High-order compact difference method for two-dimension elliptic and parabolic equations with mixed derivatives

Tingfu Ma[1,2], Yongbin Ge[*,1]

[1] Institute of Applied Mathematics and Mechanics, Ningxia University, Yinchuan, 750021, P.R. China

[2] School of Mathematics and Computer Science, Ningxia Normal University, Guyuan, 756000, P.R. China

[*] Corresponding author

E-mail: nxmtf_2018@163.com[T.Ma], gyb@nxu.edu.cn[Y.Ge]

Abstract

In this article, firstly, based on Taylor series expansion and truncation error correction technology, combined with the fourth-order Padé schemes of the first-order derivatives, a new fourth-order compact difference (CD) scheme is constructed to solve the two-dimensional (2D) linear elliptic equation with a mixed derivative. In this new scheme, unknown function and its first-order derivatives are regarded as the unknown variables in calculation. Then, the method is extended to solve the 2D parabolic equation with a mixed derivative. To match the spatial fourth-order accuracy, The backward differentiation formula (BDF) is employed to gain the fourth-order accuracy for the temporal discretization. Truncation error is analyzed to display that the present scheme is fourth-order accuracy in space. In order to solve the resulting large-scale linear equations, a multigrid method is employed to accelerate the convergence speed of the conventional relaxation methods. Finally, numerical results indicate that the present schemes obtain fourth-order convergence and are more accurate than those in the literature.

2000 Mathematics Subject Classification. **65N06**. 65N15, 65N22.
Keywords. Compact difference scheme; High-order; Elliptic equation; Parabolic equation; Mixed derivative..

1 Introduction

In this article, we discuss the 2D parabolic model problem with the mixed derivatives

$$\alpha t - (u_{xx} + u_{yy}) + \beta(x,y,t)u_{xy} + p(x,y,t)u_x + q(x,y,t)u_y = f(x,y,t), \quad (x,y) \in \Omega,\ t \in (0,T] \tag{1.1}$$

with the initial condition

$$u(x,y,0) = u_0(x,y,0), \quad (x,y) \in \Omega \tag{1.2}$$

and the Dirichlet boundary condition

$$u(x,y,t) = g(x,y,t), \quad (x,y) \in \partial\Omega,\ t \in (0,T]. \tag{1.3}$$

Here Ω is a domain and $\partial\Omega$ is the boundary. $(0,T]$ is time interval. Unknown function $u(x,y,t)$, the variable coefficients functions $\beta(x,y,t), p(x,y,t)$ and $q(x,y,t)$, the forcing function $f(x,y,t)$ and the initial condition $u(x,y,0)$ and boundary condition $g(x,y,t)$ are assumed to be continuously differentiable and have the required partial derivatives on Ω. α is a constant and $\beta(x,y,t)$ satisfies the condition to make the spatial operators elliptic, i.e., $\beta(x,y,t)^2 < 4$. This equation is very

Tbilisi Mathematical Journal 13(4) (2020), pp. 141–167.
Tbilisi Centre for Mathematical Sciences.
Received by the editors: 11 May 2020 .
Accepted for publication: 08 October 2020.

important in many fields such as financial mathematics [1], European call options [2] and fluid mechanics [3].

In the past decades, various special techniques have been developed rapidly based on high-order compact difference (CD) schemes [4–32] which enjoy high-order accuracy and high resolution. In 1985, Gupta et al. [3] proposed a fourth-order CD scheme for the 2D elliptic equation without a mixed derivative term and extended the method to the 2D incompressible Navier-Stokes equations [6]. Later, Gupta and Kouatchou used the method and a symbolic procedure with software package Mathematica to get several difference schemes for the 3D Poisson equation [8]. Zhang et al. used the method to resolve the 2D and 3D boundary layer convection-diffusion problems on non-uniform grids [9,10] and derived the high-order compact (HOC) difference scheme for solving the 2D convection-diffusion equation on rotated grid [11]. Recently, Karaa also developed two high-order CD schemes to solve the 2D elliptic and parabolic problems with the mixed derivatives [12]. The methods mentioned above are to utilize the Taylor expansions by 2D power series of all the functions included in the differential equation at grid point (i, j) and they are truncated up to a certain power of the polynomials $x^i y^j$, i.e., the coefficients of higher orders in the Taylor series are set to be zero. Then, substitute all expansions into the original equation and compare the coefficients of $x^i y^j$ to obtain the linear constraints on the unknown coefficients. Finally, finite difference scheme can be derived by resolving the system of linear equations arising from constraints on the unknown coefficients. In Ref. [13] , the authors also used this method to develop HOC difference schemes for the general 2D elliptic equation. And it was pointed out in [13] that over a 9-point grid stencil, the fourth-order CD schemes for the general 2D elliptic equation with variable coefficients and mixed derivative exist only if the coefficients of the second-order derivatives u_{xx} and u_{yy} are same. Namely, if the coefficients of the second-order derivatives u_{xx} and u_{yy} are not same, the coefficient of the mixed derivative must be zero (without mixed derivatives) to get a fourth-order CD scheme, as considered and done by Gupta et al. in Ref. [5].

Alternatively, another technique for developing HOC difference schemes was advocated by Choo and Schultz [14–16] and Spotz and Carey [17–19]. This method can be regarded as a correction of truncation error remainder of the low-order approximations. Firstly, a low-order approximation scheme (e.g. the central difference scheme) with its leading truncation errors is given out; then the higher-order truncation error terms are substituted by the original differential equation; lastly, higher-order representations are approximated to a certain order to yield a high-order approximation scheme. This approach is employed by many authors to develop HOC difference schemes for other elliptic equations [20, 22–24], Helmholtz equations [25–28] and the incompressible Navier-Stokes equations [14, 16, 18, 29], etc. It needs to be pointed out that with the same grid stencils, schemes derived by two approaches are same. We can get this conclusion by comparing the 9-point scheme in [6] and [15] for the 2D convection diffusion equation, the 19-point scheme in [8] and [19] for the 3D Poisson equation or the 19-point scheme in [20,21] for the 3D convection diffusion equation. It is no strange because the both methods are based on the Taylor expansions although the procedures seem different. Gupta et al.'s method is more flexible to derive different HOC schemes with different grid stencils. For instance, besides the 19-point scheme, which was also derived in [19] , a 15-point scheme and a 21-point scheme were also derived in [8] for the 3D Poisson equation. But the procedure of Gupta et al.'s method is extremely tedious, especially for the derivation in 3D [8].

In 1998, Chu and Fan [34] proposed a combined compact difference (CCD) scheme, which was derived by using local Hermitian polynomials and is the sixth-order accuracy with 3 points for one dimension. It is a pity that the CCD scheme only deals with elliptic equation. More recently,

Sen [30] proposed implicit CD schemes for the 2D parabolic equation with the mixed derivative term. The scheme is fourth-order accuracy in space and second-order accuracy in time. In this study, firstly, we propose a new fourth-order CD scheme for solving the 2D elliptic equation with variable coefficients of the first-derivatives and the mixed derivatives. Then we extend the method to solve the parabolic equation (1.1). The fourth-order backward differentiation formula is employed to discrete temporal derivative, so a full implicit fourth-order accurate CD scheme is derived for solving equation (1.1). The distinguishing feature of the present method is that the schemes contain the representations of the spatial first-derivatives of the unknown function, which are computed by the fourth-order Padé schemes separately. The schemes are different from the 9-point scheme derived by Karaa in [12] and Sen in [30]. The advantage of the present method is that it has small amplitude of the coefficients for higher-order leading truncation error terms, especially for those problems with large first-derivatives (i.e., high Reynolds problems in fluid dynamics). The present schemes can get more accurate solution than the Karaa's scheme [12] and Sen's scheme [30]. To solve the large-scale sparse linear systems arising from the CD schemes, a multigrid method is employed to accelerate the convergence speed of conventional relaxation methods.

The remainder of this paper is organized as follows. Section 2 presents the formulation of the fourth-order compact difference scheme for solving elliptic equation with a mixed derivative. In Section 3, truncation error analysis is given to show that the scheme is fourth-order accuracy for the interior grid points. Then, the present method is extended to the parabolic equation (1.1) and a full implicit fourth-order CD scheme is proposed in Section 4. Next, multigrid method is introduced and a multigrid algorithm is designed to solve the linear systems in Section 5. Numerical experiments are conducted in Section 6, and we compare the results with other known methods. Finally, a brief conclusion is given in Section 7.

2 Fourth-order compact scheme for elliptic equation

For convenience, we symbolize the derivatives by $\{u, u_x, u_y, u_{xx}, u_{yy}, u_{xy}\}$, respectively. We consider the general Dirichlet boundary condition. In the following, we will discuss how to implement the fourth-order CD scheme.

2.1 Computation for inner grid points

Firstly, we start by briefly deriving a 9-point fourth-order CD scheme for steady form of equation (1.1)

$$u_{xx} + u_{yy} + \beta(x,y)u_{xy} + p(x,y)u_x + q(x,y)u_y = f(x,y), \quad (x,y) \in \Omega \tag{2.1}$$

Where unknown function $u(x,y)$, the variable coefficients functions $\beta(x,y), p(x,y)$ and $q(x,y)$ and the forcing function $f(x,y)$ are assumed to be continuously differentiable and have the required partial derivatives on Ω.

For the sake of simplicity, we use Fig.1 to denote the placement of nine points. Assuming a uniform mesh in both x - and y - directions, where h is the mesh size, we number the mesh point (x,y), $(x+h,y)$, $(x,y+h)$, $(x-h,y)$, $(x,y-h)$, $(x+h,y+h)$, $(x-h,y+h)$, $(x-h,y-h)$ and $(x+h,y-h)$ as 0, 1, 2, 3, 4, 5, 6, 7 and 8, respectively. The nodal values of u, u_x and u_y are denoted by u_0, u_1, u_2, u_3, u_5, u_6, u_7, u_8, u_{x0}, u_{x1}, u_{x2}, u_{x3}, u_{x4}, u_{x5}, u_{x6}, u_{x7}, u_{x8}, u_{y0}, u_{y1}, u_{y2}, u_{y3}, u_{y4}, u_{y5}, u_{y6}, u_{y7}, u_{y8}, etc.

