RESEARCH

Boundary Value Problems a SpringerOpen Journal

Open Access



A reduced fourth-order compact difference scheme based on a proper orthogonal decomposition technique for parabolic equations

Baozou Xu¹ and Xiaohua Zhang^{1,2*}

¹Correspondence: zhangxiaohua07@163.com ¹College of Science, China Three Gorges University, Yichang, China ²Three Gorges Mathematical Research Center, China Three Gorges University, Yichang, China

Abstract

In this paper, we present a reduced high-order compact finite difference scheme for numerical solution of the parabolic equations. CFDS4 is applied to attain high accuracy for numerical solution of parabolic equations, but its computational efficiency still needs to be improved. Our approach combines CFDS4 with proper orthogonal decomposition (POD) technique to improve the computational efficiency of the CFDS4. The validation of the proposed method is demonstrated by four test problems. The numerical solutions are compared with the exact solutions and the solutions obtained by the CFDS4. Compared with CFDS4, it is shown that our method has greatly improved the computational efficiency without a significant loss in accuracy for solving parabolic equations.

Keywords: Proper orthogonal decomposition; Parabolic equations; Compact finite difference scheme

1 Introduction

Many problems in physical phenomena, engineering equipment, and living organisms, such as the proliferation of gas, the infiltration of liquid, the conduction of heat, and the spread of impurities in semiconductor materials, can be described with parabolic equations [1-3]. In addition, the system related to parabolic equation can be used to describe the phase separation in material sciences [4] and a class of abstract control systems concerned with parabolic equation has been developed [5]. Especially, there exists analysis about linear and nonlinear boundary value problems [6]. Thus, it is of great value to study their exact solutions. In practical engineering calculations, their exact solutions are not easy to obtain directly. We have to find their numerical solutions by various numerical methods. Therefore, the study of numerical solutions of parabolic equations is of great theoretical and practical value. Until now, several types of numerical methods have been developed for numerical simulation of the parabolic problems. For example, the authors in [7] used two-level finite difference schemes to solve one-dimensional parabolic equation [8]. In [9], a new finite volume method for parabolic equation is presented. Zhang et al. provided



© The Author(s) 2019. This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

the Crank–Nicolson-type difference scheme to solve the equations with spatial variable coefficient [10].

In the methods mentioned above, the finite difference scheme is always regarded as one of the effective methods to solve partial differential equations due to its simple form and easiness to be understood and programmed. In the last decade, the high-order compact finite difference scheme (CFDS) has been implemented for numerical solution of various types of partial differential equations. Such as the Rosenau-regularized long wave (RLW) equation [11], integro-differential equations [12], Burgers' equation [13], Helmholtz equation [14], Navier–Stokes equations [15], Schrödinger equations [16], Poisson equation [17], sine-Gordon equation [18]. Although the high-order CFDS can get high accurate solution, it usually either needs small spatial discretization or extended finite difference stencils and a small time step, which results in computationally expensive calculations. In general, the computational accuracy and computational efficiency are often seen as the two important factors to evaluate a numerical algorithm. Though the CFDS can achieve a satisfying numerical solution and possesses fast rate of convergence, it also brings many difficulties to practical application such as heavily computational load. Therefore, it is necessary for us to develop the high-order CFDS that not only simplifies the computational load and saves computational time of calculation in the actual engineering problems, but also holds sufficiently accurate solution. In recent years, some reduced models based on proper orthogonal decomposition (POD) have attracted more and more attention in the field of computational mechanics [19, 20]. POD, also known as Karhunen-Loève decomposition (KLD), principal component analysis (PCA), or singular value decomposition (SVD), provides a powerful technique to reduce a large number of interdependent variables to a much smaller number of uncorrelated variables while retaining as much as possible of the variation in the original variables [21-24]. Thus, we can use the POD technique to greatly reduce the computational cost.

In the past decades, POD has been widely applied to numerical solutions to construct some reduced models. For example, Luo and Sun combined POD with finite difference method, finite element method, and finite volume method to solve the parabolic equations [25–28], Navier–Stokes equations [29, 30], Sobolev equations [31], viscoelastic wave equation [32], hyperbolic equations [33], and Burgers' equation [34]. Dehghan and Abbaszadeh proposed POD with upwind local radial basis functions-differential quadrature method to solve compressible Euler equation [35], applied POD into Galerkin method to simulate two-dimensional solute transport problems [36], and combined POD with empirical interpolation method [37]. Zhang proposed a fast and efficient meshless method based on POD for transient heat conduction problem [38] and convection-diffusion problems [19]. Especially, the authors in [39] gave the Carleman estimates for singular parabolic equation. However, to our best knowledge, there are no published results when POD is used to reduce the CFDS4 for parabolic equations. The main goal of this paper is to construct a numerical algorithm which has high computational accuracy and efficiency for solving parabolic equations. Thus, the focus of this paper is on combining the CFDS4 and the POD method, namely the R-CFDS4, to solve parabolic equations.

The paper is organized as follows. The CFDS4 for 1D parabolic equations is derived in Sect. 2. In Sect. 3, R-CFDS4 based on POD for 1D parabolic equations is developed. In Sect. 4, the error estimates between the exact solutions are given. In Sect. 5, a reduced alternating direction implicit fourth-order compact finite difference scheme (R- ADI-CFDS4) based on POD for 2D parabolic equations is developed. In Sect. 6, some numerical examples are presented to demonstrate the efficiency and reliability of the algorithm. Section 7 provides main conclusions.

2 The fourth-order compact FDS for 1D parabolic equation

Firstly, we consider the following 1D initial-boundary value problem P1:

$$\begin{cases} \frac{\partial u}{\partial t} - a \frac{\partial^2 u}{\partial x^2} = f(x, t), & 0 < x < L, \\ u(0, t) = u(L, t) = 0, & 0 \le t \le T, \\ u(x, 0) = \varphi(x), & 0 \le x \le L, \end{cases}$$
(1)

where *a* is a positive constant, f(x, t) and $\varphi(x)$ are two given enough smooth functions. Let *h* be the spatial step increment on *x*-direction and τ be the time step increment, and then write $x_j = jh$ (j = 0, 1, 2, ..., J), $t_n = n\tau$ (n = 0, 1, 2, ..., N), $u_j^n \approx u(x_j, j_n)$.

Next, we derive the CFDS4 for problem P₁, noting the following Taylor formulas:

$$u(x_{j-1},t) = u(x_{j},t) - h \frac{\partial u}{\partial x} \Big|_{(x_{j},t)} + \frac{h^{2}}{2} \frac{\partial^{2} u}{\partial x^{2}} \Big|_{(x_{j},t)} - \frac{h^{3}}{6} \frac{\partial^{3} u}{\partial x^{3}} \Big|_{(x_{j},t)} + \frac{h^{4}}{24} \frac{\partial^{4} u}{\partial x^{4}} \Big|_{(x_{j},t)} - \cdots,$$

$$u(x_{j+1},t) = u(x_{j},t) + h \frac{\partial u}{\partial x} \Big|_{(x_{j},t)} + \frac{h^{2}}{2} \frac{\partial^{2} u}{\partial x^{2}} \Big|_{(x_{j},t)} + \frac{h^{3}}{6} \frac{\partial^{3} u}{\partial x^{3}} \Big|_{(x_{j},t)} + \frac{h^{4}}{24} \frac{\partial^{4} u}{\partial x^{4}} \Big|_{(x_{j},t)} + \cdots.$$
(2)
$$(2)$$

Adding Eq. (2) to Eq. (3), we get

$$\frac{u(x_{j-1},t) - 2u(x_j,t) + u(x_{j+1},t)}{h^2} = \frac{\partial^2 u}{\partial x^2}|_{(x_j,t)} + \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}\Big|_{(x_j,t)} + O(h^4).$$
(4)

Let $v(x, t) = \frac{\partial^2 u}{\partial x^2}$, then Eq. (4) can be rewritten as follows:

$$\frac{u(x_{j-1},t) - 2u(x_{j},t) + u(x_{j+1},t)}{h^{2}} = v(x_{j},t) + \frac{h^{2}}{12} \frac{\partial^{2} v}{\partial x^{2}} \Big|_{(x_{j},t)} + O(h^{4}) = v(x_{j},t) + \frac{h^{2}}{12} \frac{v(x_{j-1},t) - 2v(x_{j},t) + v(x_{j+1},t)}{h^{2}} + O(h^{4}) = \frac{1}{12} v(x_{j-1},t) + \frac{10}{12} v(x_{j},t) + \frac{1}{12} v(x_{j+1},t) + O(h^{4}).$$
(5)

In Eq. (5), if we choose $t = t_{n+\frac{1}{2}}$, then we obtain

$$\frac{u(x_{i-1}, t_{n+\frac{1}{2}}) - 2u(x_i, t_{n+\frac{1}{2}}) + u(x_{i+1}, t_{n+\frac{1}{2}})}{h^2} = \frac{1}{12} \left(\nu(x_{i-1}, t_{n+\frac{1}{2}}) + 10\nu(x_i, t_{n+\frac{1}{2}}) + \nu(x_{i+1}, t_{n+\frac{1}{2}}) \right) + O(h^4).$$
(6)

