = -1, 0, 1$ of the FDM in Theorem 3.1 satisfy*

$$
c_{\pm 1,\pm 1,0} = c_{\pm 1,\mp 1,0} = \tfrac{1}{6}, \qquad c_{\pm 1,0,0} = c_{0,\pm 1,0} = \tfrac{2}{3}, \qquad c_{0,0,0} = -\tfrac{10}{3}, \qquad \sum_{k=1}^{-1}\sum_{\ell=1}^{-1} c_{k,\ell,0} = 0.
\tag{3.31}
$$

That is, the FDM in Theorem 3.1 satisfies the discrete maximum principle, and generates an M-matrix, when the mesh size h is sufficiently small.

Proof. The proof follows similar arguments as in Proposition 2.4. $\qquad\square$

Combining the FDM in Theorem 3.1 with the iteration methods (3.5), (3.10), and (3.15), we provide the following Algorithm 2 with the CN method, Algorithm 3 with the BDF3 method, and Algorithm 4 with the BDF4 method to solve (1.2). Recall that u is the exact solution of (1.2), and $u^n = u|_{t=t_n=n\tau}$ in (3.2). Now, we define that u_h^n is the numerical solution computed by our proposed FDM and the iteration method at $t = t_n = n\tau$, where τ is defined in (3.2) and $N_2\tau = 1$.

Algorithm 2. *(CN method) Step 1: Apply the first and second statements in the iteration method (3.5) and the fourth-order FDM in Theorem 3.1 to solve (3.7) with (3.6) to compute u_h after 20 iterations. Step 2: $u_h^{n+1/2} = \mathsf{u}_h$ and $u_h^{n+1} = 2u_h^{n+1/2} - u_h^n$. Step 3: Repeat steps 1-2 with $n := n + 1$ until $n = N_2 - 1$.*

Algorithm 3. *(BDF3 method) Step 1: Apply the CN method to compute u^1, u^2. Step 2: Apply the iteration method (3.10) and the fourth-order FDM in Theorem 3.1 to solve (3.12) with (3.11) to compute u_h after 20 iterations. Step 3: $u_h^{n+3} = \mathsf{u}_h$ and repeat the step 2 with $n := n + 1$ until $n = N_2 - 3$.*

Algorithm 4. *(BDF4 method) Step 1: Apply the CN method to compute u^1, u^2, and use BDF3 method to calculate u^3. Step 2: Apply the iteration method (3.15) and the fourth-order FDM in Theorem 3.1 to solve (3.17) with (3.16) to compute u_h after 20 iterations. Step 3: $u_h^{n+4} = \mathsf{u}_h$ and repeat the step 2 with $n := n + 1$ until $n = N_2 - 4$.*

3.3. Approximations of the high-order partial derivatives of $\mathsf{a}^{(m,n)}$, $\mathsf{b}^{(m,n)}$, $\mathsf{c}^{(m,n)}$, and $\psi^{(m,n)}$

By the symbolic calculation of (3.27) of FDM in Theorem 3.1, we need $\mathsf{a}^{(m,n)}$, $\mathsf{b}^{(m,n)}$, $\mathsf{c}^{(m,n)}$ with $m + n \le 4$ and $\psi^{(m,n)}$ with $m + n \le 5$. As κ and f are available in (1.2), the expressions of (3.6), (3.11), and (3.16) imply that, we need to evaluate

$$
\left(\frac{\alpha_u(u)}{\kappa}\right)^{(m,n)} \text{ and } \left(\frac{\beta_u(u)}{\kappa}\right)^{(m,n)} \text{ with } m + n \le 4; \quad \text{and} \quad \left(\frac{u}{\kappa}\right)^{(m,n)} \text{ with } m + n \le 5.
\tag{3.32}
$$

We use the FDM in Section 2.5 to evaluate the high-order partial derivatives in (3.32).

