Preconditioned quasi-compact boundary value methods for space-fractional diffusion equations

Numer Algo manuscript No.
(will be inserted by the editor)

Preconditioned quasi-compact boundary value methods for space-fractional diffusion equations

Yongtao Zhou[1,2] · **Chengjian Zhang**[1,2] · **Luigi Brugnano**[3]

Received: date / Accepted: date

Abstract This paper focuses on highly-efficient numerical methods for solving space-fractional diffusion equations. By combining the fourth-order quasi-compact difference scheme and boundary value methods, a class of quasi-compact boundary value methods are constructed. In order to accelerate the convergence rate of this class of methods, the Kronecker product splitting (KPS) iteration method and the preconditioned method with KPS preconditioner are proposed. A convergence criterion for the KPS iteration method is derived. A numerical experiment further illustrates the computational efficiency and accuracy of the proposed methods. Moreover, a numerical comparison with the preconditioned method with Strang-type preconditioner is given, which shows that the preconditioned method with KPS preconditioner is comparable in computational efficiency.

Keywords Space-fractional diffusion equations · Quasi-compact difference scheme · Boundary value methods · KPS preconditioner · Preconditioned methods

This work is supported by NSFC (Grant No. 11571128).

Yongtao Zhou
E-mail: yongtaozh@126.com
(✉) Chengjian Zhang
E-mail: cjzhang@mail.hust.edu.cn
Luigi Brugnano
E-mail: luigi.brugnano@unifi.it
[1] School of Mathematics and Statistics, Huazhong University of Science and Technology, Wuhan 430074, China
[2] Hubei Key Laboratory of Engineering Modeling and Scientific Computing, Huazhong University of Science and Technology, Wuhan 430074, China
[3] Dipartimento di Matematica e Informatica "U. Dini", Università di Firenze, Viale Morgagni 67/A, Firenze I-50134, Italy

1 Introduction

We consider the following initial-boundary value problem of two-sided space-fractional diffusion equations (SFDEs):

$$\begin{cases} \dfrac{\partial u(x,t)}{\partial t} = d^{+}\,{}_{x_L}D_x^{\alpha}u(x,t) + d^{-}\,{}_{x}D_{x_R}^{\alpha}u(x,t) + f(x,t), \quad (x,t) \in (x_L,x_R)\times(t_0,T], \\ u(x,t_0) = \varphi(x), \quad x \in [x_L,x_R], \\ u(x_L,t) = u(x_R,t) = 0, \quad t \in [t_0,T], \end{cases} \tag{1.1}$$

where $f(x,t)$ is the source term, $d^{\pm}$ are nonnegative diffusion coefficients, $\varphi(x)$ is an assigned initial function, and ${}_{x_L}D_x^{\alpha}u(x,t)$ and ${}_{x}D_{x_R}^{\alpha}u(x,t)$ are the left-sided and right-sided Riemann-Liouville fractional derivatives of order $\alpha \in (1,2)$, respectively, which are defined by (see e.g. [35])

$${}_{x_L}D_x^{\alpha}u(x,t) = \frac{\frac{d^2}{dx^2}\int_{x_L}^{x}\frac{u(\xi,t)}{(x-\xi)^{\alpha-1}}d\xi}{\Gamma(2-\alpha)}, \quad {}_{x}D_{x_R}^{\alpha}u(x,t) = \frac{\frac{d^2}{dx^2}\int_{x}^{x_R}\frac{u(\xi,t)}{(\xi-x)^{\alpha-1}}d\xi}{\Gamma(2-\alpha)}.$$

It is known from [38] that the Riesz fractional derivative can be reviewed as a linear combination of the left-sided and right-sided Riemann-Liouville fractional derivatives, namely,

$$\frac{\partial^{\alpha}u(x,t)}{\partial|x|^{\alpha}} = -\frac{1}{2\cos(\pi\alpha/2)}\left[{}_{x_L}D_x^{\alpha}u(x,t) + {}_{x}D_{x_R}^{\alpha}u(x,t)\right].$$

Hence, when

$$d^{+} = d^{-} = -\frac{\kappa}{2\cos(\pi\alpha/2)}, \quad \kappa > 0, \tag{1.2}$$

(1.1) becomes a problem of SFDEs with Riesz fractional derivatives. Moreover, SFDEs in (1.1) are also viewed as the generalizations of the classical second order diffusion equations.