Noting that Eq. (1) and $v(x, t) = \frac{\partial^2 u}{\partial x^2}$, we get

$$\nu(x,t) = \frac{1}{a} \left(\frac{\partial u}{\partial t} - f \right). \tag{7}$$

For the first-order derivative of time, using the center difference quotient formula, we get

$$\frac{\partial u}{\partial t}\Big|_{(x_j,t_{n+\frac{1}{2}})} = \frac{u(x_j,t_{n+1}) - u(x_j,t_n)}{\tau} + O(\tau^2).$$
(8)

Thus, substituting Eq. (8) to Eq. (7), we obtain

$$\nu(x_{j}, t_{n+\frac{1}{2}}) = \frac{1}{a} \left(\frac{\partial u}{\partial t} \Big|_{(x_{j}, t_{n+\frac{1}{2}})} - f(x_{j}, t_{n+\frac{1}{2}}) \right)$$
$$= \frac{1}{a} \left(\frac{u(x_{j}, t_{n+1}) - u(x_{j}, t_{n})}{\tau} - f(x_{j}, t_{n+\frac{1}{2}}) \right) + O(\tau^{2}).$$
(9)

Similar with Eq. (2) and Eq. (3), we have the following formulas:

$$u(x, t_{n+1}) = u(x, t_{n+0.5}) + \frac{\tau}{2} \frac{\partial u}{\partial t} \Big|_{(x, t_{n+0.5})} + O(\tau^2)$$
(10)

and

$$u(x,t_n) = u(x,t_{n+0.5}) - \frac{\tau}{2} \frac{\partial u}{\partial t} \bigg|_{(x,t_{n+0.5})} + O(\tau^2).$$
(11)

By adding Eq. (10) to Eq. (11), we can get the following formula:

$$u(x, t_{n+0.5}) = \frac{u(x, t_n) + u(x, t_{n+1})}{2} + O(\tau^2).$$
(12)

Substituting Eq. (9) and Eq. (12) into Eq. (6), we get the formula

$$\frac{u(x_{j-1}, t_n) + u(x_{j-1}, t_{n+1}) - 2u(x_j, t_n) - 2u(x_j, t_{n+1}) + u(x_{j+1}, t_n) + u(x_{j+1}, t_{n+1})}{2h^2} = \frac{1}{12a} \left(\frac{u(x_{j-1}, t_{n+1}) - u(x_{j-1}, t_n)}{\tau} - f(x_{j-1}, t_{n+\frac{1}{2}}) + 10 \frac{u(x_j, t_{n+1}) - u(x_j, t_n)}{\tau} - 10f(x_j, t_{n+\frac{1}{2}}) + \frac{u(x_{j+1}, t_{n+1}) - u(x_{j+1}, t_n)}{\tau} - f(x_{j+1}, t_{n+\frac{1}{2}}) \right) + O(\tau^2 + h^4).$$
(13)

Let $r = \frac{a\tau}{h^2}$, and using numerical solution u_j^n to replace exact solution $u(x_j, t_n)$ and ignoring higher-order items, we get

$$\frac{r}{2} \left(u_{j-1}^{n} + u_{j-1}^{n+1} - 2u_{j}^{n} - 2u_{j}^{n+1} + u_{j+1}^{n} + u_{j+1}^{n+1} \right)$$

$$= \frac{1}{12} \left(u_{j-1}^{n+1} - u_{j-1}^{n} + 10u_{j}^{n+1} - 10u_{j}^{n} + u_{j+1}^{n+1} - u_{j+1}^{n} \right)$$

$$- \frac{\tau}{12} \left(f(x_{j-1}, t_{n+\frac{1}{2}}) + 10f(x_{j}, t_{n+\frac{1}{2}}) + f(x_{j+1}, t_{n+\frac{1}{2}}) \right).$$
(14)

Finally, considering the initial and boundary conditions Eq. (1), we obtain the compact difference formula for problem P_1 :

$$\begin{pmatrix} \frac{1}{12} - \frac{r}{2} \end{pmatrix} u_{j-1}^{n+1} + \begin{pmatrix} \frac{5}{6} + r \end{pmatrix} u_{j}^{n+1} + \begin{pmatrix} \frac{1}{12} - \frac{r}{2} \end{pmatrix} u_{j+1}^{n+1} = \begin{pmatrix} \frac{1}{12} + \frac{r}{2} \end{pmatrix} u_{j-1}^{n} + \begin{pmatrix} \frac{5}{6} - r \end{pmatrix} u_{j}^{n} + \begin{pmatrix} \frac{1}{12} + \frac{r}{2} \end{pmatrix} u_{j+1}^{n} + \frac{\tau}{12} (f(x_{j-1}, t_{n+\frac{1}{2}}) + 10f(x_{j}, t_{n+\frac{1}{2}}) + f(x_{j+1}, t_{n+\frac{1}{2}})), 1 \le j \le J - 1, \qquad 0 \le n \le N - 1, u_{j}^{0} = \varphi(x_{j}), \qquad 0 \le j \le J, \qquad u_{0}^{n} = 0 = u_{J}^{n} = 0, \qquad 1 \le n \le N, J, u_{0}^{n} = 0 = u_{J}^{n} = 0, \qquad 1 \le n \le N.$$
 (15)

The stability of Eq. (15) is unconditionally stable [40], and the error of the above compact difference scheme is as follows:

$$|u(x_j, t_n) - u_j^n| = O(\tau^2, h^4).$$
(16)

3 A reduced fourth-order compact FDS for 1D parabolic equation

In the scheme mentioned above, we only use three points to achieve high accuracy. However, the CFDS4 can get high accuracy by the scheme in Eq. (15), which usually needs extended finite difference stencils or small temporal step. Nevertheless, it will bring overlong computational time in the large real-life engineering problems. Hence, a key issue for engineering problems is how to build a reduced model with the high efficiency and hold sufficiently high accuracy compared with the scheme in Eq. (15).

The POD is an effective approach which can not only reduce the computational time significantly, but also guarantee the high accuracy of algorithm. The main idea of POD is to generate a group of optimal bases of the assemble of snapshots. Then, the reduced CFDS4 is constructed by the optimal basis. The snapshots, in this paper, are collected from some numerical solutions of CFDS4. As a matter of fact, when we compute the actual problem, since the development and change of a large number of future nature phenomena are closely related to previous results, one may choose the numerical experiments or observation data as the snapshots. Therefore, in this section, we first obtain the optimal basis from the snapshot and then use the optimal basis to derive a R-CFDS4 for parabolic equations.

The set of snapshots $\{u_j^{n_i}\}_{i=1}^d$ $(j = 1, 2, ..., J-1, 1 \le n_1 \le n_2 < \cdots < n_d \le N)$ can be written as an $(J-1) \times d$ matrix **A** as follows:

$$\boldsymbol{A}_{(J-1)\times d} = \begin{pmatrix} u_1^{n_1} & u_1^{n_2} & \cdots & u_1^{n_d} \\ u_2^{n_1} & u_2^{n_2} & \cdots & u_2^{n_d} \\ \vdots & \vdots & \vdots & \vdots \\ u_{l-1}^{n_1} & u_{l-1}^{n_2} & \cdots & u_{l-1}^{n_d} \end{pmatrix}.$$
(17)

Applying the singular value decomposition (SVD) on matrix **A**, we obtain

$$\boldsymbol{A}_{(J-1)\times d} = \boldsymbol{U} \begin{pmatrix} \boldsymbol{D}_r & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} \boldsymbol{V}^{\mathrm{T}},$$
(18)

where $\boldsymbol{U} = \boldsymbol{U}_{(J-1)\times(J-1)}$ and $\boldsymbol{V} = \boldsymbol{V}_{d\times d}$ are all orthogonal matrices, $\boldsymbol{D}_r = \operatorname{diag}(\lambda_1, \lambda_2, \dots, \lambda_r)$. The matrix $\boldsymbol{U} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_{J-1})$ contains the orthogonal eigenvectors to $\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}$. The singular values λ_i $(i = 1, 2, \dots, r)$ satisfy the relation of $\lambda_1 \ge \lambda_2 \ge \dots \ge \lambda_r > 0$. Denote *d* columns of $\boldsymbol{A}_{(J-1)\times d}$ by $\boldsymbol{\alpha}^i = (\boldsymbol{u}_1^{n_i}, \boldsymbol{u}_2^{n_i}, \dots, \boldsymbol{u}_{J-1}^{n_i})^{\mathrm{T}}$ $(i = 1, 2, \dots, d)$, and define a projection P_M by

$$P_M(\boldsymbol{\alpha}^i) = \sum_{j=1}^M (\boldsymbol{\theta}_j, \boldsymbol{\alpha}^i) \boldsymbol{\theta}_j, \tag{19}$$

where $0 < M \le d$ and (\cdot, \cdot) is the inner product of vectors. Then there exists the following result[41, 42]:

$$\left\|\boldsymbol{\alpha}^{i}-P_{M}\left(\boldsymbol{\alpha}^{i}\right)\right\|_{2}\leq\lambda_{M+1},$$
(20)

where $\|\cdot\|_2$ is the standard normal of vector. Thus $\{\theta_l\}_{l=1}^M$ is a group of optimal bases and $\theta = (\theta_1, \theta_2, \dots, \theta_M)$ is a matrix constructed by the orthogonal eigenvectors such that $\theta^T \theta = I(I \text{ is a unit matrix of order } M)$