4. Numerical experiments

Recall that the spatial domain $\Omega = (0,1)^2$, the temporal domain $I = [0,1]$, $u = u(x,y)$ and $u = u(x,y,t)$ are the exact solutions of the model problems (1.1) and (1.2), respectively, u_h is the numerical solution of (1.1) computed by Algorithm 1, and u_h^n is the numerical solution of (1.2) at $t = t_n$ computed by Algorithms 2 to 4, where

$$(x_i, y_j) = (ih, jh), \qquad h = 1/N_1, \qquad t_n = n\tau, \qquad \tau = 1/N_2 = rh, \qquad N_1/N_2 = r, \qquad N_1, N_2 \in \mathbb{N},$$

$i, j = 0, \ldots, N_1$ and $n = 0, \ldots, N_2$. In our numerical examples, we choose $r = 1/2$ for the CN method, and $r = 1$ for the BDF3 and BDF4 methods.

To verify the accuracy and the convergence rates of Algorithms 1 to 4, we use the following l_2 and l_∞ norms of errors for the model problems (1.1) and (1.2),

$$\|u_h - u\|_2^2 := h^2 \sum_{i,j=0}^{N_1} \left((u_h)_{i,j} - u(x_i, y_j) \right)^2, \qquad \|u_h - u\|_\infty := \max_{0 \le i,j \le N_1} \left| (u_h)_{i,j} - u(x_i, y_j) \right|,$$

for the steady nonlinear convection-diffusion equation (1.1);

$$\|u_h - u\|_2^2 := h^2 \sum_{i,j=0}^{N_1} \left((u_h^{N_2})_{i,j} - u(x_i, y_j, 1) \right)^2, \qquad \|u_h - u\|_\infty := \max_{0 \le i,j \le N_1} \left| (u_h^{N_2})_{i,j} - u(x_i, y_j, 1) \right|,$$

for the time-dependent nonlinear convection-diffusion equation (1.2);

where $(u_h)_{i,j}$ and $(u_h^{N_2})_{i,j}$ are values of u_h and $u_h^{N_2}$ at (x_i, y_j), respectively. In this section, we provide 2 examples for (1.1) and (1.2) in Section 4.1 and Section 4.2, respectively.

4.1. Two examples of the steady nonlinear convection-diffusion equation

In the following Example 4.1, we choose the variable diffusion coefficient $\kappa = \kappa(x,y)$ to examine the performance of Algorithm 1 with Theorems 2.2 and 2.3.

Example 4.1. The exact solution, the diffusion coefficient, and the nonlinear convection term in (1.1) are given by

$$u = \sin(3x)\cos(7y), \qquad \kappa = 2 + \sin(5x - 2y), \qquad \alpha = \cos(u), \qquad \beta = \sin(u),$$

f and g are obtained by plugging the above functions into (1.1). The numerical results are presented in Table 1 and Fig. 3. As we reduce the pollution effects of $O(h^4)$ and $O(h^5)$ in Theorem 2.3, we observe that Theorem 2.3 produces the smaller errors than Theorem 2.2 in Table 1. Furthermore, the convergence rates of Algorithm 1 with Theorem 2.3 are higher than 4 when $h \ge 1/2^5$.

Table 1: The performance in Example 4.1 of the proposed Algorithm 1 with Theorems 2.2 and 2.3, where 'RPE' denotes the 'reduced pollution effect'.

	Algorithm 1 with Theorem 2.2				Algorithm 1 with Theorem 2.3			
	The 4th-order FDM without RPE				The 4th-order FDM with RPE			
h	$\|u_h - u\|_2$	order	$\|u_h - u\|_\infty$	order	$\|u_h - u\|_2$	order	$\|u_h - u\|_\infty$	order
$1/2^3$	1.2164E-03		3.0974E-03		1.1144E-04		3.7841E-04	
$1/2^4$	7.2612E-05	4.07	2.0035E-04	3.95	2.2191E-06	5.65	5.9879E-06	5.98
$1/2^5$	4.5074E-06	4.01	1.2469E-05	4.01	1.1455E-07	4.28	2.9631E-07	4.34
$1/2^6$	2.8133E-07	4.00	7.7980E-07	4.00	7.1254E-09	4.01	1.8216E-08	4.02
$1/2^7$	1.7585E-08	4.00	4.8776E-08	4.00	4.4590E-10	4.00	1.1612E-09	3.97
$1/2^8$	1.0995E-09	4.00	3.0505E-09	4.00	2.7868E-11	4.00	7.3088E-11	3.99