A number of researches show that the problem (1.1) of SFDEs is able to describe some scientific and engineering phenomena better than the classical second order diffusion model (see e.g. [3,39]) and it has many applications in other settings, such as in finance, image processing, electrochemistry, biological systems, hydrology, etc. (cf. [2,20,21,33,36]). However, owing to the nonlocal nature of the fractional derivatives, it is difficult to derive a closed form for the solution of SFDEs. Therefore, developing highly-efficient numerical methods to solve SFDEs has become an important issue. For this topic, in recent year, some progresses have been made. For example, Meerschaert and Tadjeran [32] proposed to use the shifted Grünwald-Letnikov formula to approximate the Riemann-Liouville fractional derivatives. In order to raise the computational accuracy of the algorithms, based on Lubich's weighted-shifted difference operator (cf. [31]), Chen and Deng [17,18] gave the second-, third- and fourth-order approximation formulas for the Riemann-Liouville fractional derivatives and applied them to solve SFDEs. By combining the weighted Grünwald-Letnikov formula and a compact difference formula, Hao, Sun and Cao [23] developed a quasi-compact method with fourth-order accuracy in space for one- and two-dimensional

SFDEs. Most of these methods mainly concern the improvement of spatial accuracy. As to the improvement on temporal accuracy, it is remarkable to mention that Lei and Huang [27] combine the pth-order BVMs (cf. [7]) with the fourth-order quasi-compact weighed-shifted Grünwald-Letnikov difference method to construct a class of numerical methods with convergence order $\mathscr{O}(h^4+\tau^p)$, where h and τ are the spatial and temporal stepsizes of the methods, respectively, and the temporal convergence order p can be appropriately large.

Generally speaking, a high-accuracy method often needs a large cost (see e.g. [41]). Hence, improving the computational efficiency of the numerical methods is also an important research issue. Recently, several fast iterative techniques have been presented for finite difference methods for solving SFDEs, such as the KPS iteration method (cf. [13–16]), the Strang-type preconditioned method (cf. [22,27]), the conjugate gradient normal residual method with circulant preconditioner (cf. [28]), the multigrid method (cf. [34]), the super fast-preconditioned iterative method (cf. [40]), and so forth. These methods accelerate the convergence rate of the used finite difference methods, while preserving their accuracy. Motivated by the above researches, this paper deals with preconditioned methods for solving problem (1.1).

With this premises, the structure of this paper is as follows. In Section 2, by combining a four-order quasi-compact difference method and BVMs, a class of quasi-compact BVMs for (1.1) are constructed. In Section 3, in order to raise the computational efficiency of the proposed methods, we consider the KPS iteration method and the generalized minimal residual (GMRES) method with KPS preconditioner. Moreover, a convergence criterion for the KPS iteration method is derived. In Section 4, the computational efficiency and accuracy of the proposed methods is assessed by some numerical experiments, also providing numerical comparisons with the GMRES method with a Strang-type preconditioner.

2 A class of quasi-compact BVMs for SFDEs

In this section, we will combine the fourth-order quasi-compact scheme (cf. [23]) and BVMs (cf. [7]) to construct a class of space-time discretization methods for solving the initial-boundary value problem (1.1).