In the following, we construct a reduced CFDS4 for problem P_1 . Equation (15) can be rewritten as follows:

$$\begin{aligned} u_{j}^{n+1} &- u_{j}^{n} \\ &= \left(\frac{3}{5}r - \frac{1}{10}\right)u_{j-1}^{n+1} - \frac{6}{5}ru_{j}^{n+1} + \left(\frac{3}{5}r - \frac{1}{10}\right)u_{j+1}^{n+1} + \left(\frac{3}{5}r + \frac{1}{10}\right)u_{j-1}^{n} \\ &- \frac{6}{5}ru_{j}^{n} + \left(\frac{3}{5}r + \frac{1}{10}\right)u_{j+1}^{n} + \frac{\tau}{10}\left(f(x_{j-1}, t_{n+\frac{1}{2}}) + 10f(x_{j}, t_{n+\frac{1}{2}}) + f(x_{j+1}, t_{n+\frac{1}{2}})\right). \end{aligned}$$
(21)

Let $\mathbf{u}^n = (u_1^n, u_2^n, \dots, u_{J-1}^n)^T$, then we can rewrite Eq. (21) as the following matrix form:

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \boldsymbol{K}_1 \boldsymbol{u}^{n+1} + \boldsymbol{K}_2 \boldsymbol{u}^n + \frac{\tau}{10} \boldsymbol{F}^n, \quad n = 0, 1, \dots, N-1,$$
(22)

where

$$\begin{split} \mathbf{K_1} &= \begin{pmatrix} -\frac{6}{5}r & \frac{3}{5}r - \frac{1}{10} & 0 \\ \frac{3}{5}r - \frac{1}{10} & -\frac{6}{5}r & \frac{3}{5}r - \frac{1}{10} & \\ & \ddots & \ddots & \ddots & \\ & & \frac{3}{5}r - \frac{1}{10} & -\frac{6}{5}r & \frac{3}{5}r - \frac{1}{10} \\ & 0 & & \frac{3}{5}r - \frac{1}{10} & -\frac{6}{5}r \end{pmatrix}, \\ \mathbf{K_2} &= \begin{pmatrix} -\frac{6}{5}r & \frac{3}{5}r + \frac{1}{10} & 0 & \\ \frac{3}{5}r + \frac{1}{10} & -\frac{6}{5}r & \frac{3}{5}r + \frac{1}{10} & \\ & \ddots & \ddots & \ddots & \\ & & \frac{3}{5}r + \frac{1}{10} & -\frac{6}{5}r & \frac{3}{5}r + \frac{1}{10} \\ & 0 & & \frac{3}{5}r + \frac{1}{10} & -\frac{6}{5}r \end{pmatrix} \end{split}$$

and

$$\boldsymbol{F}^{n} = \begin{pmatrix} f_{1}^{n} + (\frac{3}{5}r + \frac{1}{10})u_{0}^{n} - (\frac{3}{5}r - \frac{1}{10})u_{J}^{n+1} \\ f_{2}^{n} \\ \vdots \\ f_{J-2}^{n} \\ f_{J-1}^{n} + (\frac{3}{5}r + \frac{1}{10})u_{0}^{n} - (\frac{3}{5}r - \frac{1}{10})u_{J}^{n+1} \end{pmatrix},$$

 $\begin{aligned} f_j^n &= f(x_{j-1}, t_{n+\frac{1}{2}}) + 10 f(x_j, t_{n+\frac{1}{2}}) + f(x_{j+1}, t_{n+\frac{1}{2}}) \ (j = 1, 2, \dots, J-1). \\ \text{Substituting} \end{aligned}$

$$\boldsymbol{u}^{*n} = \boldsymbol{\theta} \boldsymbol{\beta}^n = \boldsymbol{\theta}_{(J-1) \times M} (\boldsymbol{\beta}^n)_{M \times 1}, \quad n = 0, 1, \dots, N,$$
(23)

to \boldsymbol{u}^n of Eq. (22) and noting that $\boldsymbol{\theta}^T \boldsymbol{\theta} = I$, we obtain the R-CFDS4 as follows:

$$\boldsymbol{\beta}^{n+1} = \boldsymbol{\beta}^{n} + \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{K}_{1} \boldsymbol{\beta}^{n+1} + \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{K}_{2} \boldsymbol{\beta}^{n} + \frac{\tau}{10} \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{F}^{n}, n = 0, 1, \dots, N-1,$$
(24)

where

$$\boldsymbol{\beta}^{\mathbf{0}} = \boldsymbol{\theta}^{\mathsf{T}} \boldsymbol{u}^{\mathbf{0}} = \boldsymbol{\theta}^{\mathsf{T}} \left(\boldsymbol{u}_{1}^{0}, \boldsymbol{u}_{2}^{0}, \dots, \boldsymbol{u}_{J-1}^{0} \right)^{T}.$$
(25)

After β^n is obtained from Eq. (24), one gets the POD optimal solution $u^{*n} = \theta \beta^n$. From the above derivation process, it can be seen that CFDS needs to solve the $(J - 1) \times (J - 1)$ equations in each time step, while R-CFDS4 Eq. (24) only needs $M \times M$ equations, usually $J \gg M$, which means R-CFDS4 requires much less computational time than CFDS4 within each time step. Thus, R-CFDS4 can save a lot of computational time during the whole process of solving parabolic equations.

4 The error analysis of a reduced fourth-order FDS for 1D parabolic problem

Theorem 1 Let \mathbf{u}^n be the solution vector of Eq. (15) and \mathbf{u}^{*n} be the solution vectors of Eq. (24). If $\{u_i^l\}_{l=1}^d$ are uniformly chosen from $\{u_i^n\}_{n=1}^N$ and $\|\mathbf{K}_1\|_2 \leq \frac{1}{2}$, $\|\mathbf{K}_2\|_2 \leq \frac{1}{2}$, then

$$\| \boldsymbol{u}^{*n} - \boldsymbol{u}^{n} \|_{2} \le \lambda_{M+1} \exp(N/d), \quad n = 1, 2, \dots, N,$$
 (26)

where $\|\mathbf{K}_1\|_2$ and $\|\mathbf{K}_2\|_2$ are the normal of matrix \mathbf{K}_1 and \mathbf{K}_2 , respectively.

Proof There are two cases in Eq. (26). The first case is $n = n_l$ (l = 1, 2, ..., d). Using the definition (Eq. (19))

$$P_M(\boldsymbol{u}^{n_l}) = \sum_{j=1}^M \left(\boldsymbol{\theta}_j, \boldsymbol{u}^{n_l}\right) \boldsymbol{\theta}_j$$
(27)

and Eq. (23) and Eq. (24), we have that $\boldsymbol{u}^{*n_l} = P_M(\boldsymbol{u}^{n_l})$. Applying Eq. (20), we have the following conclusion:

$$\left\| \boldsymbol{u}^{\boldsymbol{n}_{l}} - \boldsymbol{u}^{*\boldsymbol{n}_{l}} \right\|_{2} = \left\| \boldsymbol{u}^{\boldsymbol{n}_{l}} - P_{M} \left(\boldsymbol{u}^{\boldsymbol{n}_{l}} \right) \right\|_{2} \le \lambda_{M+1} \quad (l = 1, 2, \dots, d).$$
(28)

The second case is $n \neq n_l$. We let $t_n \in (t_{n_l}, t_{n_l+1})$, comparing Eq. (22) with Eq. (24), Eq. (24) can be written as similar forms of Eq. (22) as follows:

$$\boldsymbol{u}^{*n+1} = \boldsymbol{u}^{*n} + \boldsymbol{K}_1 \boldsymbol{u}^{*n+1} + \boldsymbol{K}_2 \boldsymbol{u}^{*n} + \frac{\tau}{10} \boldsymbol{F}^n, \quad n = 0, 1, \dots, N-1.$$
(29)

Subtracting Eq. (29) from Eq. (22), we obtain that

$$\|\boldsymbol{u}^{n+1} - \boldsymbol{u}^{*n+1}\|_{2} \leq \|\boldsymbol{u}^{n} - \boldsymbol{u}^{*n}\|_{2} + \|\boldsymbol{K}_{1}\|_{2} \|\boldsymbol{u}^{n+1} - \boldsymbol{u}^{*n+1}\|_{2} + \|\boldsymbol{K}_{2}\|_{2} \|\boldsymbol{u}^{n} - \boldsymbol{u}^{*n}\|_{2}.$$
 (30)