In the following Example 4.2, we choose the constant κ to compare Algorithm 1 with the discontinuous Galerkin method constructed in [24]. Since we do not apply the postprocessing procedure in Algorithm 1, we compare the numerical results from [24] without the postprocessing procedure to ensure a fair comparison.
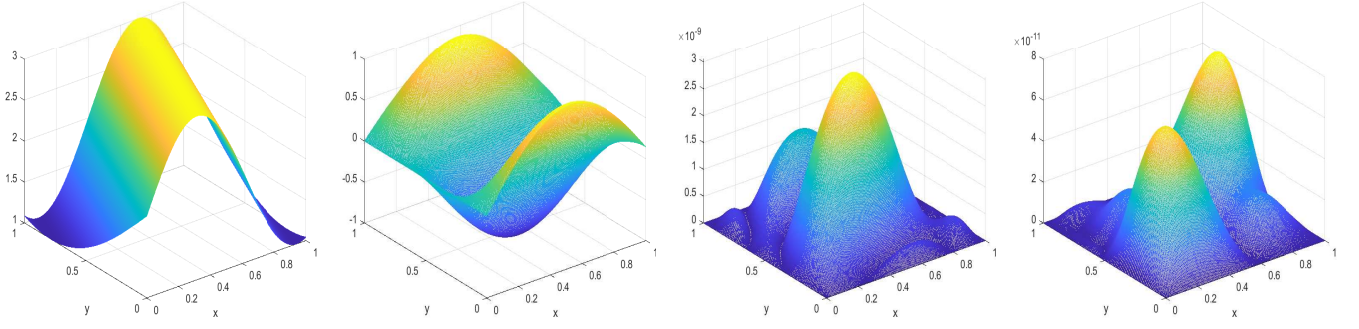
17

Figure 3: Example 4.1: The diffusion coefficient κ (first), the exact solution u (second), the error $|u_h - u|$ with the numerical solution u_h computed by Algorithm 1 and Theorem 2.2 (third), and the error $|u_h - u|$ with the numerical solution u_h computed by Algorithm 1 and Theorem 2.3 (fourth) on the closure of the spatial domain $[0, 1]^2$ with $h = 2^{-8}$.

Example 4.2. The exact solution, the diffusion coefficient, and the nonlinear convection term in (1.1) are given by

$$u = xy \tanh((1 - x)/\kappa) \tanh((1 - y)/\kappa), \qquad \kappa = 1/10, \qquad \alpha = u^2/2, \qquad \beta = u^2/2.$$

The numerical results are presented in Table 2 and Fig. 4. The error from Algorithm 1 with Theorem 2.3 is less than one-sixth of that in [24] when $h = 1/2^6$ in Table 2. We do not have the data of [24] if $h < 1/2^6$, but Table 2 indicates that Algorithm 1 with Theorem 2.3 achieves a stable convergence order of 4 if $h < 1/2^6$. So we can expect that the errors of [24] are approximately six times larger than those of our proposed method when $h < 1/2^6$. Furthermore, Algorithm 1 generates a matrix with only 9 nonzero bands, whereas [24] requires more than 9 nonzero bands to produce the results in Table 2.

Table 2: The performance in Example 4.2 of the proposed Algorithm 1 with Theorems 2.1 and 2.3, where 'RPE' denotes the 'reduced pollution effect'. The ratio r is equal to $\|u_h - u\|_2$ of [24] divided by $\|u_h - u\|_2$ of Algorithm 1 with Theorem 2.3.