The basic idea, on which the fourth-order quasi-compact scheme relies, is using the weighted average to zero the low order leading terms in the asymptotic expansion of the truncation error in the shifted Grünwald-Letnikov formula. In order to give the scheme, we need to introduce a preparatory result from Hao, Sun and Cao [23]. For this, a quasi-compact operator $\mathscr{A}$ is defined as follows:

$$\mathscr{A}v(x)=c_\alpha v(x-h)+(1-2c_\alpha)v(x)+c_\alpha v(x+h),\quad h>0, \tag{2.1}$$

where $v(x)$ is an any given function and $c_\alpha=(-\alpha^2+\alpha+4)/24$ which, for $\alpha\in(1,2)$, satisfies:

$$\frac{1}{12}<c_\alpha<\frac{1}{6}. \tag{2.2}$$

Lemma 1 (cf. [23]) *Suppose that*

$$v(x)\in\mathscr{L}^{4+\alpha}(\mathbb{R}):=\left\{v(x)\in L_1(\mathbb{R}):\int_{-\infty}^{+\infty}(1+|\xi|)^{4+\alpha}\left|\int_{-\infty}^{+\infty}e^{i\xi x}v(x)dx\right|d\xi<\infty\right\}.$$

Then, the quasi-compact operator $\mathscr{A}$ satisfies, for all $x\in\mathbb{R}$,

$$\mathscr{A}[{}_{-\infty}D_x^{\alpha}v(x)]=\delta_{x,+}^{\alpha}v(x)+\mathscr{O}(h^4),\quad \mathscr{A}[{}_xD_{+\infty}^{\alpha}v(x)]=\delta_{x,-}^{\alpha}v(x)+\mathscr{O}(h^4),\quad h\to 0,$$

where $\delta_{x,\pm}^{\alpha}v(x)=\frac{1}{h^{\alpha}}\sum\limits_{l=0}^{+\infty}w_l^{(\alpha)}v(x\mp(l-1)h)$ with

$$w_0^{(\alpha)}=\frac{\alpha^2+3\alpha+2}{12},\quad w_1^{(\alpha)}=-\frac{(\alpha-1)(\alpha+2)(\alpha+4)}{12},$$

$$w_l^{(\alpha)}=\left[\frac{(\alpha-1)(\alpha+1)(\alpha+2)(\alpha+4)}{12(l-1)}-\frac{(\alpha+1)^2(\alpha+2)^2}{12l}+1\right](-1)^l\binom{\alpha}{l-2},\quad l\geq 2.$$

In addition, when $\alpha\in(1,2)$, the coefficients $\{w_l^{(\alpha)}\}$ have the following properties:

$$\begin{cases} w_0^{(\alpha)}>0,\quad w_1^{(\alpha)}<0,\quad w_l^{(\alpha)}>0,\ \ l\geq 3,\\ \sum\limits_{l=0}^{+\infty}w_l^{(\alpha)}=0,\quad \sum\limits_{l=0}^{m}w_l^{(\alpha)}<0,\ \ m\geq 1,\\ w_0^{(\alpha)}+w_2^{(\alpha)}>0. \end{cases}\tag{2.3}$$

Let function $v(x)$ be defined on $[x_L,x_R]$ with $v(x_L)=v(x_R)=0$, and then make a zero-extension of $v(x)$ on $\mathbb{R}$, where it is assumed that $v(x)\in\mathscr{L}^{4+\alpha}(\mathbb{R})$. In this way, the following equalities hold:

$$\delta_{x,+}^{\alpha}v(x)=\frac{1}{h^{\alpha}}\sum_{l=0}^{\lfloor\frac{x-x_L}{h}\rfloor}w_l^{(\alpha)}v(x-(l-1)h),\ \ \delta_{x,-}^{\alpha}v(x)=\frac{1}{h^{\alpha}}\sum_{l=0}^{\lfloor\frac{x_R-x}{h}\rfloor}w_l^{(\alpha)}v(x+(l-1)h),\tag{2.4}$$

and thus, by Lemma 1, one has that

$$\mathscr{A}[{}_{x_L}D_x^{\alpha}v(x)]=\delta_{x,+}^{\alpha}v(x)+\mathscr{O}(h^4),\quad \mathscr{A}[{}_xD_{x_R}^{\alpha}v(x)]=\delta_{x,-}^{\alpha}v(x)+\mathscr{O}(h^4).\tag{2.5}$$