By the Taylor series expansions at point (x, y), we have

$$u_x = \delta_x u - \frac{h^2}{6}u_{xxx} + O\left(h^4\right), \tag{2.2}$$

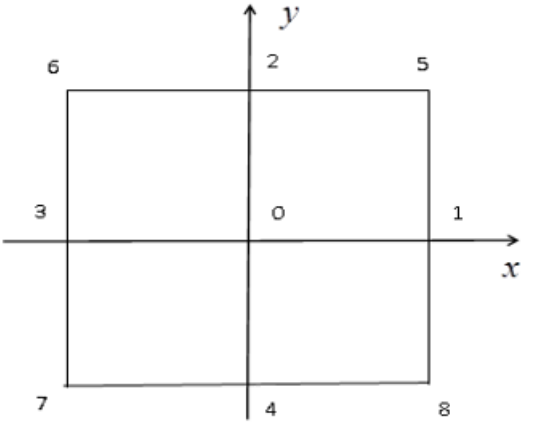


FIG. 1. Labeling of the nine grid points.

$$u_y = \delta_y u - \frac{h^2}{6} u_{yyy} + O\left(h^4\right). \tag{2.3}$$

Here δ_x and δ_y are central difference operators for the first-order derivatives (see Appendix A). Substituting (2.2) and (2.3) into (2.1) we obtain

$$u_{xx} + u_{yy} + \beta u_{xy} + p\left(\delta_x u - \frac{h^2}{6} u_{xxx}\right) + q\left(\delta_y u - \frac{h^2}{6} u_{yyy}\right) + O\left(h^4\right) = f. \tag{2.4}$$

Differentiating Equation (2.1) with respect to x and y respectively, we get

$$u_{xxx} = f_x - \left(u_{yyx} + \beta_x u_{xy} + \beta u_{xxy} + p_x u_x + p u_{xx} + q u_{xy} + q_x u_y\right), \tag{2.5}$$
$$u_{yyy} = f_y - \left(u_{xxy} + \beta_y u_{xy} + \beta u_{xyy} + p_y u_x + p u_{xy} + q_y u_y + q u_{yy}\right), \tag{2.6}$$

Substituting (2.5) and (2.6) into (2.4), and rearranging it, we have

$$\begin{aligned} &A u_{xx} + B u_{yy} + C u_{xy} + D u_{xxy} + E u_{yyx} + F u_x + G u_y + p\delta_x u + q\delta_y u \\ &= f + \frac{ph^2}{6} f_x + \frac{qh^2}{6} f_y + O\left(h^4\right) \end{aligned} \tag{2.7}$$

in which

$$A = 1 + \frac{p^2 h^2}{6}, \qquad B = 1 + \frac{q^2 h^2}{6},$$
$$C = \beta + \frac{p\beta_x h^2}{6} + \frac{pqh^2}{3} + \frac{q\beta_y h^2}{6}, \qquad D = \frac{qh^2}{6} + \frac{p\beta h^2}{6},$$
$$E = \frac{ph^2}{6} + \frac{q\beta h^2}{6}, \qquad F = \frac{pp_x h^2}{6} + \frac{qp_y h^2}{6},$$

$$G = \frac{pq_x h^2}{6} + \frac{qq_y h^2}{6}.$$

To obtain a fourth-order CD for Equation (2.7), consider the following approximations for all the second- and third-order derivatives $\{u_{xx}, u_{yy}, u_{xy}, u_{xxy}, u_{yyx}\}$

$$u_{xx} = 2\delta_x^2 u - \delta_x u_x + O(h^4), \tag{2.8}$$
$$u_{yy} = 2\delta_y^2 u - \delta_y u_y + O(h^4), \tag{2.9}$$
$$u_{xy} = \delta_x u_y + \delta_y u_x - \delta_x \delta_y u + O(h^4), \tag{2.10}$$
$$u_{xxy} = \delta_x^2 u_y + O(h^2), \tag{2.11}$$
$$u_{yyx} = \delta_y^2 u_x + O(h^2). \tag{2.12}$$

Substituting (2.8)-(2.12) into (2.7), we have

$$\begin{aligned}&\left(2A\delta_x^2 + 2B\delta_y^2 + p\delta_x + q\delta_y - C\delta_x\delta_y\right) u + \left(-A\delta_x + C\delta_y + E\delta_y^2 + F\right) u_x \\ &+ \left(-B\delta_y + C\delta_x + D\delta_x^2 + G\right) u_y = f + \frac{ph^2}{6} f_x + \frac{qh^2}{6} f_y + O\left(h^4\right).\end{aligned} \tag{2.13}$$

Substituting $\delta_x, \delta_y, \delta_x^2, \delta_y^2, \delta_x\delta_y$ (see Appendix A) into (2.13), neglecting the truncation error terms, we obtain the following CD scheme.

$$\begin{aligned}&\left(-\frac{4A}{h^2} - \frac{4B}{h^2}\right) u_0 + \left(\frac{2A}{h^2} + \frac{p_0}{2h}\right) u_1 + \left(\frac{2B}{h^2} + \frac{q_0}{2h}\right) u_2 + \left(\frac{2A}{h^2} - \frac{p_0}{2h}\right) u_3 + \left(\frac{2B}{h^2} - \frac{q_0}{2h}\right) u_4 \\ &+ \frac{C}{4h^2}\left(-u_5 + u_6 - u_7 + u_8\right) + \left(-\frac{2E}{h^2} + F\right) u_{x0} - \frac{A}{2h} u_{x1} + \left(\frac{C}{2h} + \frac{E}{h^2}\right) u_{x2} \\ &+ \frac{A}{2h} u_{x3} + \left(-\frac{C}{2h} + \frac{E}{h^2}\right) u_{x4} + \left(-\frac{2D}{h^2} + G\right) u_{y0} + \left(\frac{C}{2h} + \frac{D}{h^2}\right) u_{y1} - \frac{B}{2h} u_{y2} \\ &+ \left(-\frac{C}{2h} + \frac{D}{h^2}\right) u_{y3} + \frac{B}{2h} u_{y4} = f_0 + \frac{p_0 h^2}{6} f_{x0} + \frac{q_0 h^2}{6} f_{y0}.\end{aligned} \tag{2.14}$$

We notice that for each interior grid point (x, y), there are three unknowns $\{u, u_x, u_y\}$ to be determined. Compatible fourth-order accurate Padé approximations for space derivatives given by

$$\left(1 + \frac{h^2}{6}\delta_x^2\right) u_x = \delta_x u \tag{2.15}$$

and

$$\left(1 + \frac{h^2}{6}\delta_y^2\right) u_y = \delta_y u \tag{2.16}$$

are used to close the system.

2.2 Computation for boundary grid points

In Section 2.1, difference equations for each interior grid point have been developed. The calculation of many practical problems may encounter the case of dealing with non-periodic boundary conditions. Then, we will introduce approximations for the first-order derivatives at the near boundary

nodes. These approximations are of necessity, non-central or one-sided. Equations (2.14)-(2.16) show that the fourth-order scheme can be applied from $i = 1$ to $N-1$ (or $j = 1$ to $N-1$). Additional fourth-order expressions are needed at the boundary nodes to close the system. In this study, we simply borrow the consistent fourth-order schemes of the first-order derivatives $\{u_x, u_y\}$ in Ref. [31] for boundary nodes computation, because they are very stable under the condition of convection dominance. They are expressed as follows.

Left boundary condition of u_x

$$\begin{aligned}(u_x)_{0,j} + \frac{14}{15}(u_x)_{1,j} = \frac{1}{h}\Bigg(&-\frac{184}{75}u_{0,j} + \frac{703}{180}u_{1,j} - \frac{89}{30}u_{2,j} + \frac{67}{30}u_{3,j} \\ &-\frac{77}{90}u_{4,j} + \frac{41}{300}u_{5,j}\Bigg), \quad j = 1, 2\cdots, N.\end{aligned} \tag{2.17}$$

Right boundary condition of u_x

$$\begin{aligned}(u_x)_{N,j} - \frac{14}{15}(u_x)_{N-1,j} = \frac{1}{h}\Bigg(&\frac{52}{25}u_{N,j} - \frac{1067}{180}u_{N-1,j} + \frac{67}{10}u_{N-2,j} - \frac{41}{10}u_{N-3,j} \\ &+\frac{133}{90}u_{N-4,j} - \frac{69}{300}u_{N-5,j}\Bigg), \quad j = 1, 2\cdots, N.\end{aligned} \tag{2.18}$$

Bottom boundary condition of u_y

$$\begin{aligned}(u_y)_{i,0} + \frac{14}{15}(u_y)_{i,1} = \frac{1}{h}\Bigg(&-\frac{184}{75}u_{i,0} + \frac{703}{180}u_{i,1} - \frac{89}{30}u_{i,2} + \frac{67}{30}u_{i,3} \\ &-\frac{77}{90}u_{i,4} + \frac{41}{300}u_{i,5}\Bigg), \quad i = 1, 2\cdots, N.\end{aligned} \tag{2.19}$$

Top boundary condition of u_y

$$\begin{aligned}(u_y)_{i,N} - \frac{14}{15}(u_y)_{i,N-1} = \frac{1}{h}\Bigg(&\frac{52}{25}u_{i,N} - \frac{1067}{180}u_{i,N-1} + \frac{67}{10}u_{i,N-2} - \frac{41}{10}u_{i,N-3} \\ &+\frac{133}{90}u_{i,N-4} - \frac{69}{300}u_{i,N-5}\Bigg), \quad i = 1, 2\cdots, N.\end{aligned} \tag{2.20}$$

3 Truncation error analysis

By the Taylor series expansions at point (x, y), we get

$$u_x = \delta_x u - \frac{h^2}{6}u_{xxx} - \frac{h^4}{120}u_{x^{(5)}} + O\left(h^6\right), \tag{3.1}$$

$$u_y = \delta_y u - \frac{h^2}{6}u_{yyy} - \frac{h^4}{120}u_{y^{(5)}} + O\left(h^6\right). \tag{3.2}$$

Meanwhile, differentiating Eq. (2.1) with respect to x and y, we obtain

$$u_{xxx} = f_x - \left(u_{yyx} + \beta_x u_{xy} + \beta u_{xxy} + p_x u_x + p u_{xx} + q u_{xy} + q_x u_y\right), \tag{3.3}$$

$$u_{yyy} = f_y - (u_{xxy} + \beta_y u_{xy} + \beta u_{xyy} + p_y u_x + p u_{xy} + q_y u_y + q u_{yy}). \tag{3.4}$$

Substituting (3.3) and (3.4) into (2.4), we have

$$\begin{aligned} &Au_{xx} + Bu_{yy} + Cu_{xy} + Du_{xxy} + Eu_{yyx} + Fu_x + Gu_y + p\delta_x u + q\delta_y u \\ &= f + \frac{ph^2}{6} f_x + \frac{qh^2}{6} f_y + O\left(h^4\right). \end{aligned} \tag{3.5}$$

To obtain the fourth-order CD formulation for Eq. (3.5), consider the following approximations for all the derivatives.

$$u_{xx} = 2\delta_x^2 u - \delta_x u_x + \frac{h^4}{360} u_{x^{(6)}} + O\left(h^6\right) \tag{3.6}$$

$$u_{yy} = 2\delta_y^2 u - \delta_y u_y + \frac{h^4}{360} u_{y^{(6)}} + O\left(h^6\right) \tag{3.7}$$

$$u_{xy} = \delta_x u_y + \delta_y u_x - \delta_x \delta_y u + \frac{h^4}{36} u_{x^{(3)}y^{(3)}} + O\left(h^6\right) \tag{3.8}$$

$$u_{xxy} = \delta_x^2 u_y - \frac{h^2}{12} u_{x^{(4)}y} + O\left(h^4\right) \tag{3.9}$$

$$u_{yyx} = \delta_y^2 u_x - \frac{h^2}{12} u_{xy^{(4)}} + O\left(h^4\right) \tag{3.10}$$

Substituting (3.6)-(3.10) into Eq. (3.5), we have

$$\begin{aligned} &A(2\delta_x^2 u - \delta_x u_x + \frac{h^4}{360} u_{x^{(6)}}) + B(2\delta_y^2 u - \delta_y u_y + \frac{h^4}{360} u_{y^{(6)}}) + D(\delta_x^2 u_y - \frac{h^2}{12} u_{x^{(4)}y}) \\ &+ C(\delta_x u_y + \delta_y u_x - \delta_x \delta_y u + \frac{h^4}{36} u_{x^{(3)}y^{(3)}}) + E(\delta_y^2 u_x - \frac{h^2}{12} u_{xy^{(4)}}) + Fu_x + Gu_y \\ &+ p\delta_x u + q\delta_y u - \frac{h^4 p}{120} u_{x^{(5)}} - \frac{h^4 q}{120} u_{y^{(5)}} = f + \frac{ph^2}{6} f_x + \frac{qh^2}{6} f_y + O(h^6) \end{aligned} \tag{3.11}$$

Notice that the first-derivatives are calculated by the fourth-order accurate Padé approximations independently.

$$\left(1 + \frac{h^2}{6}\delta_x^2\right) u_x = \delta_x u + \frac{h^4}{180} u_{x^{(5)}} + O\left(h^6\right) \tag{3.12}$$

$$\left(1 + \frac{h^2}{6}\delta_y^2\right) u_y = \delta_y u + \frac{h^4}{180} u_{y^{(5)}} + O\left(h^6\right) \tag{3.13}$$

Substituting the truncation errors of all these derivatives in Eq. (3.11) , we have

$$\begin{aligned} \text{Truncation} = &\frac{h^4}{180}\left[\left(A + C + F - \frac{3p}{2}\right) u_{x^{(5)}} + \left(B + C + G - \frac{3q}{2}\right) u_{y^{(5)}}\right] \\ &- \frac{h^2}{12}(Eu_{xy^{(4)}} + Du_{x^{(4)}y}) + \frac{h^4}{360}(Au_{x^{(6)}} + Bu_{y^{(6)}} + 10Cu_{x^{(3)}y^{(3)}}). \end{aligned} \tag{3.14}$$

Finally, we can get main term of truncation errors is

$$-\frac{h^4}{180}\left[\left(1-\beta+\frac{3p}{2}+\frac{3q}{2}\right)\left(u_{x^{(5)}}+u_{y^{(5)}}\right)+\frac{1}{2}\left(u_{x^{(6)}}+u_{y^{(6)}}+10\beta u_{x^{(3)}y^{(3)}}\right)\right] \tag{3.15}$$

It is clear that the truncation of this scheme is $O\left(h^4\right)$.