	Theorem 2.1 (the 4th-order FDM)		Theorem 2.3 (the 4th-order FDM)		[24] (the 4th-order DG)		
	Without RPE		With RPE				
			col4		col6		r = col6/col4
h	$\|u_h - u\|_2$	order	$\|u_h - u\|_2$	order	$\|u_h - u\|_2$	order	r
$1/2^3$	1.0157E-02		1.1651E-02		5.97E-04		0.05
$1/2^4$	3.9676E-04	4.68	6.6129E-05	7.46	4.14E-05	3.85	0.63
$1/2^5$	3.3975E-05	3.55	8.8686E-07	6.22	2.79E-06	3.89	3.15
$1/2^6$	2.0859E-06	4.03	2.8940E-08	4.94	1.77E-07	3.98	6.12
$1/2^7$	1.3078E-07	4.00	1.7415E-09	4.05			
$1/2^8$	8.2163E-09	3.99	1.1497E-10	3.92			
$1/2^9$	5.1521E-10	4.00	7.6643E-12	3.91			

Next, we present 2 examples to verify the accuracy and the convergence rates of Algorithms 2 to 4 for the time-dependent nonlinear convection-diffusion equation (1.2) in the following Section 4.2.

4.2. Two examples of the time-dependent nonlinear convection-diffusion equation

Recall that $\tau = rh$ in (3.6), (3.11), and (3.16). In the following Examples 4.3 and 4.4, we choose $r = 1/2$ in Algorithm 2 and $r = 1$ in Algorithms 3 and 4.

Example 4.3. The exact solution, the diffusion coefficient, and the nonlinear convection term in (1.2) are given by

$$u = \sin(3t) \cos(2x - y), \qquad \kappa = 3 + \cos(x + 3y + t), \qquad \alpha = -u^3/3, \qquad \beta = \sin(u).$$

The numerical results are presented in Table 3 and Fig. 5. Table 3 confirms the third-order and fourth-order convergence rates of the BDF3 method in Algorithm 3 and the BDF4 method in Algorithm 4, respectively, for the diffusion coefficient $\kappa = \kappa(x, y, t)$.
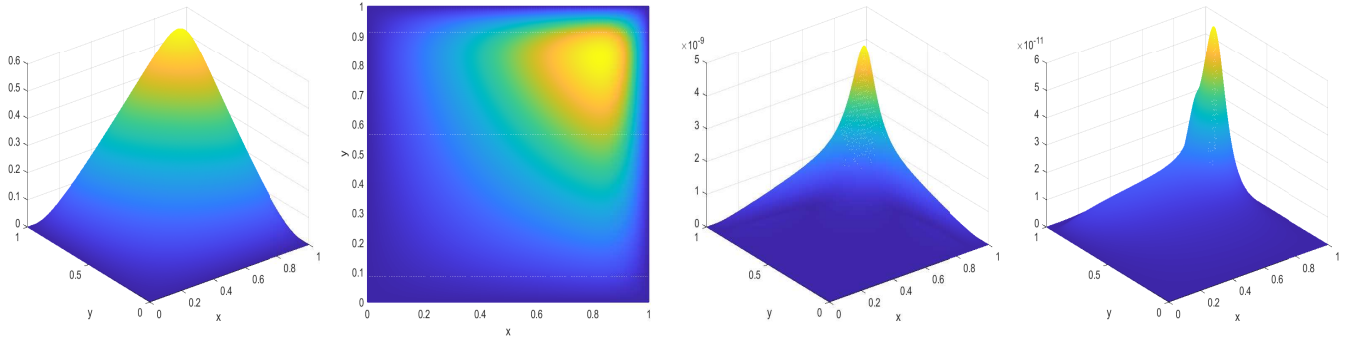
18

Figure 4: Example 4.2: The exact solution u (first and second), the error $|u_h - u|$ with the numerical solution u_h computed by Algorithm 1 with Theorem 2.1 (third), and the error $|u_h - u|$ with the numerical solution u_h computed by Algorithm 1 with Theorem 2.3 (fourth) on the closure of the spatial domain $[0, 1]^2$ with $h = 2^{-9}$.