Taking a spatial stepsize $h=\frac{x_R-x_L}{M}$ ($M\in\mathbb{N}$) and grid points $x_i=x_L+ih$ ($0\leq i\leq M$), then (1.1) implies that

$$\frac{\partial u(x_i,t)}{\partial t}=d^{+}\,{}_{x_L}D_x^{\alpha}u(x_i,t)+d^{-}\,{}_xD_{x_R}^{\alpha}u(x_i,t)+f(x_i,t),\quad 0\leq i\leq M.$$

Acting operator $\mathscr{A}$ on both sides of the above equations yields

$$\mathscr{A}\frac{\partial u(x_i,t)}{\partial t}=d^{+}\mathscr{A}[{}_{x_L}D_x^{\alpha}u(x_i,t)]+d^{-}\mathscr{A}[{}_xD_{x_R}^{\alpha}u(x_i,t)]+\mathscr{A}f(x_i,t),\quad 1\leq i\leq M-1.$$

It follows from (2.5) that

$$\mathscr{A}\frac{\partial u(x_i,t)}{\partial t}=d^{+}\delta_{x,+}^{\alpha}u(x_i,t)+d^{-}\delta_{x,-}^{\alpha}u(x_i,t)+\mathscr{A}f(x_i,t)+\mathscr{O}(h^4),\ \ 1\leq i\leq M-1.\tag{2.6}$$

Introducing the notation:

$$\hat{u}(t) = (u(x_1,t), u(x_2,t), \ldots, u(x_{M-1},t))^T,$$

$$J = \frac{1}{h^\alpha}\left(d^+ W_\alpha + d^- W_\alpha^T\right), \quad W_\alpha = \begin{bmatrix} w_1^{(\alpha)} & w_0^{(\alpha)} & & & \\ w_2^{(\alpha)} & w_1^{(\alpha)} & w_0^{(\alpha)} & & \\ \vdots & \vdots & \ddots & \ddots & \\ w_{M-2}^{(\alpha)} & w_{M-3}^{(\alpha)} & \cdots & w_1^{(\alpha)} & w_0^{(\alpha)} \\ w_{M-1}^{(\alpha)} & w_{M-2}^{(\alpha)} & \cdots & w_2^{(\alpha)} & w_1^{(\alpha)} \end{bmatrix},$$

$$f(t) = (f(x_1,t), f(x_2,t), \ldots, f(x_{M-1},t))^T, \quad r(t) = (c_\alpha f(x_0,t), 0, \ldots, 0, c_\alpha f(x_M,t))^T$$

and

$$K = \begin{bmatrix} 1-2c_\alpha & c_\alpha & & & \\ c_\alpha & 1-2c_\alpha & c_\alpha & & \\ & \ddots & \ddots & \ddots & \\ & & c_\alpha & 1-2c_\alpha & c_\alpha \\ & & & c_\alpha & 1-2c_\alpha \end{bmatrix} \in \mathbb{R}^{(M-1)\times(M-1)},$$

then (2.6) can be cast in the following matrix form:

$$K\hat{u}'(t) = J\hat{u}(t) + Kf(t) + r(t) + \mathscr{O}(h^4), \ \ t \in [t_0, T]. \tag{2.7}$$

For matrices K and J, the following results hold.

Lemma 2 *Matrix K is symmetric-positive-definite and has the decomposition form: $K = C^T C$ with*

$$\|C\|, \|C^{-1}\| < \sqrt{2}. \tag{2.8}$$

Moreover, matrix J is symmetric-negative-definite.