It is worth noting that in the fourth-order CD scheme [17], the third- and fourth-order derivatives of the truncation errors that are leaded by the Taylor series expansions are compactly approximated by regarding the original second-order differential equation as an auxiliary relation, but here we only treat the leading third-order derivatives, while leave the fourth-order derivatives to be replaced by transport variable u and its first-order derivatives u_x and u_y. So that the coefficient of the main term of the truncation errors is relatively small. In other words, the present scheme has small amplitude of the coefficients for higher leading truncation error terms.

4 Extension to the parabolic equation

In this section, we discuss the temporal differencing of the parabolic equation (1.1). We next use implicit schemes based on the backward time-differencing. This scheme called backward differentiation formula (BDF), has been proved unconditionally stable and widely used for solving ordinary differential equations and the Navier-Stokes equations (see, e.g. [35, 36]).

To ensure the fourth-order accuracy in time, we select the BDF with fourth-order accuracy (BDF4 for short). Note that this is an implicit scheme with five time levels. Besides known initial time level(0 time level), we will use Crank-Nicolson scheme to compute the first and second time levels, then BDF3 to compute the third time level. Then, we use BDF4 for subsequent time levels. BDF3 and BDF4 are expressed as follows [35].

BDF3

$$u_t^{n+1}=\left(11u^{n+1}-18u^n+9u^{n-1}-2u^{n-2}\right)/(6\tau)+O\left(\tau^3\right) \tag{4.1}$$

BDF4

$$u_t^{n+1}=\left(25u^{n+1}-48u^n+36u^{n-1}-16u^{n-2}+3u^{n-3}\right)/(12\tau)+O\left(\tau^4\right) \tag{4.2}$$

Where, the time level by a superscript n and the time step size by τ.

For the unsteady case, Eq. (1.1) is similar to Eq. (2.1), the HOC approach developed for the steady case can be extended directly to the unsteady case by simply replacing $f(x,y)$ by $\alpha u_t - f(x,y,t), \beta(x,y)$ by $-\beta(x,y,t), p(x,y)$ by $-p(x,y,t)$ and $q(x,y)$ by $-q(x,y,t)$ in Eq. (2.1), then Eq. (1.1) is rewritten as

$$u_{xx}+u_{yy}-\beta(x,y,t)u_{xy}-p(x,y,t)u_x-q(x,y,t)u_y=\alpha u_t-f(x,y,t) \tag{4.3}$$

We next approximate the unsteady equation with the help of Eq. (2.13) as

$$\begin{aligned}&\left(2A\delta_x^2+2B\delta_y^2-p\delta_x-q\delta_y-C\delta_x\delta_y\right)u+\left(-A\delta_x+C\delta_y+E\delta_y^2+F\right)u_x\\&+\left(-B\delta_y+C\delta_x+D\delta_x^2+G\right)u_y=\alpha u_t-f-\frac{ph^2}{6}\left(\alpha u_{xt}-f_x\right)-\frac{qh^2}{6}\left(\alpha u_{yt}-f_y\right)\end{aligned} \tag{4.4}$$

in which

$$A = 1 + \frac{p^2h^2}{6}, \qquad B = 1 + \frac{q^2h^2}{6},$$
$$C = -\beta + \frac{p\beta_x h^2}{6} + \frac{pqh^2}{3} + \frac{q\beta_y h^2}{6}, \qquad D = -\frac{qh^2}{6} + \frac{p\beta h^2}{6},$$
$$E = -\frac{ph^2}{6} + \frac{q\beta h^2}{6}, \qquad F = \frac{pp_x h^2}{6} + \frac{qp_y h^2}{6},$$
$$G = \frac{pq_x h^2}{6} + \frac{qq_y h^2}{6}.$$

We notice that besides u_t, u_{xt} and u_{yt}, are also needed to be discretized similarly. The relevant BDF4 formulas are given as follows. (The detail discussion can be found in [35]).

$$u_{xt}^{n+1} = \left(25u_x^{n+1} - 48u_x^n + 36u_x^{n-1} - 16u_x^{n-2} + 3u_x^{n-3}\right)/(12\tau) + O\left(\tau^4\right) \tag{4.5}$$
$$u_{yt}^{n+1} = \left(25u_y^{n+1} - 48u_y^n + 36u_y^{n-1} - 16u_y^{n-2} + 3u_y^{n-3}\right)/(12\tau) + O\left(\tau^4\right) \tag{4.6}$$

Note that all spatial derivatives are discretized at the $n+1$ time level, and exploiting the standard second-order difference operators $\delta_x\delta_y$, δ_x^2, δ_y^2 and δ_{xy}. (Details are given in appendix A). Omitting the high-order items and making suitable rearrangements, Eq. (4.4) can be simplified as

$$\begin{aligned}
&\left(-\frac{4A}{h^2} - \frac{4B}{h^2} - \frac{25\alpha}{12\tau}\right)u_0^{n+1} + \left(\frac{2A}{h^2} - \frac{p_0}{2h}\right)u_1^{n+1} + \left(\frac{2B}{h^2} - \frac{q_0}{2h}\right)u_2^{n+1} + \left(\frac{2A}{h^2} + \frac{p_0}{2h}\right)u_3^{n+1} \\
&+ \left(\frac{2B}{h^2} + \frac{q_0}{2h}\right)u_4^{n+1} + \frac{C}{4h^2}\left(-u_5^{n+1} + u_6^{n+1} - u_7^{n+1} + u_8^{n+1}\right) + \left(-\frac{2E}{h^2} + F + \frac{25\alpha h^2 p_0}{72\tau}\right)u_{x0}^{n+1} \\
&- \frac{A}{2h}u_{x1}^{n+1} + \left(\frac{C}{2h} + \frac{E}{h^2}\right)u_{x2}^{n+1} + \frac{A}{2h}u_{x3}^{n+1} + \left(-\frac{C}{2h} + \frac{E}{h^2}\right)u_{x4}^{n+1} + \left(-\frac{2D}{h^2} + G + \frac{25\alpha h^2 q_0}{72\tau}\right)u_{y0}^{n+1} \\
&+ \left(\frac{C}{2h} + \frac{D}{h^2}\right)u_{y1}^{n+1} - \frac{B}{2h}u_{y2}^{n+1} + \left(-\frac{C}{2h} + \frac{D}{h^2}\right)u_{y3}^{n+1} + \frac{B}{2h}u_{y4}^{n+1} \\
&= -\frac{4\alpha}{\tau}u_0^n + \frac{2\alpha h^2 p_0}{3\tau}u_{x0}^n + \frac{2\alpha h^2 q_0}{3\tau}u_{y0}^n + \frac{3\alpha}{\tau}u_0^{n-1} - \frac{\alpha h^2 p_0}{2\tau}u_{x0}^{n-1} - \frac{\alpha h^2 q_0}{2\tau}u_{y0}^{n-1} - \frac{4\alpha}{3\tau}u_0^{n-2} \\
&+ \frac{2\alpha h^2 p_0}{9\tau}u_{x0}^{n-2} + \frac{2\alpha h^2 q_0}{9\tau}u_{y0}^{n-2} + \frac{\alpha}{4\tau}u_0^{n-3} - \frac{\alpha h^2 p_0}{24\tau}u_{x0}^{n-3} - \frac{\alpha h^2 q_0}{24\tau}u_{y0}^{n-3} + \frac{p_0 h^2}{6}f_{x0}^{n+1} + \frac{q_0 h^2}{6}f_{y0}^{n+1} \\
&- f_0^{n+1}
\end{aligned} \tag{4.7}$$

While $n \geq 3$. Eq. (4.7) is full-implicit high-order CD scheme for Eq. (1.1). in which, values of p, q and β are computed at the time level t^{n+1}. The truncation error is $O\left(\tau^4 + h^4\right)$. In order to compute the nodal values on the first and second time levels, considering equation (4.4) at time level $t^{n+1/2}$

$$\begin{aligned}
&\left[A\left(2\delta_x^2 u - \delta_x u_x\right)\right]^{n+\frac{1}{2}} + \left[B\left(2\delta_y^2 u - \delta_y u_y\right)\right]^{n+\frac{1}{2}} + \left[C\left(\delta_x u_y + \delta_y u_x - \delta_x\delta_y u\right)\right]^{n+\frac{1}{2}} \\
&+ D^{n+\frac{1}{2}}\delta_y^2 u_{xy}^{n+\frac{1}{2}} + E^{n+\frac{1}{2}}\delta_y^2 u_x^{n+\frac{1}{2}} + F^{n+\frac{1}{2}}u_x^{n+\frac{1}{2}} + G^{n+\frac{1}{2}}u_y^{n+\frac{1}{2}} - \left(R^{n+\frac{1}{2}}\delta_x + S^{n+\frac{1}{2}}\delta_y\right)u^{n+\frac{1}{2}}
\end{aligned}$$

$$= \alpha u_t^{n+\frac{1}{2}} - f^{n+\frac{1}{2}} - \frac{p^{n+\frac{1}{2}}h^2}{6}\left(\alpha u_{xt}^{n+\frac{1}{2}} - f_x^{n+\frac{1}{2}}\right) - \frac{q^{n+\frac{1}{2}}h^2}{6}\left(\alpha u_{yt}^{n+\frac{1}{2}} - f_y^{n+\frac{1}{2}}\right) \tag{4.8}$$