We let $\frac{c}{2} = \max\{\|\mathbf{K}_1\|_2, \|\mathbf{K}_2\|_2\}$, then

$$\|\boldsymbol{u}^{n+1} - \boldsymbol{u}^{*n+1}\|_{2} \leq \|\boldsymbol{u}^{n} - \boldsymbol{u}^{*n}\|_{2} + \frac{c}{2} \left[\|\boldsymbol{u}^{n+1} - \boldsymbol{u}^{*n+1}\|_{2} + \|\boldsymbol{u}^{n} - \boldsymbol{u}^{*n}\|_{2} \right].$$
(31)

Summing Eq. (31) from n_l , $n_l + 1, ..., n - 1$, we get

$$\|\boldsymbol{u}^{n+1} - \boldsymbol{u}^{*n+1}\|_{2} \leq \|\boldsymbol{u}^{n_{l}} - \boldsymbol{u}^{*n_{l}}\|_{2} + c \sum_{j=n_{l}}^{n-1} \|\boldsymbol{u}^{j} - \boldsymbol{u}^{*j}\|_{2}.$$
(32)

By the discrete Gronwall lemma, we get

$$\|\boldsymbol{u}^{n+1} - \boldsymbol{u}^{*n+1}\|_{2} \le \|\boldsymbol{u}^{n_{l}} - \boldsymbol{u}^{*n_{l}}\|_{2} \exp(c(n-n_{l})).$$
(33)

If $\|\mathbf{K}_1\|_2 \leq \frac{1}{2}$ and $\|\mathbf{K}_2\|_2 \leq \frac{1}{2}$, $\{u_i^{n_l}\}_{l=1}^d$ are uniformly chosen from $\{u_i^n\}_{n=1}^N$, which implies $n - n_l \leq N/d$, then we obtain from the above inequality that

$$\left\|\boldsymbol{u}^{n+1}-\boldsymbol{u}^{*n+1}\right\|_{2}\leq\left\|\boldsymbol{u}^{n_{l}}-\boldsymbol{u}^{*n_{l}}\right\|_{2}\exp(N/d)\leq\lambda_{M+1}\exp(N/d).$$

In the synthesis, the above two cases obtain Eq. (26), which completes Theorem 1. \Box

Theorem 2 Under the assumption of Theorem 1, let u_j^{*n} be the solution of the R-CFDS4 and $u(x_j, t_n)$ be the exact solution of problem (P₁). Then there are the following error estimates:

$$\left|u_{j}^{*n}-u(x_{j},t_{n})\right|=O(\lambda_{M+1}\exp(N/d),\tau^{2},h^{4}), \quad n=1,2,\ldots,N.$$
(34)

Proof Let u_i^n be the solution of the CFDS4, then

$$|u_{j}^{*n} - u(x_{j}, t_{n})| = |(u_{j}^{*n} - u_{j}^{n}) + u_{j}^{n} - u(x_{j}, t_{n})| \le |u_{j}^{*n} - u_{j}^{n}| + |u_{j}^{n} - u(x_{j}, t_{n})|,$$

$$j = 1, 2, \dots, J - 1,$$
(35)

since

$$|u_{j}^{n} - u(x_{j}, t_{n})| = O(\tau^{2}, h^{4}).$$
(36)

Meanwhile, according to Theorem 1 and vector norm, we get

$$\left|u_{j}^{n}-u(x_{j},t_{n})\right|\leq\left\|\boldsymbol{u}^{*n}-\boldsymbol{u}^{n}\right\|_{2}\leq\lambda_{M+1}\exp(N/d).$$
(37)

Thus, we can obtain Eq. (34).

5 A reduced fourth-order compact FDS for 2D parabolic problem and its error analysis

In this section, we firstly construct alternating direction implicit fourth-order compact finite difference scheme (ADI-CFDS4) for the 2D parabolic equations, then develop R-ADI-CFDS4 for the 2D parabolic equations and provide error estimates between solutions of the R-ADI-CFDS4 and exact solutions, which is similar to 1D parabolic problems.

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain. Consider the following initial-boundary value problem (P₂):

$$\begin{cases} \frac{\partial u(x,y,t)}{\partial t} - \left(\frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2}\right) = f(x,y,t), & (x,y,t) \in \Omega \times (0,T), \\ u(x_i,y_j,t_0) = \varphi(x_i,y_j), & (x,y) \in \Omega, \\ u(x_0,y_j,t_k) = g(y_j,t_k), & (x,y,t) \in \partial\Omega \times (0,T) \end{cases}$$
(38)

Partition the domain $\Omega \times (0, T)$ with $x_i = ih_x$ (i = 0, 1, 2, ..., L), $y_j = jh_y$ (j = 0, 1, 2, ..., J), and $t_n = n\tau$ (n = 0, 1, 2, ..., N), where τ is temporal step increment, h_x and h_y are the spatial step increments on *x*-direction and *y*-direction, respectively.

Let $u_{i,j}^n \approx u(x_i, y_j, t_n)$; $r_1 = \frac{\tau}{h_r^2}$, $r_2 = \frac{\tau}{h_r^2}$

$$\begin{cases} \varepsilon_x^2 u_{i,j}^n = \frac{1}{12} (u_{i-1,j}^n + 10u_{i,j}^n + u_{i+1,j}^n), \\ \varepsilon_y^2 u_{i,j}^n = \frac{1}{12} (u_{i,j-1}^n + 10u_{i,j}^n + u_{i,j+1}^n) \end{cases}$$
(39)

and

$$\begin{cases} \delta_x^2 u_{i,j}^n = u_{i-1,j}^n - 2u_{i,j}^n + u_{i+1,j}^n, \\ \delta_y^2 u_{i,j}^n = u_{i,j-1}^n - 2u_{i,j}^n + u_{i,j+1}^n. \end{cases}$$
(40)

Then the ADI-CFDS4 of problem can be described as follows:

$$\begin{cases} (\varepsilon_x^2 - \frac{r_1}{2}\delta_x^2)\bar{u}_{i,j}^{n+1} = (r_1\varepsilon_y^2\delta_x^2 + r_2\varepsilon_x^2\delta_y^2)u_{i,j}^n + \tau\varepsilon_x^2\varepsilon_y^2\int_{i,j}^{n+\frac{1}{2}}, \\ (\varepsilon_y^2 - \frac{r_2}{2}\delta_y^2)u_{i,j}^{n+1} = \bar{u}_{i,j}^{n+1} + (\varepsilon_y^2 - \frac{r_2}{2}\delta_y^2)u_{i,j}^n. \end{cases}$$
(41)

In Eq. (41), $1 \le i \le L - 1$, $1 \le j \le J - 1$, $0 \le n \le N - 1$.

The difference scheme is unconditionally stable [40], and the error is as follows:

$$\left|u_{i,j}^{n} - u(x_{i}, y_{j}, t_{n})\right| = O(\tau^{2}, h_{x}^{4}, h_{y}^{4}).$$
(42)

Let $u_l^n = u_{i,j}^n$, $f_l^n = f_{i,j}^n$ $(l = (i - 1) \cdot L + j, m = L \cdot J, 1 \le l \le m, 1 \le j \le J, 1 \le i \le L, 1 \le n \le N)$. Using the similar methods as in Sect. 3, we obtain the POD base $\theta = \theta_{m \times M}$ such that $\theta^T \theta = I_{M \times M}$. Then the corresponding vector formulation of Eq. (41) can be written as follows:

$$\begin{cases} (\mathbf{K}_{1} - \frac{r_{1}}{2}\mathbf{K}_{2})\bar{\mathbf{u}}^{n} = (r_{1}\mathbf{K}_{2}\mathbf{K}_{1} + r_{2}\mathbf{K}_{2}\mathbf{K}_{1})\mathbf{u}^{n} + \tau\mathbf{K}_{1}\mathbf{K}_{1}\mathbf{F}^{n}, \\ (\mathbf{K}_{2} - \frac{r_{2}}{2}\mathbf{K}_{1})\mathbf{u}^{n+1} = \bar{\mathbf{u}}^{n} + (\mathbf{K}_{1} - \frac{r_{2}}{2}\mathbf{K}_{2})\mathbf{u}^{n}, \end{cases}$$
(43)

where $\boldsymbol{u}^{\boldsymbol{n}} = (u_1^n, u_2^n, u_3^n, \dots, u_m^n)^T$, $\boldsymbol{F}^{\boldsymbol{n}} = (f_1^n, f_2^n, f_3^n, \dots, f_m^n)^T$, and $\bar{\boldsymbol{u}}^{\boldsymbol{n}} = (\bar{u}_1^n, \bar{u}_2^n, \bar{u}_3^n, \dots, \bar{u}_m^n)^T$. The form of \boldsymbol{K}_1 and \boldsymbol{K}_2 is the same as Eq. (21) except its order number. If $\boldsymbol{u}^{\boldsymbol{n}}$ and $\bar{\boldsymbol{u}}^{\boldsymbol{n}}$ of