Table 3: The performance in Example 4.3 of the proposed Algorithms 3 and 4.

| h	τ	Algorithm 3 with $\tau = h$ (BDF3) Using the 4th-order FDM				Algorithm 4 with $\tau = h$ (BDF4) Using the 4th-order FDM			
		$\|u_h - u\|_2$	order	$\|u_h - u\|_\infty$	order	$\|u_h - u\|_2$	order	$\|u_h - u\|_\infty$	order
$1/2^3$	$1/2^3$	3.1777E-04		6.0055E-04		2.3851E-04		4.7264E-04	
$1/2^4$	$1/2^4$	2.7632E-05	3.52	5.4008E-05	3.48	8.2563E-06	4.85	1.5931E-05	4.89
$1/2^5$	$1/2^5$	2.5943E-06	3.41	5.0863E-06	3.41	5.8285E-07	3.82	1.1277E-06	3.82
$1/2^6$	$1/2^6$	2.6848E-07	3.27	5.2770E-07	3.27	3.7437E-08	3.96	7.2471E-08	3.96
$1/2^7$	$1/2^7$	3.0023E-08	3.16	5.9143E-08	3.16	2.3653E-09	3.98	4.5802E-09	3.98
$1/2^8$	$1/2^8$	3.5303E-09	3.09	6.9619E-09	3.09	1.4867E-10	3.99	2.8794E-10	3.99
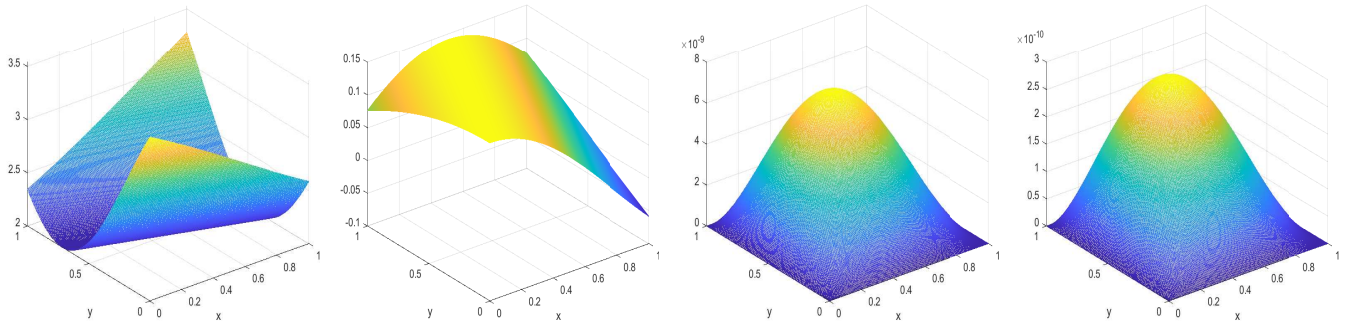
Figure 5: Example 4.3: The diffusion coefficient κ at $t = 1$ (first), the exact solution u at $t = 1$ (second), the error $|u_h - u|$ at $t = 1$ with the numerical solution u_h computed by Algorithm 3 (third), and the error $|u_h - u|$ at $t = 1$ with the numerical solution u_h computed by Algorithm 4 (fourth) on the closure of the spatial domain $[0, 1]^2$ with $h = 2^{-8}$.

To demonstrate the efficiency and accuracy of our FDM, a comparison with the results in [24] is presented in the following Example 4.4.

Example 4.4. The exact solution, the diffusion coefficient, and the nonlinear convection term in (1.2) are given by

$$u = (\exp(t) - 1)xy \tanh((1-x)/\kappa)\tanh((1-y)/\kappa), \qquad \kappa = 1/10, \qquad \alpha = u^2/2, \qquad \beta = u^2/2.$$

The numerical results are presented in Table 4 and Fig. 6. We note that the results in Table 4 from [24] are computed by the BDF3 method, while the proposed Algorithm 2 and Algorithm 3 use the CN and BDF3 methods, respectively. According to Table 4, even though Algorithm 2 is second-order accurate in the temporal discretization, and uses a larger time step τ than [24], Algorithm 2 yields the smaller errors than those from the third-order BDF3 method in [24]. Furthermore, if we apply the same BDF3 method for the time discretization, then the error from [24] is approximately 63 times greater than that of Algorithm 3 when $h = 1/2^6$, even the time step of Algorithm 3 is approximately 3 times larger than that of [24]. Since we reduce the truncation errors of $O(h^4)$ and $O(h^5)$ in Algorithms 2 and 3, the numerical orders of Algorithms 2 and 3 are higher than 2 and 3, respectively if $h \geq 1/2^7$. When $h = 1/2^8$, we obtain the desired convergence rates 2 and 3. Finally, Algorithms 2 and 3 form a nine-band matrix, but the number of nonzero bands in [24] to generate the results in Table 4 is higher than 9.