Employing the Crank-Nicolson time discretization, i.e., all function and their derivatives at the time level $t^{n+\frac{1}{2}}$ are computed by average values of t^{n+1} and t^n, for instance, $u^{n+1/2} = \frac{1}{2}\left(u^n + u^{n+1}\right)$, etc. Both sides of the equation are multiplied by two, and the above equation (4.8) can be simplified as

$$\begin{aligned}
&\left(-\frac{4\bar{A}}{h^2} - \frac{4\bar{B}}{h^2} - \frac{2\alpha}{\tau}\right)u_0^{n+1} + \left(\frac{2\bar{A}}{h^2} - \frac{p_0^{n+1}+p_0^n}{4h}\right)u_1^{n+1} + \left(\frac{2\bar{B}}{h^2} - \frac{q_0^{n+1}+q_0^n}{4h}\right)u_2^{n+1} \\
&+ \left(\frac{2\bar{A}}{h^2} + \frac{p_0^{n+1}+p_0^n}{4h}\right)u_3^{n+1} + \left(\frac{2\bar{B}}{h^2} + \frac{q_0^{n+1}+q_0^n}{4h}\right)u_4^{n+1} + \frac{\bar{C}}{4h^2}\left(-u_5^{n+1} + u_6^{n+1} - u_7^{n+1} + u_8^{n+1}\right) \\
&+ \left[-\frac{2\bar{E}}{h^2} + \bar{F} + \frac{\alpha h^2\left(p_0^{n+1}+p_0^n\right)}{12\tau}\right]u_{x0}^{n+1} - \frac{\bar{A}}{2h}u_{x1}^{n+1} + \left(\frac{\bar{C}}{2h} + \frac{\bar{E}}{h^2}\right)u_{x2}^{n+1} + \frac{\bar{A}}{2h}u_{x3}^{n+1} \\
&+ \left(-\frac{\bar{C}}{2h} + \frac{\bar{E}}{h^2}\right)u_{x4}^{n+1} + \left(-\frac{2\bar{D}}{h^2} + \bar{G} + \frac{\alpha h^2\left(q_0^{n+1}+q_0^n\right)}{12\tau h}\right)u_{y0}^{n+1} + \left(\frac{\bar{C}}{2h} + \frac{\bar{D}}{h^2}\right)u_{y1}^{n+1} \\
&- \frac{\bar{B}}{2h}u_{y2}^{n+1} + \left(-\frac{\bar{C}}{2h} + \frac{\bar{D}}{h^2}\right)u_{y3}^{n+1} + \frac{\bar{B}}{2h}u_{y4}^{n+1} \\
&= \left(\frac{4\bar{A}}{h^2} + \frac{4\bar{B}}{h^2} - \frac{2\alpha}{\tau}\right)u_0^n + \left(-\frac{2\bar{A}}{h^2} + \frac{p_0^{n+1}+p_0^n}{4h}\right)u_1^n + \left(-\frac{2\bar{B}}{h^2} + \frac{q_0^{n+1}+q_0^n}{4h}\right)u_2^n \\
&- \left(\frac{2\bar{A}}{h^2} + \frac{p_0^{n+1}+p_0^n}{4h}\right)u_3^n - \left(\frac{2\bar{B}}{h^2} + \frac{q_0^{n+1}+q_0^n}{4h}\right)u_4^n + \frac{\bar{C}}{4h^2}\left(u_5^n - u_6^n + u_7^n - u_8^n\right) \\
&+ \left[\frac{2\bar{E}}{h^2} - \bar{F} + \frac{\alpha h^2\left(p_0^{n+1}+p_0^n\right)}{12\tau}\right]u_{x0}^n + \frac{\bar{A}}{2h}u_{x1}^n - \left(\frac{\bar{C}}{2h} + \frac{\bar{E}}{h^2}\right)u_{x2}^n - \frac{\bar{A}}{2h}u_{x3}^n \\
&+ \left(\frac{\bar{C}}{2h} - \frac{\bar{E}}{h^2}\right)u_{x4}^n + \left(\frac{2\bar{D}}{h^2} - \bar{G} + \frac{\alpha h^2\left(q_0^{n+1}+q_0^n\right)}{12\tau h}\right)u_{y0}^n - \left(\frac{\bar{C}}{2h} + \frac{\bar{D}}{h^2}\right)u_{y1}^n + \frac{\bar{B}}{2h}u_{y2}^n \\
&+ \left(\frac{\bar{C}}{2h} - \frac{\bar{D}}{h^2}\right)u_{y3}^n - \frac{\bar{B}}{2h}u_{y4}^n + \frac{\left(p_0^{n+1}+p_0^n\right)\left(f_{x0}^{n+1}+f_{x0}^n\right)h^2}{12} \\
&+ \frac{\left(q_0^{n+1}+q_0^n\right)\left(f_{y0}^{n+1}+f_{y0}^n\right)h^2}{12} - \left(f_0^{n+1}+f_0^n\right)
\end{aligned} \tag{4.9}$$

In which

$$\begin{aligned}
\bar{A} &= 1 + \frac{\left(p^n + p^{n-1}\right)^2 h^2}{24}, \\
\bar{B} &= 1 + \frac{\left(q^n + q^{n-1}\right)^2 h^2}{24}, \\
\bar{C} &= \frac{\left(p^n + p^{n-1}\right)\left(\beta_x^n + \beta_x^{n-1}\right)h^2}{24} + \frac{\left(p^n + p^{n-1}\right)\left(q^n + q^{n-1}\right)h^2}{12}, \\
&\quad + \frac{\left(q^n + q^{n-1}\right)\left(\beta_y^n + \beta_y^{n-1}\right)h^2}{24} - \frac{\beta^n + \beta^{n-1}}{2},
\end{aligned}$$

$$\bar{D}=\frac{\left(p^{n}+p^{n-1}\right)\left(\beta^{n}+\beta^{n-1}\right)h^{2}}{24}-\frac{\left(q^{n}+q^{n-1}\right)h^{2}}{12},$$
$$\bar{E}=\frac{\left(q^{n}+q^{n-1}\right)\left(\beta^{n}+\beta^{n-1}\right)h^{2}}{24}-\frac{\left(p^{n}+p^{n-1}\right)h^{2}}{12},$$
$$\bar{F}=\frac{\left(p^{n}+p^{n-1}\right)\left(p_{x}^{n}+p_{x}^{n-1}\right)h^{2}}{24}+\frac{\left(q^{n}+q^{n-1}\right)\left(p_{y}^{n}+p_{y}^{n-1}\right)h^{2}}{24},$$
$$\bar{G}=\frac{\left(p^{n}+p^{n-1}\right)\left(q_{x}^{n}+q_{x}^{n-1}\right)h^{2}}{24}+\frac{\left(q^{n}+q^{n-1}\right)\left(q_{y}^{n}+q_{y}^{n-1}\right)h^{2}}{24},$$
$$\bar{R}=\frac{p^{n}+p^{n-1}}{2},\quad \bar{S}=\frac{q^{n}+q^{n-1}}{2}.$$

Where $n=0,1$ and Eq. (4.9) is the second-order accurate in time and the fourth-order accurate in space. Next, in order to obtain value on the third time level, employing the BDF3 time discretization, we can get

$$\begin{aligned}
&\left(-\frac{4A}{h^{2}}-\frac{4B}{h^{2}}-\frac{11\alpha}{6\tau}\right)u_{0}^{n+1}+\left(\frac{2A}{h^{2}}-\frac{p_{0}}{2h}\right)u_{1}^{n+1}+\left(\frac{2B}{h^{2}}-\frac{q_{0}}{2h}\right)u_{2}^{n+1}+\left(\frac{2A}{h^{2}}+\frac{p_{0}}{2h}\right)u_{3}^{n+1}\\
&+\left(\frac{2B}{h^{2}}+\frac{q_{0}}{2h}\right)u_{4}^{n+1}+\frac{C}{4h^{2}}\left(-u_{5}^{n+1}+u_{6}^{n+1}-u_{7}^{n+1}+u_{8}^{n+1}\right)+\left(-\frac{2E}{h^{2}}+F+\frac{11\alpha h^{2}p_{0}}{36\tau}\right)u_{x0}^{n+1}\\
&-\frac{A}{2h}u_{x1}^{n+1}+\left(\frac{C}{2h}+\frac{E}{h^{2}}\right)u_{x2}^{n+1}+\frac{A}{2h}u_{x3}^{n+1}+\left(-\frac{C}{2h}+\frac{E}{h^{2}}\right)u_{x4}^{n+1}+\left(\frac{C}{2h}+\frac{D}{h^{2}}\right)u_{y1}^{n+1}\\
&+\left(-\frac{2D}{h^{2}}+G+\frac{11\alpha h^{2}q_{0}}{36\tau}\right)u_{y0}^{n+1}-\frac{B}{2h}u_{y2}^{n+1}+\left(-\frac{C}{2h}+\frac{D}{h^{2}}\right)u_{y3}^{n+1}+\frac{B}{2h}u_{y4}^{n+1}\\
&=-\frac{3\alpha}{\tau}u_{0}^{n}+\frac{\alpha h^{2}p_{0}}{2\tau}u_{x0}^{n}+\frac{\alpha h^{2}q_{0}}{2\tau}u_{y0}^{n}+\frac{3\alpha}{2\tau}u_{0}^{n-1}-\frac{\alpha h^{2}p_{0}}{4\tau}u_{x0}^{n-1}-\frac{\alpha h^{2}q_{0}}{4\tau}u_{y0}^{n-1}\\
&-\frac{3\alpha}{\tau}u_{0}^{n-2}+\frac{\alpha h^{2}p_{0}}{18\tau}u_{x0}^{n-2}+\frac{\alpha h^{2}q_{0}}{18\tau}u_{y0}^{n-2}+\frac{p_{0}h^{2}}{6}f_{x0}^{n+1}+\frac{q_{0}h^{2}}{6}f_{y0}^{n+1}-f_{0}^{n+1}
\end{aligned}\tag{4.10}$$

Where $n=2$ and Eq. (4.10) is the third-order accurate in time and the fourth-order accurate in space. In which, values of p,q and β are computed at the time level t^{n+1}. Notice that when the BDF schemes are applied to the parabolic equation, according to previous literature [35], the use of a second-order accuracy scheme to initialize a fourth-order accuracy scheme appears not to reduce the overall order of accuracy. The following numerical experimental results also fully illustrate this point.

5 Mulitgrid method

To achieve high stability, a fully implicit method for time discretization of Equation (4.3) may be adopted. This will result in a system of algebraic equations that is sparse, which may require a large amount of computational effort. Although the algebraic systems associated with the currently proposed finite difference approximation formulation can be solved by using the conventional iterative methods (such as the Gauss-Seidel or successive over relaxation (SOR) methods etc.), the simple iteration method may result in slow convergence and is hence very time consuming. To improve the iteration convergence, we will choose multigrid method instead of the conventional iterative methods. The multigrid method, which is independent of the grid size, has been proved

to be among the fastest and most efficient iterative algorithms for solving linear systems arising from discretizing a wide class of partial differential equations. During a long time, its combination with HOC difference schemes [37–46] has been proved to be high efficient. The basic idea of multigrid method is to approximate the smooth (long wavelength) component of the error on coarser grids. The non-smooth component is reduced with a small number of iterations with a conventional iterative method on the fine grid. Restriction, interpolation and relaxation operators are three important components in multigrid method and only when they are suitably chosen and match well each other, can the efficiency of multigrid method get to optimal extend. The detailed description about multigrid method, readers are referred to [47, 48] and the references therein. In this paper, we select the alternating direction line Gauss-Seidel (ALGS) relaxations to remove the residuals on each coarse grid, the bi-linear interpolation to interpolate the coarse grid correction from the coarse grid back to the fine grid and a full weighting restriction operator to restriction the residuals to the coarse grid. In this paper, a multigrid V(1,1) algorithm is adopted, and Algorithm 1 for elliptic equation and Algorithm 2 for parabolic equation are described as following:

Algorithm 1

Step 1: The first ALGS iteration is done on the finest grid layer based on the initial guess value.
Step 2: The residual on the fine grid layer is to be calculated and restricted to the adjacent coarse grid layer with the restriction operator.
Step 3: An error "polishing" (called pre-smoothing) is to be carried out with ALGS iteration on the coarse grid layer, then the problem of the coarse grid is solved.
Step 4: The results of the residual correction on the coarse grid layer are sent back to the adjacent fine grid layer through the interpolation operator.
Step 5: An error "polishing" (called after-smoothing) is carried out with ALGS iteration on the fine grid layer.
Step 6: Repeat steps (2)-(5) until the residual reaches the convergence criterion.

Algorithm 2

Step 1: As to the time level n, the first ALGS iteration is done on the finest grid layer based on the calculated value on the $n-1$ time level. If $n-1$, start the iteration with initial value.
Step 2: The residual on the fine grid layer is to be calculated and restricted to the adjacent coarse grid layer with the restriction operator.
Step 3: An error "polishing" (called pre-smoothing) is to be carried out with ALGS iteration on the coarse grid layer, then the problem of the coarse grid is solved.
Step 4: The results of the residual correction on the coarse grid layer are sent back to the adjacent fine grid layer through the interpolation operator.
Step 5: An error "polishing" (called after-smoothing) is carried out with ALGS iteration on the fine grid layer.
Step 6: Repeat steps (2)-(5) until the residual reaches the convergence criterion.
Step 7: Set $n = n+1$, then, the calculation turns to the next time level. Repeat the above procedure until the calculation reaches the final time moment.