Eq. (38) are substituted for $\boldsymbol{u^{*n}} = \boldsymbol{\theta}_{m \times M}(\boldsymbol{\alpha^n})_{M \times 1}$, $\boldsymbol{\bar{u}^n} = \boldsymbol{\theta}_{m \times M}(\boldsymbol{\bar{\alpha}^n})_{M \times 1}$, $\boldsymbol{F^n} = \boldsymbol{\theta}_{m \times M}(\boldsymbol{\bar{F}^n})_{M \times 1}$ (*n* = 0, 1, 2, ..., *N*), respectively, then we obtain the R-ADI-CFDS4 as follows:

$$\begin{cases} \theta^{\mathsf{T}}(\mathsf{K}_{1} - \frac{r_{1}}{2}\mathsf{K}_{2})\theta\bar{\alpha}^{n} = \theta^{\mathsf{T}}(r_{1}\mathsf{K}_{2}\mathsf{K}_{1} + r_{2}\mathsf{K}_{2}\mathsf{K}_{1})\theta\alpha^{n} + \tau\theta^{\mathsf{T}}\mathsf{K}_{1}\mathsf{K}_{1}\theta\bar{\mathsf{F}}^{n}, \\ \theta^{\mathsf{T}}(\mathsf{K}_{2} - \frac{r_{2}}{2}\mathsf{K}_{1})\theta\alpha^{n+1} = \bar{\alpha}^{n} + \theta^{\mathsf{T}}(\mathsf{K}_{2} - \frac{r_{2}}{2}\mathsf{K}_{1})\theta\alpha^{n}, \end{cases}$$
(44)

where n = 0, 1, 2, 3, ..., N, $\alpha^{0} = \theta^{T} u^{0} = \theta^{T} (u_{1}^{0}, u_{2}^{0}, ..., u_{m}^{0})^{T}$.

After solving α^n from Eq. (44), one gets the POD optimal solutions $u^{*n} = \theta \alpha^n$ (n = 0, 1, 2, ..., N) of problem (P₂).

Theorem 3 Let $u_{i,j}^{*n}$ be the POD solutions $\mathbf{u}^{*n} = \theta \alpha^n$ (n = 0, 1, 2, ..., N) of Eq. (44). $u_{i,j}^n$ be the solutions of Eq. (43). $u(x_i, y_j, t_n)$ be the exact solutions of problem (P₂). If $\|\mathbf{K}_1\|_2 \leq \frac{1}{2}$ and $\|\mathbf{K}_2\|_2 \leq \frac{1}{2}$ are uniformly chosen from $\{u_{i,j}^n\}_{n=1}^N$, we conclude that

$$\left|u_{i,j}^{*n} - u(x_i, y_j, t_n)\right| = O(\lambda_{M+1} \exp(N/d), \tau^2, h_x^4, h_y^4).$$
(45)

6 Numerical examples

In this section, we use the four test problems to demonstrate the advantages of the R-CFDS4 for 1D parabolic equations and the R-ADI-CFDS4 for 2D parabolic equations. Our algorithm is implemented with MATLAB R2017a running on a desktop with Intel Core i7 4790 CPU at 2.93 GHz and 7.98 GB memory.

Example 1 Consider 1D parabolic equation (SP₁)

$$\begin{cases} \frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = 0, & 0 < t < 2, 0 < x < \pi, \\ u(0, t) = u(\pi, t) = 0, & 0 \le t \le 2, \\ u(x, 0) = \sin(x), & 0 \le x \le \pi. \end{cases}$$

The exact solution is $u(x, t) = \exp(-t)\sin(x)$. We take the numerical solutions of CFDS4 with $h = 0.02\pi$, $\tau = 0.01$ at $t = 0.1, 0.2, 0.3, \dots, 2$ as snapshots. It can be obtained by computing that eigenvalue $\lambda_6 \leq 6 \times 10^{-16}$. The number of optimal bases is referred to the paper [31]. By computation in Theorem 1, we know that as long as the initial five or more eigenvectors of matrix $\mathbf{A}\mathbf{A}^{\mathrm{T}}$ are chosen as the optimal basis, the R-CFDS4 can satisfy the desirable accuracy requirement. Numerical solutions of R-CFDS4 and the difference between CFDS4 and R-CFDS4 at t = 0.4, 0.8, 1.2, 1.6, 2.0 have been drawn in Fig. 1, which shows that the results of CFDS4 are in very good agreement with those of R-CFDS4. The errors of CFDS4 and R-CFDS4 at several points and the computational time are also reported in Table 1 and Table 2. This implies that the R-CFDS4 is more efficient than the CFDS4 under guaranteeing with the sufficient accuracy of numerical solutions for solving the 1D parabolic equation. In Fig. 2, the errors of CFDS4 and R-CFDS4 are almost similar. Besides, the difference between the CFDS4 and R-CFDS4 is drawn, which means that R-CFDS4 can achieve almost the same accuracy as CFDS4 under the same nodes and step. The differences between the CFDS4 and R-CFDS4 with five optimal bases are no more than 3×10^{-15} in Fig. 1. In Fig. 3 and Table 3, the R-CFDS4 solution based five POD bases at different moments and details for consuming time at t = 6 have been given, which shows



Table 1 Numerical results of SP₁ at t = 2, with $h = 0.02\pi$, $\tau = 0.01$

x	Exact	R-CFDS4	Error (R-CFDS4)	CFDS4	Error(CFDS4)
0.2 π	0.079548	0.079547	1.3155 e -06	0.079547	1.3155 e -06
0.4 π	0.128712	0.128709	2.1285 e -06	0.128709	2.1285 e -06
0.6 π	0.128712	0.128709	2.1285 e -06	0.128709	2.1285 e -06
0.8 π	0.079548	0.079547	1.3155 e -06	0.079547	1.3155 e -06
Time		0.025744 s		2.695269 s	

Table 2 Numerical results of SP₁ at t = 2, with $h = 0.01\pi$, $\tau = 0.0025$

x	Exact	R-CFDS4	Error (R-CFDS4)	CFDS4	Error (CFDS4)
0.2 π	0.079548	0.079548	8.2217 e -08	0.079548	8.2217 e -08
0.4 π	0.128712	0.128711	1.3303 e -07	0.128711	1.3303 e -07
0.6 π	0.128712	0.128711	1.3303 e -07	0.128711	1.3303 e -07
0.8 π	0.079548	0.079548	8.2217 e -08	0.079548	8.2217 e -08
Time		0.126824 s		49.099276 s	



that the R-CFDS4 can be extended to a time interval that is longer than the time interval on which the snapshots were collected.

To show that Eq. (34) is fourth-order accurate in space and second-order in time, we first let $\tau = 0.0002$, then reduce *h* by factor of 2 each time. The data in Table 4 clearly show that Eq. (34) is fourth-order accurate in space since the maximal error is reduced by a factor



Table 3 The consuming time of two schemes for SP₁ at t = 6, with $h = 0.02\pi$, $\tau = 0.01$

		Seconds
CFDS4	Preparation	0.006603
	Computational time	5.135296
R-CFDS4	Preparation	0.006708
	SVD	0.001014
	Computational time	0.048574

Table 4 The convergence order for R-CFDS4 at $\tau = 0.0002$

Spatial step	size Maximum e	rror Convergence order
$h = 0.1\pi$	1.1028 e -05	
$h = 0.05\pi$	6.8638 e -07	4.0060
$h = 0.025\pi$	4.2021 e -08	4.0298
$h = 0.0125\pi$	1.7803 e -09	4.5609

Table 5 The convergence order for R-CFDS4 at $h = 0.0025\pi$

Temporal step size	Maximum error	Convergence order
$\tau = 0.025$	1.4098 e -05	
$\tau = 0.0125$	3.5244 e -06	2.0003
$\tau = 0.00625$	8.8109 e -07	2.0001
$\tau = 0.003125$	2.2027 e -07	2.0001

about 2^4 each time. In a similar way, the second-order accurate in time of Eq. (34) has been confirmed because the maximal error is reduced by a factor about 2^2 each time at $h = 0.0025\pi$ in Table 5.