Proof. We can write matrix K as

$$K = (1-2c_\alpha)\begin{pmatrix} 1 & \sigma_\alpha & & & \\ \sigma_\alpha & \ddots & \ddots & & \\ & \ddots & \ddots & \sigma_\alpha & \\ & & \sigma_\alpha & 1 \end{pmatrix} =: (1-2c_\alpha)\hat{K}, \qquad \sigma_\alpha = \frac{c_\alpha}{1-2c_\alpha}. \tag{2.9}$$

On the other hand, one has that

$$\hat{K} = LDL^T, \qquad L = \begin{pmatrix} 1 & & & \\ \sigma_\alpha d_1^{-1} & 1 & & \\ & \ddots & \ddots & \\ & & \sigma_\alpha d_{M-2}^{-1} & 1 \end{pmatrix}, \qquad D = \begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_{M-1} \end{pmatrix},$$

with $d_1 = 1$, $d_i = 1 - \sigma_\alpha^2/d_{i-1}$, $i = 2,\ldots,M-2$. By taking into account (2.2) and (2.9) one then obtains $\frac{1}{10} < \sigma_\alpha < \frac{1}{4}$ and, consequently (cf. [1,5]),

$$1 \geq d_i \geq \frac{1+\sqrt{1-4\sigma_\alpha^2}}{2} \qquad \Rightarrow \qquad 0 < \sigma_\alpha d_i^{-1} < \frac{1}{2+\sqrt{3}}.$$

It then follows that:

$$\|L\| < \frac{3+\sqrt{3}}{2+\sqrt{3}},\ \|D\| = 1 \ \Rightarrow\ \|C\| < \frac{3+\sqrt{3}}{2+\sqrt{3}} < \sqrt{2}, \tag{2.10}$$

$$\|L^{-1}\| < \frac{2+\sqrt{3}}{1+\sqrt{3}},\ \|D^{-\frac{1}{2}}\| < \frac{2}{\sqrt{2+\sqrt{3}}} \ \Rightarrow\ \|C^{-1}\| \leq \|L^{-1}\|\,\|D^{-\frac{1}{2}}\| < \frac{2\sqrt{2+\sqrt{3}}}{1+\sqrt{3}} = \sqrt{2}.$$

Concerning matrix J, its symmetry is clear from (1.2), and its negative definiteness then follows from (2.3), as it can be found in [27, Lemma 2.4]. □

As a simple consequence of the above lemma (see also [27]), one obtains the following result.

Lemma 3 *When $\alpha \in (1,2)$, all eigenvalues of matrix $K^{-1}J$ are in the left-half complex plane $\mathbb{C}^- := \{q \in \mathbb{C} : \Re(q) < 0\}$.*

From Lemma 2, considering that the bounds (2.10) are independent of M (i.e., of the space-step h), we obtain that (2.7) is equivalent to:

$$\bar{u}'(t) = \bar{J}\bar{u}(t) + \bar{f}(t) + \bar{r}(t) + \mathscr{O}(h^4),\ \ t \in [t_0, T], \tag{2.11}$$

where $\bar{J} = C^{-T}JC^{-1}, \bar{u}(t) = C\hat{u}(t), \bar{f}(t) = Cf(t)$ and $\bar{r}(t) = C^{-T}r(t)$. Since matrix $\bar{J}$ is symmetric-negative-definite, the semi-discretization obtained by dropping the remainder $\mathscr{O}(h^4)$ in (2.11), and then in (2.7), is stable. Consequently, using $u_i(t)$ to denote the semi-discrete approximation of $u(x_i,t)$, we obtain the following quasi-compact finite difference scheme for the space discretization of SFDEs (1.1):

$$Ku'(t) = Ju(t) + Kf(t) + r(t),\ \ t \in [t_0, T], \tag{2.12}$$

where $u(t) = (u_1(t), u_2(t), \ldots, u_{M-1}(t))^T$.