6 Numerical experiments

In this section, we perform numerical experiments to illustrate the high-order accuracy and high efficiency of the present fourth-order compact scheme and multigrid method. All problems are defined on the rectangular domain. We use multigrid method to solve the resulting linear systems in all test problems. While Karaa [12] used GMRES preconditioned by $ILU(\tau, s)$ and Sen [30] used BiCGStab algorithm to solve the resulting linear systems. All iterative procedures are started with zero initial guesses and are terminated when the Euclidean norm of the residual vector is reduced by 10^{-10}. All calculations are done on a personal computer with an Intel(R) core(TM) i3-5005U double 2.0GHz CPU and 4 GB memory. The code is written in Fortran 77 programming language with 64-bit double precision arithmetic.

We employ the standard coarsening technique (the mesh size of the coarse grid doubles that of the fine grid). The rate of convergence of the method is computed with the following definition

$$\text{Order } = \frac{\log(\text{Error 1/ Error 2})}{\log(h_1/h_2)},$$

where Error 1 and Error 2 are the errors with a certain norm estimated for two different grid sizes h_1 and h_2

6.1 Problem 1: Stommel Ocean Model [30, 34]

Firstly, we chose ocean model problem which was used by Stommel to explain intense crowding of streamlines towards the western border of a homogenous rectangular ocean [30,34]. The streamlines are the solutions of the steady equation in the model problem

$$\psi_{xx} + \psi_{yy} + \alpha\psi_x = -\gamma \sin\left(\frac{\pi}{b}y\right), \quad 0 \le x \le \lambda,\ 0 \le y \le b$$

with analytic solution:

$$\psi(x, y) = -\gamma\left(\frac{b}{\pi}\right)\sin\left(\frac{\pi}{b}y\right)\left(pe^{Ax} + qe^{Bx} - 1\right),$$

where

$$A, B = -\frac{\alpha}{2} \pm \sqrt{\frac{\alpha^2}{4} + \left(\frac{\pi}{b}\right)^2},\ p = \frac{1 - e^{B\lambda}}{e^{A\lambda} - e^{B\lambda}},\quad q = 1 - p.$$

The details description of the parameters α, γ, λ and b can be found in [30, 34] . The dimensionless physical parameters are selected as [30]

$$\lambda = 10, b = 2\pi, \gamma = 25, \alpha = 0 \text{ (or } \alpha = \frac{10}{3}, 10)$$

We use the present fourth-order CD scheme with exact boundary condition (Present 1), and with fourth-order boundary scheme (Present 2), and the Sen's scheme [30] with exact boundary condition to compute the numerical solutions of Problem 1. Table1 gives the L_1 errors and the convergence orders for Problem 1. Fig. 2 gives the exact solution, the numerical solution of the present scheme, the L_1 errors of the present scheme and the Sen's scheme for Problem 1. Fig. 3 displays the convergence of the present scheme and the Sen's scheme for Problem 1. It is shown

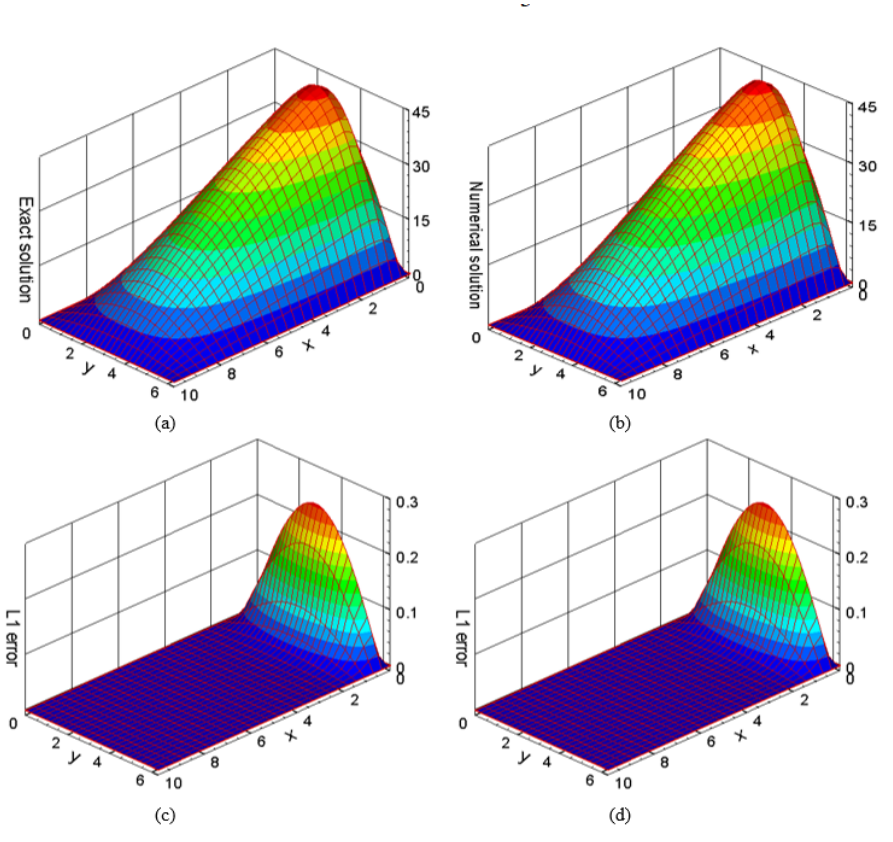


FIG. 2. When $\alpha = 10/3$: (a) Exact solution; (b) Numerical solution, (c) The L_1 error of the present scheme; (d) The L_1 error of the Karaa's scheme for Problem 1 with 20×20 grid

that when $\alpha = 0$ and $\alpha = 10/3$, the L_1 errors and convergence orders of the present scheme and the Sen's scheme [30] are almost identical. However, when $\alpha = 10$, the Sen's scheme [30] on a relatively coarse meshes ($N = 14$ and $N = 19$) is divergent, while the present scheme is still convergent. When $N = 27$ and $N = 33$, they are both convergent, while the present scheme produces slightly more accurate solution.

6.2 Problem 2: Steady-state Problem [12]

Consider the following differential equation in the presence of source term

$$u_{xx} + u_{yy} + \beta(x,y)u_{xy} + p(x,y)u_x + q(x,y)u_y = f(x,y), \quad 0 < x, y < 1,$$
$$p(x,y) = -\operatorname{Re} y, \ q(x,y) = \operatorname{Re} x, \ \beta(x,y) = (1-x)(1-y)e^{x+y},$$

where the exact solution is

$$u(x,y) = xy\left(x^2 + y^2\right).$$

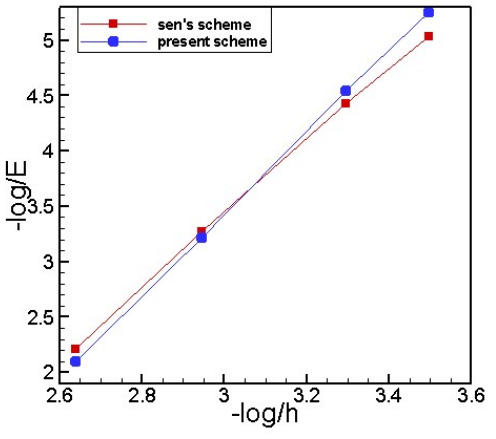


FIG. 3. Convergence of error at $\alpha = 10/3$ for Problem 1

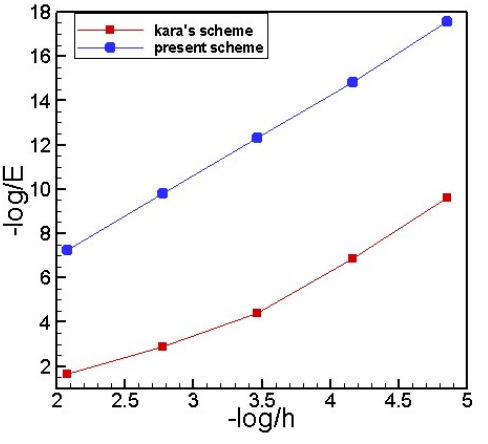


FIG. 5. Convergence of error at $Re = 10000$ for Problem 3

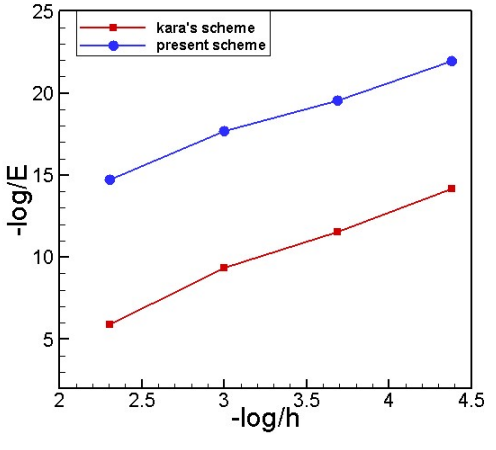


FIG. 7. Convergence of error at $Re = 1000$, $T = 1$, $\tau = h$ and $\tau = h^2$ for Problem 4

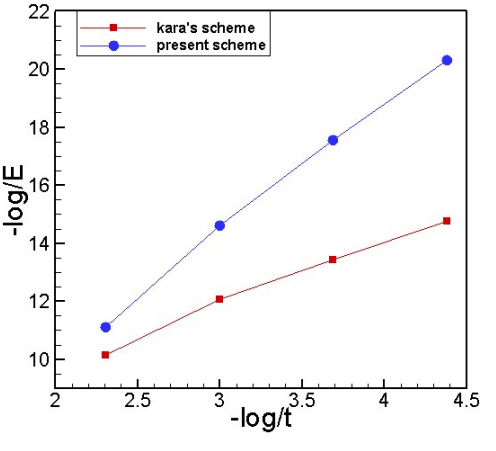


FIG. 8. Convergence of error at $Re = 10$, $T = 1$, $h = 0.02$ for Problem 4

The Dirichlet boundary condition and the source term is determined by the exact solution. The problem is solved on a unit square $\Omega = (0,1) \times (0,1)$. Re is referred to as the Reynolds number. In our experiments, Re is allowed to vary between 1 and 10^6.

We use the present scheme, the Sen's scheme [30] and the Karaa's scheme [12] to compute the numerical solutions of Problem 2. The maximum absolute errors with different values of h and Re are listed in Table 2. We can clearly see that when Re $\leq 10^2$, the maximum absolute errors of

TABLE 1. The L_1 error and convergence order for Problem 1

α	Cell number	Sen's scheme [30]		Present 1		Present 2	
		L_1 error	Order	L_1 error	Order	L_1 error	Order
0	9×9	1.71(-03)		1.72(-03)		1.88(-02)	
	10×10	1.23(-03)	3.13	1.24(-03)	3.11	1.16(-02)	4.59
	14×14	4.08(-04)	3.28	4.31(-04)	3.14	2.44(-03)	4.63
10/3	14×14	1.10(-01)		1.23(-01)		2.29(-01)	
	19×19	3.97(-02)	3.33	4.03(-02)	3.66	8.24(-02)	3.35
	27×27	1.19(-02)	3.43	1.07(-02)	3.77	2.12(-02)	3.86
10	14×14	div		9.09(-01)		6.67(-01)	
	19×19	div	-	3.33(-01)	3.29	3.20(-01)	2.41
	27×27	1.43(-01)	-	1.00(-01)	3.42	1.26(-01)	2.65
	33×33	6.34(-02)	4.05	4.94(-02)	3.51	4.98(-02)	4.67

the present scheme and the Sen's scheme [30] are almost identical. With the increase of Reynolds number, the present scheme produces much better solution while the Sen's scheme is divergent. On the other hand, we find that the present scheme is hardly influenced by Re. What's more, the present scheme on a relatively coarse mesh ($N = 8$) produces much better solution than the Karaa's scheme [12] on the fine mesh ($N = 128$) for all Re number. With the increase of Reynolds number, the maximum absolute error of the karaa's scheme gradually decrease, and fails to preserve its fourth-order accuracy. Fig. 4 displays the exact solution, the numerical solution of the present scheme, the absolute errors of the present scheme and the Karaa's scheme at Re = 1000 and 16×16 grids for Problem 2. It can be seen that the numerical results of the present scheme are in good agreement with the exact solution. And the absolute error of the present scheme is obviously much smaller than those of the Karaa's scheme.