Table 4: The performance in Example 4.4 of the proposed Algorithms 2 and 3. The ratios r_2 and r_3 are equal to $\|u_h - u\|_2$ of [24] divided by $\|u_h - u\|_2$ of Algorithm 2 and Algorithm 3, respectively.

| | Algorithm 2 with $\tau = h/2$ (CN) | | | Algorithm 3 with $\tau = h$ (BDF3) | | | [24] with $\tau = 1/200$ (BDF3) | | | col9/col3 | col9/col6 |
| | | col3 | | | col6 | | | col9 | | | |
h	τ	$\|u_h - u\|_2$	order	τ	$\|u_h - u\|_2$	order	τ	$\|u_h - u\|_2$	order	r_2	r_3
$1/2^3$	1/16	3.3269E-03		1/8	1.8605E-03		1/200	5.09E-03		1.53	2.74
$1/2^4$	1/32	7.1559E-04	2.22	1/16	3.9588E-04	2.23	1/200	7.86E-04	2.69	1.10	1.99
$1/2^5$	1/64	3.3296E-05	4.43	1/32	8.7041E-06	5.51	1/200	1.01E-04	2.97	3.03	11.6
$1/2^6$	1/128	4.6570E-06	2.84	1/64	2.0001E-07	5.44	1/200	1.26E-05	2.99	2.71	63.0
$1/2^7$	1/256	1.0770E-06	2.11	1/128	1.0019E-08	4.32					
$1/2^8$	1/512	2.6870E-07	2.00	1/256	1.3557E-09	2.89					
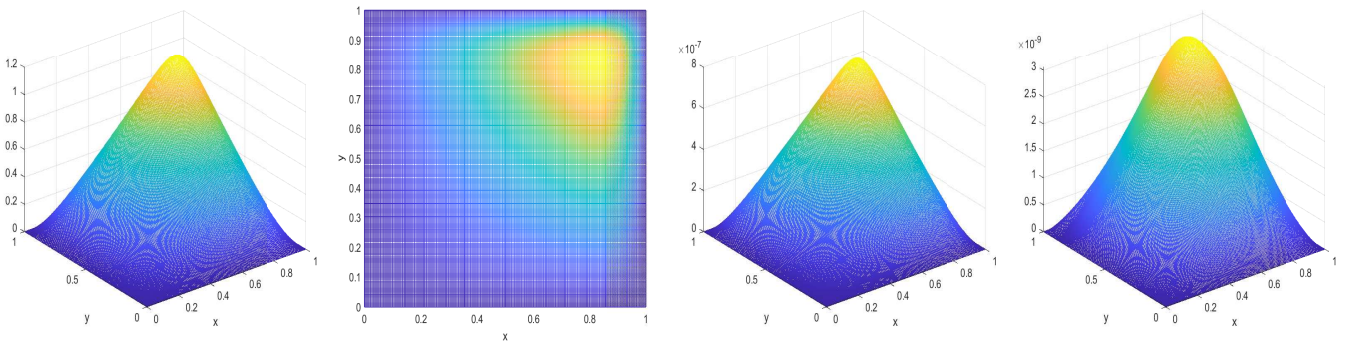
Figure 6: Example 4.4: The exact solution u (first and second), the error $|u_h - u|$ with the numerical solution u_h computed by Algorithm 2 (third), and the error $|u_h - u|$ with the numerical solution u_h computed by Algorithm 3 (fourth) on the closure of the spatial domain $[0,1]^2$ with $h = 2^{-8}$.