BVMs have been proved to be very effective for solving a number of differential and integro-differential equations (see e.g. [6–12,25,26,29,30,42–48]). In view of this, in the following, we consider using BVMs to solve (2.12). Let $t_0 < t_1 < \cdots < t_N = T$ be a uniform mesh with $t_n = t_0 + n\tau$ $(0 \leq n \leq N)$ and $\tau = \frac{T-t_0}{N}$. Applying a k-step BVM with k_1 initial conditions and k_2 $(= k - k_1)$ final conditions to (2.12) yields the following scheme:

$$\sum_{i=-k_1}^{k_2} \alpha_{i+k_1} K u_{n+i} = \tau \sum_{i=-k_1}^{k_2} \beta_{i+k_1} (Ju_{n+i} + Kf_{n+i} + r_{n+i}),\quad n = k_1, \ldots, N-k_2, \tag{2.13}$$

$$\sum_{i=0}^{k} \alpha_i^{(j)} K u_i = \tau \sum_{i=0}^{k} \beta_i^{(j)} (Ju_i + Kf_i + r_i),\quad j = 1, \ldots, k_1 - 1, \tag{2.14}$$

$$\sum_{i=0}^{k}\alpha_{k-i}^{(j)}Ku_{N-i}=\tau\sum_{i=0}^{k}\beta_{k-i}^{(j)}(Ju_{N-i}+Kf_{N-i}+r_{N-i}),\quad j=N-k_2+1,\ldots,N,\quad(2.15)$$

where $u_n\approx u(t_n)$, $f_n=f(t_n)$, $r_n=r(t_n)$, α_i, β_i, $\alpha_i^{(j)}$ and $\beta_i^{(j)}$ are some given real coefficients such that schemes (2.13)-(2.15) have the same local order. By introducing the following notations:

$$U=\left(u_1^T,u_2^T,\ldots,u_N^T\right)^T,\quad F=\left(f_1^T,f_2^T,\ldots,f_N^T\right)^T,\quad R=\left(r_1^T,r_2^T,\ldots,r_N^T\right)^T,$$

schemes (2.13)-(2.15) can be more compactly written in vector form as

$$(A\otimes K-\tau B\otimes J)U=\tau[(B\otimes K)F+(B\otimes I_{M-1})R]+\tau b_0\otimes(Ju_0+Kf_0+r_0)-a_0\otimes(Ku_0),\quad(2.16)$$

where $\otimes$ denotes the Kronecker product, I_{M-1} is the $(M-1)\times(M-1)$ identity matrix, a_0 and A are defined by matrix

$$A^e:=\left[a_0|A\right]=\left[\begin{array}{c|cccccc}
\alpha_0^{(1)} & \alpha_1^{(1)} & \cdots & \alpha_k^{(1)} & & & \\
\vdots & \vdots & \cdots & \vdots & & & \\
\alpha_0^{(k_1-1)} & \alpha_1^{(k_1-1)} & \cdots & \alpha_k^{(k_1-1)} & & & \\
\alpha_0 & \alpha_1 & \cdots & \alpha_k & & & \\
 & \alpha_0 & \alpha_1 & \cdots & \alpha_k & & \\
 & & \ddots & \ddots & \ddots & \ddots & \\
 & & & \alpha_0 & \alpha_1 & \cdots & \alpha_k \\
 & & & \alpha_0^{(N-k_2+1)} & \alpha_1^{(N-k_2+1)} & \cdots & \alpha_k^{(N-k_2+1)} \\
 & & & \vdots & \vdots & \cdots & \vdots \\
 & & & \alpha_0^{(N)} & \alpha_1^{(N)} & \cdots & \alpha_k^{(N)}
\end{array}\right]\in\mathbb{R}^{N\times(N+1)},$$

and b_0 and B are given by matrix $B^e:=\left[b_0|B\right]$, which is defined similarly with β_i (resp. $\beta_i^{(j)}$) instead of α_i (resp. $\alpha_i^{(j)}$). Since problem (2.12) can be transformed to a diagonal set of stiff ODEs by considering, as done above, at first the tranformation $z(t)=Cu(t)$, with the matrix C satisfying (2.8), and then a tranformation by a real orthogonal matrix (i.e., that diagonalizing $\bar{J}$ in (2.11)), according to the analysis in [7, Chapter 4] (see also [6]), any $A_{k_1k_2}$-stable BVM will be convergent to the solution of (2.12), as the time-step τ tends to 0.