TABLE 2. The maximum absolute error for different Re for Problem 2

Re	Sen's scheme [30]		Present scheme		Karaa' s scheme [12]	
	$h_{1/8}$	$h_{1/16}$	$h_{1/8}$	$h_{1/16}$	$h_{1/64}$	$h_{1/128}$
1	2.44(-14)	5.77(-15)	1.72(-15)	3.63(-15)	6.02(-10)	5.23(-11)
10	8.76(-15)	1.40(-15)	3.22(-15)	2.24(-15)	1.69(-08)	1.03(-09)
10^2	2.68(-13)	8.30(-14)	5.55(-16)	1.11(-15)	3.65(-07)	2.28(-08)
10^3	div	div	3.61(-16)	4.49(-15)	4.53(-06)	2.84(-07)
10^4	div	div	1.30(-15)	3.72(-15)	3.30(-05)	2.86(-06)
10^5	div	div	1.55(-15)	2.94(-15)	5.19(-05)	1.22 (-05)
10^6	div	div	1.28(-15)	3.39(-15)	5.24(-05)	1.37(-05)

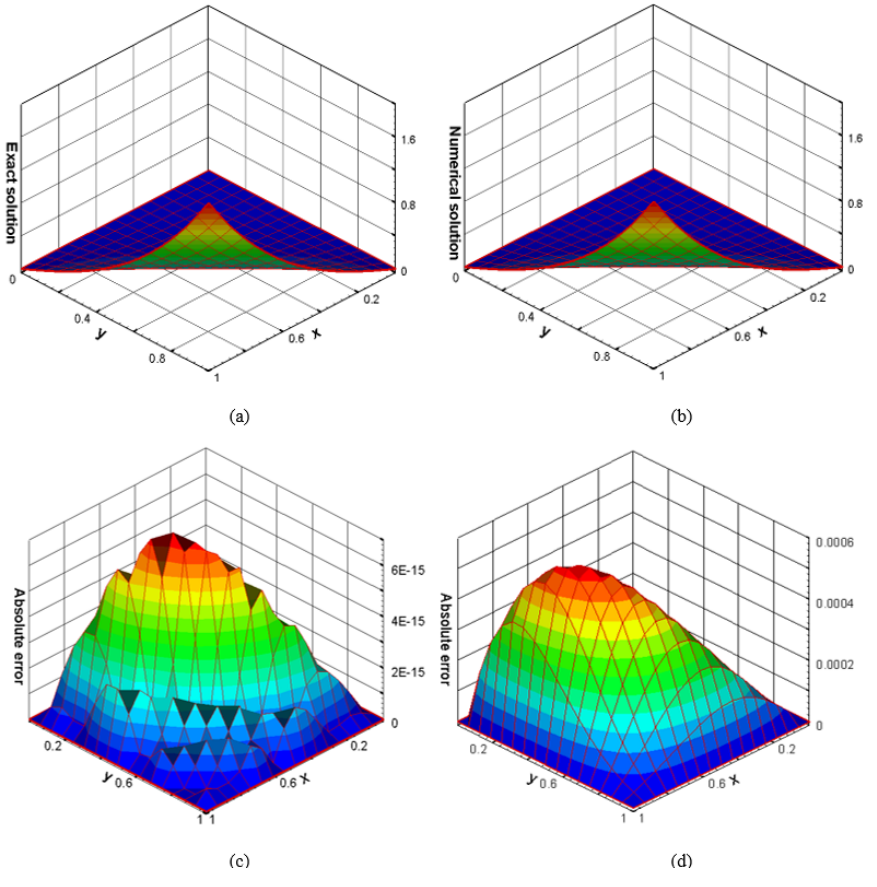


FIG. 4. When $Re = 1000$: (a) Exact solution ; (b) Numerical solution, (c) The absolute error of the present scheme; (d) The absolute error of the Karaa's scheme for Problem 2 with 16 $\times$16 grids

6.3 Problem 3: Steady-state Problem [12]

Consider the following differential equation in the presence of source term

$$u_{xx} + u_{yy} + \beta(x,y)u_{xy} + p(x,y)u_x + q(x,y)u_y = f(x,y),\ 0 < x,y < 1,$$
$$p(x,y) = -\operatorname{Re}(1-x)(1-2y),\ q(x,y) = 4\operatorname{Re} xy(1-y),\ \beta(x,y) = \cos(\pi x)\cos(\pi y),$$

where the exact solution is

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

For comparison, we also use the present scheme, the Sen's scheme [30] and the Karaa 's scheme [12] to compute the numerical solutions of Problem 3. The maximum absolute errors and estimated accuracy orders with different values of h and Re are listed in Table 3 and Table 4 respectively.

The numerical results of Table 3 show clearly that when Re ≤ 10, the maximum absolute errors of the present scheme and the Sen's scheme [30] are almost identical. With the increase of Reynolds number, the present scheme succeeds to maintain its fourth-order accuracy while the Sen's scheme is divergent. Table 4 gives the numerical results computed by the present scheme and by the Karaa's scheme. We find that both schemes show the convergence rates of $O\left(h^4\right)$ in space, but the present scheme still yields much better solution than the Karaa's scheme. Both the present scheme and the Karaa's scheme obtain the fourth-order accuracy when Re $\leq 10^4$. However, when Re $\geq 10^5$, the convergence order of the present scheme just drops a little to 3.77 and 3.74 while the Karaa's scheme drops a lot to 2.75 and 1.90. On the other hand, we notice that with the increase of Re number, the computed accuracy (the maximum absolute error) of the Karaa's scheme deteriorates greatly. while the present scheme does not drop distinctly. Fig. 5 displays the convergence of the present scheme and the Karaa's scheme at Re=10000 for Problem 3. Fig. 6 gives the exact solution, the numerical solution and the absolute error of the present scheme and the absolute error of the Karaa's scheme with 64×64 grids for Problem 3. Fig. 5 and Fig. 6 once again confirm that the present scheme for elliptic equations is obviously better than the Karaa's scheme.

Table 3. The maximum absolute error and convergence order for different *Re* for Problem 3

Re	Sen's scheme [30]			Present scheme		
	$h_{1/8}$	$h_{1/16}$	Order	$h_{1/8}$	$h_{1/16}$	Order
1	5.45(-04)	1.51(-05)	5.17	1.05(-04)	7.58(-06)	3.79
10^1	5.32(-04)	1.36(-05)	5.29	2.20(-04)	1.64(-05)	3.75
10^2	div	div	–	8.05(-04)	6.67(-05)	3.59
10^3	div	div	–	5.20(-04)	6.08(-05)	3.10
10^4	div	div	–	6.01(-04)	4.77(-05)	3.66
10^5	div	div	–	6.17(-04)	4.98(-05)	3.63
10^6	div	div	–	6.18(-04)	5.01(-05)	3.62

Table 4. The maximum error and convergence order for different *Re* for Problem 3

Re	Karaa's scheme [12]			Present scheme		
	$h_{1/64}$	$h_{1/128}$	Order	$h_{1/64}$	$h_{1/128}$	Order
1	7.78(-08)	4.86(-09)	4.00	3.61(-8)	2.16(-9)	4.06
10^1	5.57(-07)	3.48(-08)	4.00	7.36(-8)	4.53(-9)	4.02
10^2	1.12(-05)	6.99(-07)	4.00	2.78(-7)	1.73(-8)	4.01
10^3	1.08(-04)	6.78(-06)	3.99	3.41(-7)	2.09(-8)	4.03
10^4	1.05(-03)	6.65(-05)	3.98	3.69(-7)	2.27(-8)	4.02
10^5	4.03(-03)	6.01(-04)	2.75	3.27(-7)	2.40(-8)	3.77
10^6	4.26(-03)	1.14(-03)	1.90	3.40(-7)	2.54(-8)	3.74

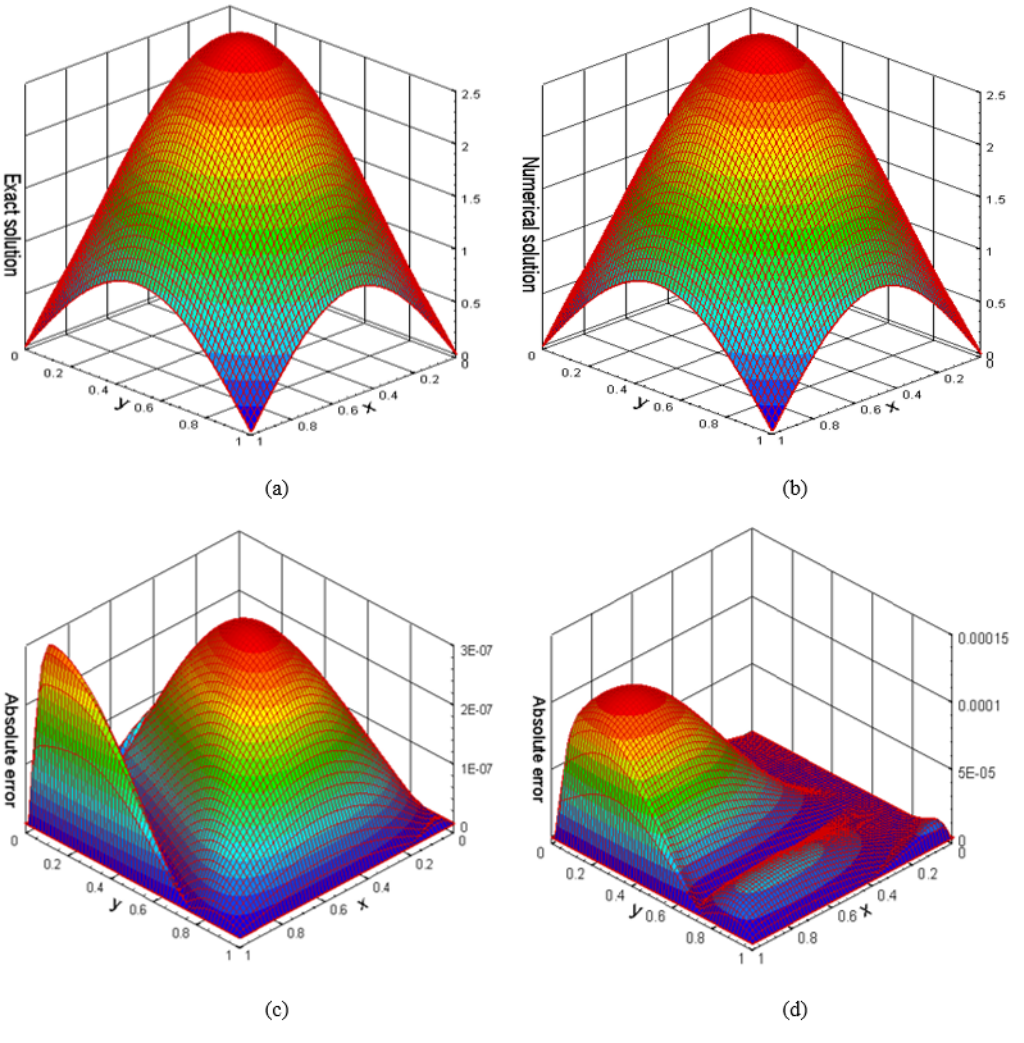


FIG. 6. When $Re = 10000$: (a) Exact solution ; (b) Numerical solution, (c) The absolute error of the present scheme; (d) The absolute error of the Karaa's scheme for Problem 3 with 64×64 grids