Example 2 Consider 1D parabolic equation (SP₂)

$$\begin{cases} \frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = xe^t - 6x, & 0 < t \le 1, 0 < x < 1, \\ u(0, t) = 0, & u(1, t) = 1 + e^t, & 0 \le t \le 1, \\ u(x, 0) = x^3 + x, & 0 \le x \le 1. \end{cases}$$

The exact solution is $u(x, t) = x(x^2 + e^t)$. We take the numerical solutions of CFDS4 with h = 0.02, $\tau = 0.01$ at t = 0.05, 0.1, 0.15, ..., 1 as snapshots. We choose the initial ten vectors



Table 6 Numerical results of SP₂ at t = 1, with h = 0.02, $\tau = 0.01$

x	Exact	R-CFDS4	Error (R-CFDS4)	CFDS4	Error (CFDS4)
0.2	0.552561	0.552561	2.2473 e -07	0.552561	2.2473 e -07
0.4	0.660730	0.660730	3.0933 e -07	0.660730	3.0933 e -07
0.6	0.792848	0.792847	3.8266 e -07	0.792847	3.8266 e -07
0.8	0.954216	0.954216	4.6805 e -07	0.954216	4.6805 e -07
Time		0.013232 s		4.530407 s	



in the U as the optimal basis to ensure desirable accuracy requirement. It is shown that eigenvalue $\lambda_{11} \leq 2 \times 10^{-13}$ by computing. Similar to Example 1, numerical solutions of R-CFDS4 and difference between R-CFDS4 and CFDS4 at t = 0.2, 0.4, 0.6, 0.8, 1.0 have been drawn in Fig. 4. It is not difficult to see that the results of CFDS4 are in excellent agreement with those of R-CFDS4. The errors of CFDS4 and R-CFDS4 are shown in Table 6 and exhibited in Fig. 5. It is not difficult to see that the CFDS4 and R-CFDS4 are basically identical. Furthermore, by comparing the expended CPU time of the R-CFDS4 with that of the CFDS4 in Table 6 and Table 7, the obvious advantages of R-CFDS4 in computational efficiency can be clearly found. The differences between the CFDS4 and R-CFDS4 solution based ten POD bases at different moments and details for consuming time at t = 5

x	Exact	R-CFDS4	Error (R-CFDS4)	CFDS4	Error (CFDS4
0.2	0.552561	0.552561	1.4043 e -08	0.552561	1.4043 e -08
0.4	0.660730	0.660730	1.9333 e -08	0.660730	1.9333 e -08
0.6	0.792848	0.792847	2.3916 e -08	0.792847	2.3916 e -08
0.8	0.954216	0.954216	2.9253 e -08	0.954216	2.9253 e -08
Time		0.153751 s		63.113938 s	

Table 7 Numerical results of SP₂ at t = 1, with h = 0.01, $\tau = 0.0025$



Table 8 The consuming time of two schemes for SP₂ at t = 5, with h = 0.02, $\tau = 0.01$

		Seconds
CFDS4	Preparation	0.005328
	Computational time	6.029794
R-CFDS4	Preparation	0.005478
	SVD	0.002758
	Computational time	0.128595

Table 9 The convergence order for R-CFDS4 at τ = 0.0025

Spatial step size	Maximum error	Convergence order
h = 0.05	0.017367	
h = 0.025	0.001197	3.8588
h = 0.0125	0.000076	3.9772
h = 0.00625	4.841 e -06	3.9726

have been given, which shows that the R-CFDS4 can be extended to a time interval that is longer than the time interval on which the snapshots were collected.

In order to explore whether the convergence order is consistent with the theoretical results, τ is fixed as 0.0025, then *h* is reduced by a factor of 2 each time. The maximal error in Table 9 is reduced by a factor about 2⁴ each time and has confirmed the fourth-order accurate in space of Eq. (34). In a similar way, we fix *h* as 0.0025, the maximal error is reduced by a factor about 2² each time in Table 10, which indicates that the R-CFDS4 is second-order accurate in time.

Table 10 The convergence order for R-CFDS4 at h = 0.0025

Temporal step size	Maximum error	Convergence orde
$\tau = 0.05$	1.6572 e -05	
$\tau = 0.025$	4.1433 e -06	1.9998
$\tau = 0.0125$	1.0359 e -06	1.9997
$\tau = 0.00625$	2.7897 e -07	1.8927



Figure 7 The figures of solutions of R-ADI-CFDS4(left) and difference(right) between R-ADI-CFDS4 and ADI-CFDS4 at t = 4, with $h_x = h_y = 0.02\pi$ and $\tau = 0.02$ for SP_3



Example 3 Consider the 2D parabolic equation (SP₃)

$$\begin{cases} \frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}, & (x, y) \in \Omega, 0 < t \le 2, \\ u(x, y, 0) = \sin x \sin y, & (x, y) \in \Omega, \\ u(x, y, t) = 0, & (x, y) \in \partial\Omega, 0 < t < 2, \end{cases}$$

where $\Omega = \{(x, y); 0 \le x \le \pi, 0 \le y \le \pi\}, \partial \Omega$ denotes the boundary of Ω .

In this example, the exact solution is $u(x, y, t) = e^{-2t} \sin x \sin y$. We take the numerical solutions of ADI-CFDS4 with $h_x = h_y = 0.02\pi$, $\tau = 0.02$ at $t = 0.1, 0.2, 0.3, \dots, 2$ as snapshots. It is shown that eigenvalue $\lambda_6 \leq 2 \times 10^{-15}$ by computing. Similar with Example 1, we also choose five POD bases for our computation of 2D parabolic equations. Numerical solutions of R-ADI-CFDS4 and the difference between R-ADI-CFDS4 and ADI-CFDS4 have been drawn in Fig. 7 and Fig. 8. We can clearly find that the numerical solutions of R-ADI-CFDS4 are in very excellent agreement with ADI-CFDS4. In Table 13 and Table 14, Error 1 is the error of R-ADI-CFDS4, Error 2 is the error of ADI-CFDS4. The errors of

Spatial step size	Maximum error	Convergence orde
$h_x = h_y = 0.1\pi$	2.9853 e -06	
$h_x = h_y = 0.05\pi$	1.8597 e –07	4.0047
$h_x = h_y = 0.025\pi$	1.1557 e –08	4.0082
$h_x = h_y = 0.0125\pi$	6.6468 e -10	4.1199

Table 11 The convergence order for R-ADI-CFDS4 at τ = 0.0001

Table 12 The convergence order for R-ADI-CFDS4 at $h_x = h_y = 0.02\pi$

Temporal step size	Maximum error	Convergence orde
$\tau = 0.05$	6.1037 e -05	
$\tau = 0.025$	1.5258 e -05	2.0001
$\tau = 0.0125$	3.8110 e –06	2.0013
$\tau = 0.00625$	9.4918 e -07	2.0054



Table 13 Numerical results of SP₃ at time t = 2, with $h_x = h_y = 0.02\pi$, $\tau = 0.02$

(x, y)	Exact	R-ADI-CFDS4	Error 1	ADI-CFDS4	Error 2
$(0.2\pi, 0.2\pi)$	0.006328	0.006327	8.4207 e -07	0.006327	8.4207 e -07
$(0.4\pi, 0.4\pi)$	0.016567	0.016564	2.2046 e -06	0.016564	2.2046 e -06
$(0.6\pi, 0.6\pi)$	0.016567	0.016564	2.2046 e -06	0.016564	2.2046 e -06
$(0.8\pi, 0.8\pi)$	0.006328	0.006327	8.4207 e -07	0.006327	8.4207 e -07
Time		0.371496 s		86.610579 s	

ADI-CFDS4 and R-ADI-CFDS4 at several points and the computational time are shown and compared in Table 13 and Table 14, which indicates that R-ADI-CFDS4 could save more computational time than ADI-CFDS4 under guaranteeing the sufficient accuracy of ADI-CFDS4. In Fig. 9 and Fig. 7, it can be found that the numerical solution of R-ADI-CFDS4 is almost similar with the numerical solution of ADI-CFDS4. In Fig. 8, the error of ADI-CFDS4 is depicted, which shows the sufficient accuracy of ADI-CFDS4. The differences between the ADI-CFDS4 and R-ADI-CFDS4 with five optimal bases are no more than 3×10^{-14} in Fig. 9 and Fig. 7. In Table 15, the details for consuming time at t = 4 have been given, which shows the advantages of R-ADI-CFDS4.

In order to verify the convergence order of Eq. (42), we fix τ as 0.0001 in Table 11. The maximal error is reduced by a factor about 2⁴ each time, which shows the R-ADI-CFDS4 is fourth-order accurate in space. Similarly, the data in Table 12 have confirmed the second-order accurate in time of R-ADI-CFDS4.