5. Contribution

In this paper, we consider the steady and time-dependent nonlinear convection-diffusion equations in a square domain with the Dirichlet boundary condition. The main contributions of this paper are as follows:

- We present the fourth-order compact 9-point FDM for the steady nonlinear equation, and derive the second-order to fourth-order compact 9-point FDMs for the time-dependent nonlinear equation. To increase the accuracy, we modify FDMs to reduce the pollution effects. Each proposed FDM preserves the discrete maximum principle and forms an M-matrix, when h is sufficiently small.

- We compare our method with the discontinuous Galerkin (DG) method in [24], and the numerical results demonstrate that our proposed FDM generates the smaller errors. Precisely, when we apply the second-order CN method, our FDM scheme produces the smaller errors than those from the third-order BDF3 and the DG methods in [24]. Particularly, if the same BDF3 method is used, then we achieve the error that is 1.6% of that in [24].

- Our proposed method is accurate, robust, and stable for the variable and time-dependent diffusion coefficient $\kappa(x, y, t)$, and the challenging nonlinear term $\nabla \cdot \boldsymbol{F}(u)$ (not limited to the Burgers equation). The examples verify the accuracy and the theoretical convergence rates in the l_2 and l_∞ norms.

- The matrix of the corresponding linear system constructed by our FDM only contains 9 nonzero bands. Due to the structure of the compact 9-point FDM, no special treatment is required for the grid points near the boundary. In comparison with the FEM, FVM, and DG methods, our high-order FDM avoids the numerical integration, resulting in the reduced computational cost. This advantage becomes particularly significant for the highly oscillatory κ, α, β, f.

The proposed method can be naturally extended to a 3D spatial domain and the more general nonlinear convection-diffusion-reaction equation: $u_t - \nabla \cdot (\kappa \nabla u) + \nabla \cdot \boldsymbol{F}(u) + r(u) = f$, where $\kappa = \kappa(x, y, t, u)$. We also plan to extend our method to solve the more complicated problems of the two-phase flow in porous media in [17] and the incompressible Boussinesq equation in [21, 26].

6. Declarations

Conflict of interest: The authors declare that they have no conflict of interest.
Data availability: Data will be made available on reasonable request.

References

[1] M. Bause and K. Schwegler, Analysis of stabilized higher-order finite element approximation of nonstationary and nonlinear convection-diffusion-reaction equations. *Comput. Methods Appl. Mech. Engrg.* **209-212** (2012), 184-196.

[2] J. Burkardt and C. Trenchea, Refactorization of the midpoint rule. *Appl. Math. Lett.* **107** (2020), 106438.

[3] E. Burman and A. Ern, Nonlinear diffusion and discrete maximum principle for stabilized Galerkin approximations of the convection-diffusion-reaction equation. *Comput. Methods Appl. Mech. Engrg.* **191** (2002), 3833-3855.

[4] C. Cancés, C. C-Hillairet, M. Herda, and S. Krell, Large time behavior of nonlinear finite volume schemes for convection-diffusion equations. *SIAM J. Numer. Anal.* **58** (2020), no. 5, 2544-2571.

[5] S. Clain, D. Lopes, R. M. S. Pereira, and P. A. Pereira, Very high-order finite difference method on arbitrary geometries with Cartesian grids for non-linear convection diffusion reaction equations. *J. Comput. Phys.* **498** (2024), 112667.

[6] B. Cockburn and C-W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal.* **35** (1998), no. 6, 2440-2463.

[7] V. Dolejší, M. Feistauer, and J. Hozman, Analysis of semi-implicit DGFEM for nonlinear convection-diffusion problems on nonconforming meshes. *Comput. Methods Appl. Mech. Engrg.* **196** (2007), 2813-2827.

[8] V. Dolejší, M. Feistauer, and V. Sobotíková, Analysis of the discontinuous Galerkin method for nonlinear convection-diffusion problems. *Comput. Methods Appl. Mech. Engrg.* **194** (2005), 2709-2733.