6.4 Problem 4: Time-dependent Problem [12]

We consider the two-dimensional parabolic problem

$$u_t - (u_{xx} + u_{yy}) + \beta(x,y,t)u_{xy} + p(x,y,t)u_x + q(x,y,t)u_y = f(x,y,t),\ 0 < x,y < 1,\ t > 0,$$
$$p(x,y) = \operatorname{Re} x(1-y),\ q(x,y) = -\operatorname{Re} y,\ \beta(x,y) = (1-x)(1-y)e^{x+y},$$

where $T = 1$ and $\Omega = (0,1) \times (0,1)$ is the square region. For which the exact solution is

$$u(x,y) = e^{-\pi t}\left(x^2 - y^2\right)\cosh(x+y).$$

The initial, Dirichlet boundary condition and the source term are directly taken from this exact solution. The numerical solutions, using the present fourth-order CD scheme and the Karaa's scheme [12] , are obtained under uniform grid with different mesh sizes. We compare the maximum absolute error, convergence order and CPU time. To confirm the theoretical accuracy $O\left(\tau^4+h^4\right)$, we first solve the problem with a fixed mesh size $h = 1/50$, Re $= 10$ and various time-step sizes. The numerical results are presented in Table 5. We find that the present scheme yields fourth-order convergence, while the Karaa's scheme [12] just shows second-order convergence. We also notice that the maximum absolute errors of the present scheme on 20×20 grid are almost same as those using the Karaa's scheme [12] on 80×80 grid. To investigate the convergence rate in space, we set the time step size $\tau = 1/100$ and let h varies from 1/10 to 1/80 in Table 6. Since the present scheme is fourth- order accuracy in both time and space, $\tau = 1/100$ is small enough and will not affect the spatial accuracy. On the other hand, because the Karaa's scheme is just second-order accuracy in time, it needs much smaller time step size to guarantee spatially fourth-order accuracy can be obtained. So, $\tau = 1/1000$ was set in Ref. [12] for this case. We notice that both schemes obtain fourth-order convergence, but the present scheme yields more accurate solutions than the karaa's scheme does.

TABLE 5. Numerical results at $T = 1$, $Re = 10$ with $h = 0.02$ and different τ for Problem 4

τ	Karaa's scheme [12]			Present scheme		
	L_∞ error	Order	CPU time(s)	L_∞ error	Order	CPU time(s)
1/10	3.88(-05)		0.858	1.48(-05)		7.940
1/20	5.77(-06)	2.75	1.654	4.46(-07)	5.05	14.305
1/40	1.49(-06)	1.96	3.167	2.42(-08)	4.20	27.034
1/80	3.87(-07)	1.94	6.240	1.53(-09)	3.98	52.603

TABLE 6. Numerical results at $T = 1$, $Re = 10$ and different h for Problem 4

h	Karaa' s scheme ($\tau = 1/1000$) [12]			Present scheme ($\tau = 1/100$)		
	L_∞ error	Order	CPU time(s)	L_∞ error	Order	CPU time(s)
1/10	1.69(-05)		2.574	5.84(-07)		1.700
1/20	1.09(-06)	3.96	10.156	3.27 (-08)	4.16	8.674
1/40	6.91(-08)	3.97	45.256	2.02(-09)	4.02	39.983
1/80	5.79(-09)	3.58	197.496	1.67(-10)	3.60	175.532

In the following, we carry out computation with different mesh sizes for different Re and compare the accuracy of the computed solutions by the present scheme with $\tau = h$ and the Karaa's scheme with $\tau = h^2$ in both time and space. The comparisons of numerical results are given in Tables 7-9. It is seen that for all selected Re, the present scheme and the Karaa's scheme [12] can maintain their fourth-order accuracy. However, when Re $= 10$, the present scheme yields slightly better accurate solution than that of the Karaa's scheme [12] . And when Re $= 100$ and Re $= 1000$, the present scheme yields much better solution and expends much less CPU time than the Karaa's

TABLE 7. Numerical results at $T = 1$, $Re = 10$ and different h for Problem 4

h	Karaa's scheme ($\tau = h^2$) [12]			Present scheme ($\tau = h$)		
	L_∞ error	Order	CPU time(s)	L_∞ error	Order	CPU time(s)
1/10	1.77(-05)		0.265	1.42(-05)		0.265
1/20	1.11(-06)	4.00	4.088	4.18(-07)	5.09	1.731
1/40	6.93(-08)	4.00	68.545	2.33(-08)	4.17	14.664
1/80	4.33(-09)	4.00	1139.239	1.36(-09)	4.10	123.116

TABLE 8. Numerical results at $T = 1$, $Re = 100$ and different h for Problem 4

h	Karaa's scheme ($\tau = h^2$) [12]			Present scheme ($\tau = h$)		
	L_∞ error	Order	CPU time(s)	L_∞ error	Order	CPU time(s)
1/10	2.17(-04)		0.249	1.64(-06)		0.530
1/20	1.68(-05)	3.69	4.165	9.77(-08)	4.07	2.512
1/40	1.10(-06)	3.93	69.404	7.46(-09)	3.71	19.391
1/80	6.92(-08)	3.99	1150.976	4.58(-10)	4.03	153.972

TABLE 9. Numerical results at $T = 1$, $Re = 1000$ and different h for Problem 4

h	Karaa's scheme ($\tau = h^2$) [12]			Present scheme ($\tau = h$)		
	L_∞ error	Order	CPU time(s)	L_∞ error	Order	CPU time(s)
1/10	2.80(-03)		0.374	4.10(-07)		1.560
1/20	8.91(-05)	4.97	4.649	2.12(-08)	4.27	21.044
1/40	9.61(-06)	3.21	75.754	3.35(-09)	2.66	99.840
1/80	7.20(-07)	3.74	1249.014	2.98(-10)	3.49	405.659

scheme [12]. Fig. 7 and Fig. 8 display the convergence of the present scheme and the karaa's scheme with different Re and h for Problem 4, respectively. Fig. 9 displays the exact solution, the numerical solution of the present scheme, the absolute errors of the present scheme and the Karaa's scheme for Problem 4. We can clearly see that the present scheme for parabolic equations is better than the Karaa's scheme.

7 Conclusion

In this paper, we propose a 9-point fourth-order CD scheme for the 2D linear elliptic equation with a mixed derivative. The scheme contains the representations of the spatial first-derivatives of the unknown function, which are computed by the fourth-order Padé schemes separately. With truncation error analyses, the present scheme can reach fourth-order accuracy for the interior gird points and has relatively small coefficients amplitude for higher leading truncation error terms. Consequently, for those problems with large first-order derivatives, the present scheme can get more accurate solution than those in the literature. Then, this method is extended to solve the 2D parabolic equation

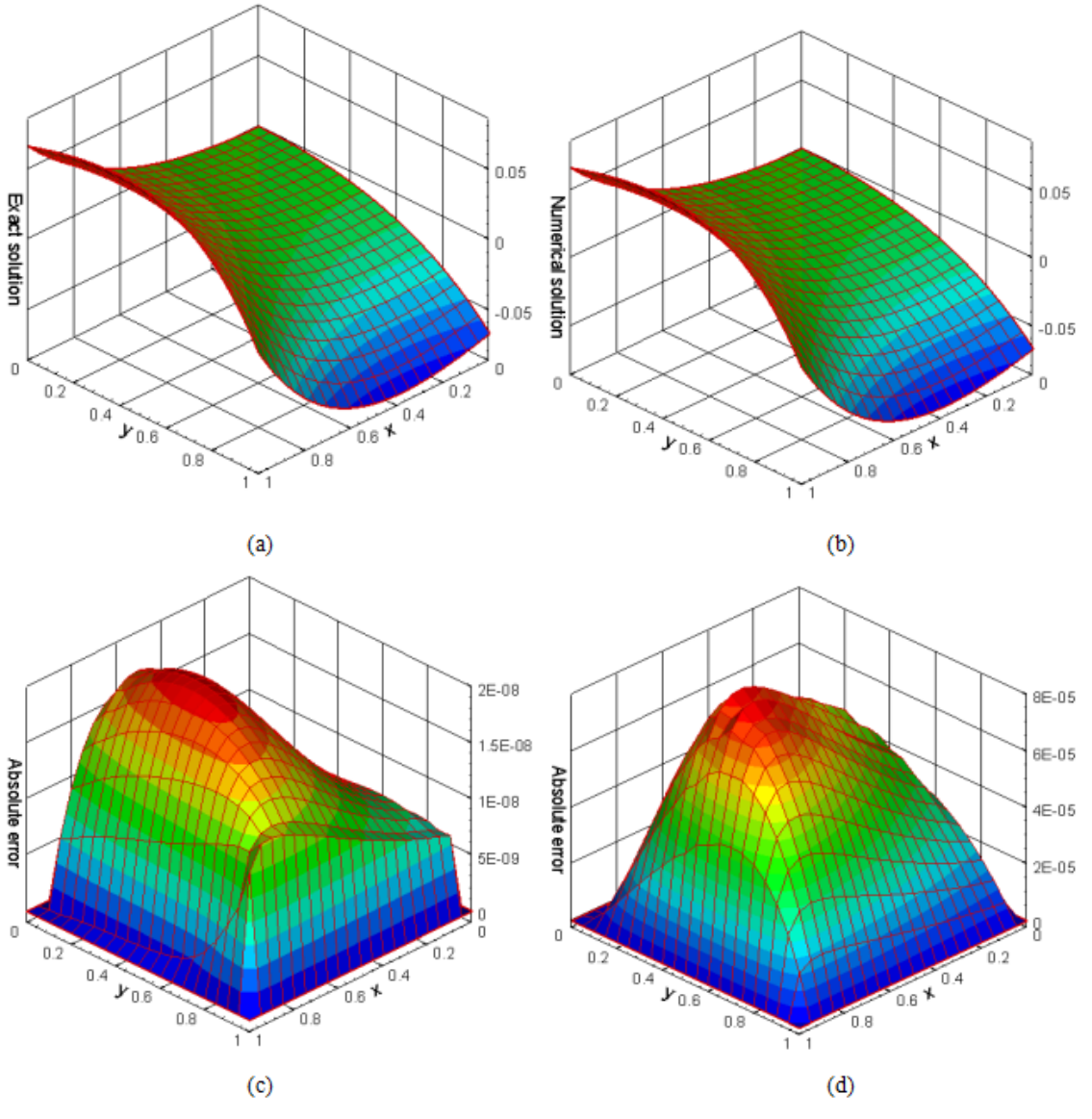


FIG. 9. When $T = 1, Re = 100, \tau = h$ (or $\tau = h^2$): (a) Exact solution; (b) Numerical solution, (c) The absolute error of the present scheme; (d) The absolute error of the Karaa's scheme for Problem 4 with 20×20 grid

with a mixed derivative. The backward Euler formula with fourth-order accuracy is employed to discrete temporal derivative, so a full implicit fourth-order accuracy compact difference scheme is derived for the parabolic equation (1.1). The scheme arrives fourth-order accuracy in both space

and time. The multigrid method is employed to solve the large-scale linear systems arising from the present difference schemes. Finally, numerical experiments are performed to exhibit the good performance of the present schemes. It demonstrates that the present schemes obtain fourth-order accurate solutions and are more accurate and more efficient than those in the literature.

Data Availability

The data sets generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgements

The authors would like to thank the editors and the anonymous referees whose constructive comments are helpful to improve the quality of this paper. This work was supported in part by the National Natural Science Foundation of China under Grants 11772165 and 11961054, the National Natural Science Foundation of Ningxia under Grant 2018AAC02003 and 2020AAC03264, the National Key Research and Development Program of Ningxia under Grant 2018BEE03007, the Scientific Research Program in Higher Institution of Ningxia under Grant NGY2020110, Major Innovation Projects for Building First-class Universities in China's Western Region under Grant ZKZD2017009 and Western First-Class Subject Project of Ningxia Normal University under Grant NXYLXK2017B11..