(<i>x</i> , <i>y</i>)	Exact	R-ADI-CFDS4	Error 1	ADI-CFDS4	Error 2
$(0.2\pi, 0.2\pi)$	0.006328	0.006328	5.2630 e -08	0.006328	5.2630 e -08
$(0.4\pi, 0.4\pi)$	0.016567	0.016567	1.3779 e –07	0.016567	1.3779 e –07
$(0.6\pi, 0.6\pi)$	0.016567	0.016567	1.3779 e –07	0.016567	1.3779 e –07
(0.8 <i>π</i> , 0.8 <i>π</i>) Time	0.006328	0.006328 5.321289 s	5.2630 e -08	0.006328 1004.683564 s	5.2630 e -08

Table 14 Numerical results of SP₃ at t = 2, with $h_x = h_y = 0.01\pi$, $\tau = 0.005$

Table 15 The consuming time of two schemes for SP₃ at t = 4, with $h_x = h_y = 0.02\pi$, $\tau = 0.02$

		Seconds
ADI-CFDS4	Preparation	0.792100
	Computational time	173.644678
R-ADI-CFDS4	Preparation	0.798521
	SVD	0.056670
	Computational time	0.874024

Table 16 Numerical results of SP₄ at t = 2, with $h = 0.02\pi$, $\tau = 0.02$

(<i>x</i> , <i>y</i>)	Exact	R-ADI-CFDS4	Error 1	ADI-CFDS4	Error 2
$(0.2\pi, 0.2\pi)$	0.046757	0.046758	6.7914 e -07	0.046758	6.7914 e -07
$(0.4\pi, 0.4\pi)$	0.122412	0.122414	1.7780 e -06	0.122414	1.7780 e -06
$(0.6\pi, 0.6\pi)$	0.122412	0.122414	1.7780 e -06	0.122414	1.7780 e -06
$(0.8\pi, 0.8\pi)$	0.046757	0.046758	6.7914 e -07	0.046758	6.7914 e -07
Time		0.361219 s		80.801341 s	

Table 17 Numerical results of SP₄ at t = 2, with $h = 0.01\pi$, $\tau = 0.005$

(<i>x</i> , <i>y</i>)	Exact	R-ADI-CFDS4	Error 1	ADI-CFDS4	Error 2
$(0.2\pi, 0.2\pi)$	0.046757	0.046757	4.2442 e -08	9.246652	4.2442 e -08
$(0.4\pi, 0.4\pi)$	0.122412	0.122412	1.1112 e -07	24.208049	1.1112 e -07
$(0.6\pi, 0.6\pi)$	0.122412	0.122412	1.1112 e -07	24.208049	1.1112 e -07
$(0.8\pi, 0.8\pi)$	0.046757	0.046757	4.2442 e -08	9.246652	4.2442 e -08
Time		5.349663 s		1049.7368756 s	

Example 4 Consider the 2D parabolic equation (SP₄)

$$\begin{cases} \frac{\partial u}{\partial t} - \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = e^{-t}\sin(x)\sin(y), (x, y) \in \Omega, \ 0 < t \le 2, \\ u(x, y, 0) = \sin(x)\sin(y), (x, y) \in \Omega, \\ u(x, y, t) = 0, (x, y) \in \partial\Omega, \ 0 < t \le 2. \end{cases}$$

 $Ω = \{(x,y); 0 \le x \le \pi, 0 \le y \le \pi\}, ∂Ω$ denotes the boundary of Ω. The exact solution is $u(x, y, t) = e^{-t} \sin x \sin y$. We take the numerical solutions of ADI-CFDS4 with $h_x = h_y = 0.02\pi$, $\tau = 0.02$ at t = 0.1, 0.2, 0.3, ..., 2 as snapshots. It is shown that eigenvalue $λ_6 \le 7 \times 10^{-15}$ by computing. Numerical solutions of R-ADI-CFDS4 and difference between R-ADI-CFDS4 and ADI-CFDS4 have been drawn in Fig. 10. We can clearly find that the approximate solutions of R-ADI-CFDS4 are in very excellent agreement with ADI-CFDS4. Error 1 and Error 2 are the same as in Example 3. The errors of several points and computational time are written in Table 16 and Table 17, which shows the high efficiency and reliability of R-ADI-CFDS4 with five optimal bases are no more than 4×10^{-14} in Fig. 10 and Fig. 11. R-ADI-CFDS4 solution based five POD bases at t = 4 are given in



Table 18 The consuming time of two schemes for SP₄ at t = 4, with $h_x = h_y = 0.02\pi$, $\tau = 0.02$

		Seconds
ADI-CFDS4	Preparation	0.759387
	Computational time	152.172127
R-ADI-CFDS4	Preparation	0.759521
	SVD	0.369322
	Computational time	0.887663



Table 19 The convergence order for R-ADI-CFDS4 at τ = 0.0001

Spatial step size	Maximum error	Convergence order
$h_x = h_y = 0.2\pi$	1.3969 e -04	
$h_x = h_y = 0.1\pi$	9.5413 e -06	3.8719
$h_x = h_y = 0.05\pi$	5.9914 e -07	3.9932
$h_x = h_y = 0.025\pi$	4.1990 e -08	3.8348

Fig. 11, while the details for consuming time at t = 4 are listed in Table 18. Meanwile, we also list the error of ADI-CFDS4 at t = 2 in Fig. 12.

The convergence order of R-ADI-CFDS4 is also verified. In Table 19, we first fix τ as 0.0001. Next, h_x and h_y are reduced a factor of 2 each time, the R-ADI-CFDS4 is fourth-order accurate in space since the maximal error is reduced by a factor about 2⁴ each time. Then, in Table 20, we take $h_x = h_y$ as 0.02π . The τ is reduced a factor of 2 each time,

Temporal step size	Maximum error	Convergence order
$\tau = 0.2$	1.9686 e -04	
$\tau = 0.1$	4.8886 e -05	2.0097
$\tau = 0.05$	1.2212 e -05	2.0011
$\tau = 0.025$	3.0630 e -06	1.9953

Table 20 The convergence order for R-ADI-CFDS4 at $h_x = h_y = 0.02\pi$



R-ADI-CFDS4 is second-order accurate in time since the maximal error is reduced by a factor about 2^2 each time.

7 Conclusions

In this paper, we developed a reduced fourth-order compact difference scheme for solving parabolic equations. The efficiency and accuracy of the proposed algorithm were examined by two 1D problems and two 2D problems. The numerical examples illustrated that the fourth-order compact difference scheme coupled with POD technique not only keeps high computational accuracy, but also brings significant computational time saving for solving parabolic equation. In the future, we plan to improve our algorithm to solve more complicated parabolic equations in three dimensions.

Appendix: Stability analysis

To study the stability of our scheme, we assume f = 0. Equation (22) can be written as $Au^{k+1} = Bu^k$, where

$$\boldsymbol{A} = \begin{pmatrix} \frac{5}{6} + r & \frac{1}{12} - \frac{r}{2} & 0 \\ \frac{1}{12} - \frac{r}{2} & \frac{5}{6} + r & \frac{1}{12} - \frac{r}{2} & \\ & \ddots & \ddots & \ddots & \\ & & \frac{1}{12} - \frac{r}{2} & \frac{5}{6} + r & \frac{1}{12} - \frac{r}{2} \\ & 0 & & \frac{1}{12} - \frac{r}{2} & \frac{5}{6} + r \end{pmatrix},$$
$$\boldsymbol{B} = \begin{pmatrix} \frac{5}{6} - r & \frac{1}{12} + \frac{r}{2} & 0 & \\ \frac{1}{12} + \frac{r}{2} & \frac{5}{6} - r & \frac{1}{12} + \frac{r}{2} & 0 \\ & & \frac{1}{12} + \frac{r}{2} & \frac{5}{6} - r & \frac{1}{12} + \frac{r}{2} \\ & & 0 & & \frac{1}{12} + \frac{r}{2} & \frac{5}{6} - r & \frac{1}{12} + \frac{r}{2} \\ & & 0 & & \frac{1}{12} + \frac{r}{2} & \frac{5}{6} - r \end{pmatrix}.$$

By computing the formula of eigenvectors, the eigenvectors of **A** and **B** can be written as follows:

$$\lambda_{\mathbf{A},j} = \frac{5}{6} + r + 2\left(\frac{1}{12} - \frac{r}{2}\right)\cos\frac{j\pi}{J}, \quad 1 \le j \le J - 1,$$

and

$$\lambda_{\mathbf{B},j} = \frac{5}{6} - r + 2\left(\frac{1}{12} + \frac{r}{2}\right)\cos\frac{j\pi}{J}, \quad 1 \le j \le J - 1.$$

It is easy to conclude that the eigenvectors of $\mathbf{A}^{-1}\mathbf{B}$ are

$$\lambda_{j} = \frac{\lambda_{\boldsymbol{B},j}}{\lambda_{\boldsymbol{A},j}} = \frac{\frac{5}{6} - r + 2(\frac{1}{12} + \frac{r}{2})\cos\frac{j\pi}{J}}{\frac{5}{6} + r + 2(\frac{1}{12} - \frac{r}{2})\cos\frac{j\pi}{J}}, \quad 1 \le j \le J - 1.$$

The $|\lambda_i| \leq 1$ is equivalent to

$$\left|\frac{5}{6} - r + 2\left(\frac{1}{12} + \frac{r}{2}\right)\cos\frac{j\pi}{J}\right| \le \left|\frac{5}{6} + r + 2\left(\frac{1}{12} - \frac{r}{2}\right)\cos\frac{j\pi}{J}\right|.$$

It is easy to see that when r > 0, there is the following inequality:

$$\left|\frac{5}{6} + \frac{1}{6}\cos\frac{j\pi}{J} - r\left(1 - \cos\frac{j\pi}{J}\right)\right|^2 \le \left|\frac{5}{6} + \frac{1}{6}\cos\frac{j\pi}{J} + r\left(1 - \cos\frac{j\pi}{J}\right)\right|^2, \quad 1 \le j \le J - 1.$$

Because $r = \frac{a\tau}{h^2} > 0$, we can conclude that $|\lambda_j| \le 1$ and $\rho(A^{-1}B) \le 1$. From that, we can deduce that Eq. (15) is unconditionally stable.