[9] V. Dolejší and M. Vlasák, Analysis of a BDF-DGFE scheme for nonlinear convection-diffusion problems. *Numer. Math.* **110** (2008), 405-447.

[10] R. Eymard, D. Hilhorst, and M. Vohralík, A combined finite volume–finite element scheme for the discretization of strongly nonlinear convection-diffusion-reaction problems on nonmatching grids. *Numer. Methods Partial Differ. Equ.* **26** (2010), 612-646.

[11] M. Feistauer, J. Felcman, and M. L-Medvid'ová, On the convergence of a combined finite volume-finite element method for nonlinear convection-diffusion problems. *Numer. Methods Partial Differ. Equ.* **13** (1997), 163-190.

[12] Q. Feng, B. Han, and P. Minev, Sixth order compact finite difference schemes for Poisson interface problems with singular sources. *Comp. Math. Appl.* **99** (2021), 2-25.

[13] Q. Feng, B. Han, and P. Minev, Compact 9-point finite difference methods with high accuracy order and/or M-matrix property for elliptic cross-interface problems. *J. Comput. Appl. Math.* **428** (2023), 115151.

[14] Q. Feng, B. Han, and P. Minev, Sixth-order hybrid finite difference methods for elliptic interface problems with mixed boundary conditions. *J. Comput. Phys.* **497** (2024), 112635.

[15] E. Hairer, S. P. Nørsett, and G. Wanner, Solving ordinary differential equations I: Nonstiff problems, 2nd ed. *Springer-Verlag Berlin Heidelberg.* 1993.

[16] W. Hundsdorfer and J. G. Verwer, Numerical solution of time-dependent advection-diffusion-reaction equations. *Springer-Verlag Berlin Heidelberg.* 2003.

[17] G. S. Jones and C. Trenchea, Discrete energy balance equation via a symplectic second-order method for two-phase flow in porous media. *Appl. Math. Comput.* **480** (2024), 128909.

[18] A. Kurganov and E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *J. Comput. Phys.* **160** (2000), 241-282.

[19] Z. Li and K. Ito, Maximum principle preserving schemes for interface problems with discontinuous coefficients. *SIAM J. Sci. Comput.* **23** (2001), no. 1, 339-361.

[20] H. Li and X. Zhang, On the monotonicity and discrete maximum principle of the finite difference implementation of C^0-Q^2 finite element method. *Numer. Math.* **145** (2020), 437-472.

[21] J-G. Liu, C. Wang, and H. Johnston, A fourth order scheme for incompressible Boussinesq equations. *J. Sci. Comput.* **18** (2003), 253-285.

[22] C. Michoski, A. Alexanderian, C. Paillet, E. J. Kubatko, and C. Dawson, Stability of nonlinear convection-diffusion-reaction systems in discontinuous Galerkin methods. *J. Sci. Comput.* **70** (2017), 516-550.

[23] M. Feistauer, V. Kučera, K. Najzar, and J. Prokopová, Analysis of space–time discontinuous Galerkin method for nonlinear convection-diffusion problems. *Numer. Math.* **117** (2011), 251-288.

[24] N. C. Nguyen, J. Peraire, and B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection-diffusion equations. *J. Comput. Phys.* **228** (2009), 8841-8855.

[25] T. E. Tezduyar and Y. J. Park, Discontinuity-capturing finite element formulations for nonlinear convection-diffusion-reaction equations. *Comput. Methods Appl. Mech. Engrg.* **59** (1986), 307-325.

[26] C. Wang, J-G. Liu, and H. Johnston, Analysis of a fourth order finite difference method for the incompressible Boussinesq equations. *Numer. Math.* **97** (2004), 555-594.

[27] Y. Xu and C-W. Shu, Error estimates of the semi-discrete local discontinuous Galerkin method for nonlinear convection-diffusion and KdV equations. *Comput. Methods Appl. Mech. Engrg.* **196** (2007), 3805-3822.

[28] J. Yan, A new nonsymmetric discontinuous Galerkin method for time dependent convection diffusion equations. *J. Sci. Comput.* **54** (2013), 663-683.