References

[1] J. G. López Salas, C. Vázquez, *PDE formulation of some SABR/LIBOR market models and its numerical solution with a sparse grid combination technique*, Computers & Mathematics with Applications, **v**ol. 75, no. 5, pp. 1616-1634, 2018.

[2] C. Hendricks, C. Heuer, M. Ehrhardt, M. Günther, *High-order ADI finite difference schemes for parabolic equations in the combination technique with application in finance*, Journal of Computational and Applied Mathematics, **v**ol. 316, pp. 175-194, 2017.

[3] P. L. Quéré, *Accuracy solution to the square thermally driven cavity at high rayleigh number*, Computers & Fluids, **v**ol. 20, pp. 29-41, 1991.

[4] M. M. Gupta, R. P. Manohar, J. W. Stephenson, *A single cell high order scheme for the convection-diffusion equation with variable coefficients*, International Journal for Numerical Methods in Fluids, **v**ol. 4, no. 7, pp. 641-651, 1984.

[5] M. M. Gupta, R. P. Manohar, J. W. Stephenson, *High-order difference schemes for two-dimensional elliptic equations*, Numerical Methods for Partial Differential Equation, **v**ol. 1, no. 1, pp. 71-80, 1985.

[6] M. M. Gupta, *High accuracy solution of incompressible Navier-Stokes equations*, Journal of Computational Physics, **v**ol. 93, no. 2, pp. 343-359, 1991.

[7] S. K. Lele, *Compact finite difference schemes with spectral-like resolution*, Journal of Computational Physics, **v**ol. 103, no. 1, pp. 16-42, 1992.

[8] M. M. Gupta, J. Kouatchou, *Symbolic derivation of finite difference approximations for the three-dimensional Poisson equation*, Numerical Methods for Partial Differential Equation, **v**ol. 14 no. 5, pp. 593-606, 1998.

[9] J. Zhang, L. Ge, M. M. Gupta, *Fourth order compact difference scheme for 3D convection diffusion equation with boundary layers on non-uniform grids*, Neural. Parallel & Scientific Computations, **v**ol. 8, pp. 373-392, 2000.

[10] L. Ge, J. Zhang, *Accuracy, robustness, and efficiency comparison in iterative computation of convection diffusion equation with boundary layers* , Numerical Methods for Partial Differential Equation, **v**ol. 16, no. 4, pp. 379-394, 2000.

[11] J. Zhang, J. Kouatchou, L. Ge, *A family of fourth order difference scheme on rotated grid for two dimensional convection-diffusion equations*, Mathematics & Computers in Simulation, **v**ol. 59, no. 5, pp. 413-429, 2002.

[12] J. S. Karaa, *High-order difference schemes for 2D elliptic and parabolic problems with mixed derivatives*, Numerical Methods for Partial Differential Equation, **v**ol. 23, no. 2, pp. 366-378, 2007.

[13] U. Ananthakrishnaiah, R. Manohar, J. W. Stephenson, *High-order methods for elliptic equation with variable coefficients*, Numerical Methods for Partial Differential Equation, **v**ol. 3, no. 3, pp. 219-227, 1987.

[14] J. Y. Choo, D. H. Schultz, *A Stable high-order method for the heated cavity problem*, International Journal for Numerical Methods in Fluids, **v**ol. 15, no. 11, pp. 1313-106, 1992.

[15] J. Y. Choo, *Stable high order methods for elliptic equations with large first order terms* , Computers & Mathematics with Applications, **v**ol. 27, no. 1, pp. 65-80, 1994.

[16] J. Y. Choo, D. H. Schultz, *A high order difference method for the steady state Navier-Stokes equations*, Computers & Mathematics with Applications, **v**ol. 27, no. 11, pp. 105-119, 1994.

[17] W. F. Spotz, *High-Order Compact Finite Difference Schemes for Computational Mechanics*, PhD thesis, University of Texas at Austin, **A**ustin, TX (1995).

[18] W. F. Spotz, G. F. Carey, *High-order compact scheme for the steady stream-function vorticity equations*, International Journal for Numerical Methods in Engineering, **v**ol. 38, no. 20, pp. 3497-3512, 1995.

[19] W. F. Spotz, G. F. Carey, *A high-order compact formulation for the 3D Poisson equation*, Numerical Methods for Partial Differential Equation, **v**ol. 12, pp. 235-243, 1996.

[20] Z. F. Tian, Y. Cui, *Fourth-order compact finite-difference method for the steady three-dimsional convection-diffusion equation*, in *Proceedings of The 6th EPMESC Conference*, D. Han, H. Mo (ed.), **G**uangzhou, China, South China University of Technology Press, Guangzhou, pp. 286-291, 1997.

[21] J. Zhang, *An explicit fourth-order compact finite difference scheme for three dimensional convection-diffusion equation*, Communications in Numerical Methods in Engineering, **v**ol. 14, no. 3, pp. 263-280, 1998.

[22] L. X. Ge, J. Zhang, *Symbolic computation of high order compact difference schemes for three dimensional linear elliptic partial differential equations with variable coefficients*, Journal of Computational and Applied Mathematics, **v**ol. 143, no. 1, pp. 9-27, 2002.

[23] M. C. Lai, *A simple compact fourth-order Poisson solver on polar geometry*, Journal of Computational Physics, **v**ol. 182, no. 1, pp. 337-345, 1992.

[24] M. C. Lai, J. M. Tseng,, *A formally fourth-order accurate compact scheme for 3D poisson equation in cylindrical coordinates*, Journal of Computation and Applied Mathematics, **v**ol. 201, no. 1, pp. 175-181, 2007.

[25] I. Singer, E. Turkel, *Sixth-order accurate finite difference schemes for the Helmholtz equation*, Journal of Computation Acoustics, vol. 14, no. 3, pp. 339-351, 2006.

[26] G. Sutmann, *Compact finite difference schemes of sixth order for the Helmholtz equation*, Journal of Computation and Applied Mathematics, **v**ol. 203, no. 1, pp. 15-31, 2007.

[27] Y. P. Fu, *Compact fourth-order finite difference schemes for Helmholtz equation with high wave numbers*, Journal of Computational Mathematics, **v**ol. 26, no. 1, pp. 98-111, 2008.

[28] E. Turkel, D. Gordon, R. Gordon, S. Tsynkov, *Compact 2D and 3D sixth order schemes for the Helmholtz equation with variable wave number*, Journal of Computational Physics, **v**ol. 232, no. 1, pp. 272-287, 2013.

[29] E. Erturk, C. Gokcol, *Fourth-order compact formulation of Navier-Stokes equations and driven cavity flow at high Reynolds numbers*, International Journal for Numerical Methods in Fluids, **v**ol. 50, no. 4, pp. 421-436, 2006.

[30] S. Sen, *Fourth order compact schemes for variable coefficient parabolic problems with mixed derivatives*, Computers & Fluids, **v**ol. 135, pp. 81-89, 2016.

[31] T. Wang, T. G. Liu, *A consistent fourth-order compact scheme for solving convection-diffusion equation*, Mathematica Numerical Sinica, **v**ol. 8, no. 4, pp. 392-404, 2016. (in Chinese)

[32] H.-B. Li, M.-Y. Song, E.-J. Zhong, X.-M. Gu, *Numerical gradient schemes for heat equations based on the collocation polynomial and hermite interpolation*, Mathematics, **v**ol. 7, no. 1, 93, 2019. DOI: https://doi.org/10.3390/math701009310.3390/math7010093.

[33] A. Brandt, I. Yavneh, *Accelerated multigrid convergence and high-Reynolds recirculating flows*, Journal Scientific Computing, **v**ol. 14, no. 3, pp. 607-626, 1993.

[34] P. C. Chu, C. Fan, *A three-point combined compact difference scheme*, Journal of Computational Physics, **v**ol. 140, no. 2, pp. 370-399, 1998.

[35] E. Hairer, S. P. Norsett, G. Wanner, *Solving Ordinary Differential Equations*, Springer, **B**erlin/NewYork, 2nd ed 1993.

[36] J. C. Strikwerda, *High-order-accurate schemes for incompressible viscous flow*, International Journal for Numerical Methods in Fluids,, **v**ol. 24, no. 7, pp. 715-734, 1997.

[37] M. M. Gupta, J. Kouatchou, J. Zhang, *Comparison of second- and fourth-order discretization for multigrid Poisson solvers*, Journal of Computational Physics, **v**ol. 132, no. 2, pp. 226-232.

[38] M. M. Gupta, J. Kouatchou, J. Zhang, *A compact multigrid solver for convection-diffusion equations*, Journal of Computational Physics, **v**ol. 132, no. 1, pp. 123-129, 1997.

[39] J. Zhang, *Accelerated multigrid high accuracy solution of the convection-diffusion equation with high Reynolds number*, Numerical Methods for Partial Differential Equation, **v**ol. 13, no. 1, pp. 77-92, 1997.

[40] J. Zhang, *Fast and high accuracy multigrid solution of the three dimensional Poisson equation*, Journal of Computational Physics, **v**ol. 143, no. 2, pp. 449-461, 1998.

[41] M. M. Gupta, J. Zhang, *High accuracy multigrid solution of the 3D convection-diffusion equation*, Applied Mathematics and Computation, **v**ol. 113, no. 2, pp. 249-274, 2000.

[42] J. Zhang, L. Ge, J. Kouatchou, *A two colorable fourth-order compact difference scheme and parallel iterative solution of the 3D convection diffusion equation*, Mathematics and Computers in Simulation, **v**ol. 54, no. 1-3, pp. 65-80, 2000.

[43] J. Zhang, *Multigrid method and fourth-order compact scheme for 2D Poisson equation with unequal mesh-size discretization*, Journal of Computational Physics, **v**ol. 179, no. 1, pp. 170-179, 2002.

[44] Y. Ge, *Multigrid method and fourth-order compact difference discretization scheme with unequal meshsizes for 3D Poisson equation*, Journal of Computational Physics, **v**ol. 229, no. 18, pp. 6381-6391, 2010.

[45] Y. Ge, F. Cao, J. Zhang, *A transformation-free HOC scheme and multigrid method for solving the 3D Poisson equation on nonuniform grids*, Journal of Computational Physics, **v**ol. 234, pp. 199-216, 2013.

[46] F. Cao, Y. Ge, H. Sun, *Partial semi-coarsening multigrid method based on the HOC scheme on nonuniform grids for the convection-diffusion problems*, International Journal of Computer Mathematics, **v**ol. 94, no.12, pp. 2356-2372, 2017.

[47] A. Brandt, *Multi-level adaptive solutions to boundary-value problems*, Mathematics of Computation, **v**ol. 31, no. 138, pp. 333-390, 1977.

[48] P. Wesseling, *An Introduction to Multigrid Methods*, Pure & Appl. Math., John Wiles & Sons, Chichester, UK (1992).

Appendix

A Details of the central difference operators

$$\delta_x^2 u_0 = \frac{u_1 - 2u_0 + u_3}{h^2}$$
$$\delta_x u_0 = \frac{u_1 - u_3}{2h}$$
$$\delta_y^2 u_0 = \frac{u_2 - 2u_0 + u_4}{h^2}$$
$$\delta_y u_0 = \frac{u_2 - u_4}{2h}$$
$$\delta_x \delta_y u_0 = \frac{u_5 - u_6 + u_7 - u_8}{4h^2}$$
$$\delta_x^2 \delta_y u_0 = \frac{u_5 + u_6 - u_7 - u_8 - 2u_2 + 2u_4}{2h^3}$$
$$\delta_y^2 \delta_x u_0 = \frac{u_5 - u_6 - u_7 + u_8 - 2u_1 + 2u_3}{2h^3}$$