Acknowledgements

The authors express their sincere thanks to the anonymous reviews for their valuable suggestions and corrections for improving the quality of this paper.

Funding

This work is financially supported by the Academic Mainstay Foundation of Hubei Province of China (No. D20171202) and the National Natural Science Foundation of China (Grant No. 11826208).

Availability of data and materials

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

All authors contributed equally and significantly in writing this article. All authors read and approved the final manuscript.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 6 December 2018 Accepted: 16 July 2019 Published online: 26 July 2019

References

- 1. Qiu, T.Q., Tien, C.L.: Short-pulse laser heating on metals. Int. J. Heat Mass Transf. 35, 719–726 (1992)
- 2. Faghri, A., Zhang, Y.: Transport Phenomena in Multiphase Systems. Academic Press, Boston (2006)
- 3. Tzou, D.Y., Dai, W.: Thermal lagging in multi-carrier systems. Int. J. Heat Mass Transf. 52, 1206–1213 (2009)
- 4. Enyi, C.D.: Robust exponential attractors for a parabolic–hyperbolic phase-field system. Bound. Value Probl. 2018, 146 (2018)

- 5. Grabowski, P.: Small-gain theorem for a class of abstract parabolic systems. Opusc. Math. 38, 651–680 (2018)
- 6. Papageorgiou, N.S., Radulescu, V.D., Repovs, D.D.: Nonlinear Analysis—Theory and Methods. Springer, Berlin (2019)
- Martín-Vaquero, J., Sajavičius, S.: The two-level finite difference schemes for the heat equation with nonlocal initial condition. Appl. Math. Comput. 342, 166–177 (2019)
- Zhou, C., Zou, Y., Chai, S., Zhang, Q., Zhu, H.: Weak Galerkin mixed finite element method for heat equation. Appl. Numer. Math. 123, 180–199 (2018)
- 9. Hermeline, F.: A finite volume method for the approximation of diffusion operators on distorted meshes. J. Comput. Phys. **160**, 481–499 (2000)
- Pu, Z., Hai, P.: The error analysis of Crank–Nicolson-type difference scheme for fractional subdiffusion equation with spatially variable coefficient. Bound. Value Probl. 2017, 15 (2017)
- Wang, H., Wang, J., Li, S.: A new conservative nonlinear high-order compact finite difference scheme for the general Rosenau-RLW equation. Bound. Value Probl. 2015, 77 (2015)
- 12. Zhao, J., Corless, R.: Compact finite difference method for integro-differential equations. Appl. Math. Comput. 177, 271–288 (2006)
- Liao, W.: An implicit fourth-order compact finite difference scheme for one-dimensional Burgers' equation. Appl. Math. Comput. 206, 755–764 (2008)
- 14. Sutmann, G.: Compact finite difference schemes of sixth order for the Helmholtz equation. J. Comput. Appl. Math. 203, 15–31 (2007)
- Shukla, R., Tatineni, M., Zhong, X.: Very high-order compact finite difference schemes on non-uniform grids for incompressible Navier–Stokes equations. J. Comput. Phys. 224, 1064–1094 (2007)
- Gao, Z., Xie, S.: Fourth-order alternating direction implicit compact finite difference schemes for two-dimensional Schrödinger equations. Appl. Numer. Math. 61, 593–614 (2011)
- Sutmann, G., Steffen, B.: High-order compact solvers for the three-dimensional Poisson equation. J. Comput. Appl. Math. 187, 142–170 (2006)
- Mohebbi, A., Dehghan, M.: High-order solution of one-dimensional sine-Gordon equation using compact finite difference and DIRKN methods. Math. Comput. Model. 51, 537–549 (2010)
- 19. Zhang, P., Zhang, X., Xiang, H., Song, L.: A fast and stabilized meshless method for the convection-dominated convection-diffusion problems. Numer. Heat Transf., Part A, Appl. **70**, 420–431 (2016)
- 20. Luo, Z., Chen, G.: Proper Orthogonal Decomposition Methods for Partial Differential Equations. Academic Press, New York (2018)
- Liang, Y.C., Lee, H.P., Lim, S.P., Lin, W.Z., Lee, K.H., Wu, C.G.: Proper orthogonal decomposition and its applications—part I: theory. J. Sound Vib. 252. 527–544 (2002)
- Rathinam, M., Petzold, L.: A new look at proper orthogonal decomposition. SIAM J. Numer. Anal. 41, 1893–1925 (2003)
- 23. Kerschen, G., Golinval, J., Vakakis, A., Bergman, L: The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: an overview. Nonlinear Dyn. **41**, 147–169 (2005)
- Luo, Z.: A POD-based reduced-order TSCFE extrapolation iterative format for two-dimensional heat equations. Bound. Value Probl. 2015, 59 (2015)
- Sun, P., Luo, Z., Zhou, Y.: Some reduced finite difference schemes based on a proper orthogonal decomposition technique for parabolic equations. Appl. Numer. Math. 60, 154–164 (2010)
- Luo, Z., Li, H., Sun, P.: A reduced-order Crank–Nicolson finite volume element formulation based on POD method for parabolic equations. Appl. Math. Comput. 219, 5887–5900 (2013)
- 27. An, J., Luo, Z., Li, H., Sun, P.: Reduced-order extrapolation spectral-finite difference scheme based on POD method and error estimation for three-dimensional parabolic equation. Front. Math. China **10**, 1025–1040 (2015)
- Luo, Z., Li, H., Sun, P.: A reduced-order Crank–Nicolson finite volume element formulation based on POD method for parabolic equations. Appl. Math. Comput. 219, 5887–5900 (2013)
- 29. Luo, Z., Chen, J., Navon, I.M., Yang, X.: Mixed finite element formulation and error estimates based on proper orthogonal decomposition for the nonstationary Navier–Stokes equations. SIAM J. Numer. Anal. 47, 1–19 (2008)
- Luo, Z., Li, H., Sun, P., Gao, J.: A reduced-order finite difference extrapolation algorithm based on POD technique for the non-stationary Navier–Stokes equations. Appl. Math. Model. 37, 5464–5473 (2013)
- Luo, Z., Teng, F.: A reduced-order extrapolated finite difference iterative scheme based on POD method for 2D Sobolev equation. Appl. Math. Comput. 329, 374–383 (2018)
- 32. Luo, Z., Teng, F.: An optimized SPDMFE extrapolation approach based on the POD technique for 2D viscoelastic wave equation. Bound. Value Probl. 2017, 6 (2017)
- 33. Luo, Z., Jin, S., Chen, J.: A reduced-order extrapolation central difference scheme based on POD for two-dimensional fourth-order hyperbolic equations. Appl. Math. Comput. **289**, 396–408 (2016)
- Luo, Z., Yang, X., Zhou, Y.: A reduced finite difference scheme based on singular value decomposition and proper orthogonal decomposition for Burgers equation. Appl. Math. Comput. 229, 97–107 (2016)
- Dehghan, M., Abbaszadeh, M.: An upwind local radial basis functions-differential quadrature (RBF-DQ) method with proper orthogonal decomposition (POD) approach for solving compressible Euler equation. Eng. Anal. Bound. Elem. 92, 244–256 (2018)
- Dehghan, M., Abbaszadeh, M.: A reduced proper orthogonal decomposition (POD) element free Galerkin (POD-EFG) method to simulate two-dimensional solute transport problems and error estimate. Appl. Numer. Math. 126, 92–112 (2018)
- Dehghan, M., Abbaszadeh, M.: A combination of proper orthogonal decomposition discrete empirical interpolation method (POD-DEIM) and meshless local RBF-DQ approach for prevention of groundwater contamination. Comput. Math. Appl. 75, 1390–1412 (2018)
- Zhang, X., Xiang, H.: A fast meshless method based on proper orthogonal decomposition for the transient heat conduction problems. Int. J. Heat Mass Transf. 84, 729–739 (2015)
- Fragnelli, G., Mugnai, D.: Carleman estimates for singular parabolic equations with interior degeneracy and non-smooth coefficients. Adv. Nonlinear Anal. 6, 61–84 (2017)
- 40. Hua, D., Li, X.: Numerical Method and Program Realization of Differential Equation. Publishing House of Electronics Industry, Beijing (2016) (in Chinese)

- Luo, Z., Chen, J., Zhu, J., Wang, R.: An optimizing reduced order FDS for the tropical Pacific Ocean reduced gravity model. Int. J. Numer. Methods Fluids 55, 143–161 (2007)
- 42. Luo, Z., Wang, R., Zhu, J.: Finite difference scheme based on proper orthogonal decomposition for the nonstationary Navier–Stokes equations. Sci. China Ser. A, Math. **50**, 1186–1196 (2007)

Submit your manuscript to a SpringerOpen[●] journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com