As a result, a class of space-time discretization methods (2.16) for SFDEs (1.1) have been obtained. In what follows, this class of methods will be called *quasi-compact boundary value methods*.

3 Two acceleration techniques

As is clear, when either one of the grid numbers M and N is large, the discrete problem generated by a quasi-compact boundary value method (2.16) amounts to a large-size linear system, potentially requiring a large computational cost for its solution. Hence, it is mandatory to look for suitable acceleration techniques to efficiently implement

method (2.16). For this purpose, in the following we will apply two acceleration techniques, namely, the KPS iteration method (cf. [13–16]) and the GMRES method (cf. [37]) with KPS preconditioner, to methods (2.16).

Let

$$Q = A \otimes K - \tau B \otimes J,$$

$$g = \tau[(B \otimes K)F + (B \otimes I_{M-1})R] + \tau b_0 \otimes (Ju_0 + Kf_0 + r_0) - a_0 \otimes (Ku_0).$$

Then, a method (2.16) can be written as a standard linear system:

$$QU = g. \tag{3.1}$$

We observe that matrix Q admits the following two Kronecker product splittings, depending on a parameter $\theta > 0$:

$$Q = (A + \theta B) \otimes K - B \otimes (\tau J + \theta K), \quad Q = -B \otimes (\tau J - \theta K) + (A - \theta B) \otimes K.$$

Based on the above splittings of matrix Q, when an initial guess $U^{(0)}$ is given, we have the following *KPS iteration method*, defined for all $k \geq 0$:

$$\begin{cases} [(A + \theta B) \otimes K] U^{\left(k+\frac{1}{2}\right)} = [B \otimes (\tau J + \theta K)] U^{(k)} + g, \\ [-B \otimes (\tau J - \theta K)] U^{(k+1)} = -[(A - \theta B) \otimes K] U^{\left(k+\frac{1}{2}\right)} + g. \end{cases} \tag{3.2}$$

Eliminating $U^{\left(k+\frac{1}{2}\right)}$ in (3.2) then gives

$$U^{(k+1)} = G(\theta) U^{(k)} + C(\theta), \quad k = 0, 1, \ldots, \tag{3.3}$$

where

$$G(\theta) = \left[(A + \theta B)^{-1}(A - \theta B)\right] \otimes \left[(\tau J - \theta K)^{-1}(\tau J + \theta K)\right]$$

and

$$C(\theta) = -2\theta (A + \theta B)^{-1} \otimes (\tau J - \theta K)^{-1} g.$$

In fact, (3.3) can also be derived directly from the following splitting of matrix Q:

$$Q = P(\theta) - R(\theta) \tag{3.4}$$

with

$$P(\theta) = -\frac{1}{2\theta}(A + \theta B) \otimes (\tau J - \theta K) \text{ and } R(\theta) = -\frac{1}{2\theta}(A - \theta B) \otimes (\tau J + \theta K). \tag{3.5}$$

In view of this, we have that

$$G(\theta) = P(\theta)^{-1} R(\theta) = I_{N(M-1)} - P(\theta)^{-1} Q, \tag{3.6}$$

and, thus, the KPS iteration method can be written in an equivalent form:

$$U^{(k+1)} = U^{(k)} + P(\theta)^{-1} r^{(k)}, \text{ where } r^{(k)} = g - QU^{(k)}. \tag{3.7}$$

For the KPS iteration method, we have the following convergence result.

Theorem 1 *Assume that all eigenvalues of matrix $B^{-1}A$ have positive real part and parameter $\theta > 0$. Then the spectral radius $\rho(G(\theta))$ of iteration matrix $G(\theta)$ of method* (3.3) *can be bounded by*

$$\Psi(\theta) := \max_{\lambda \in \sigma(B^{-1}A)} \left| \frac{\lambda - \theta}{\lambda + \theta} \right| < 1, \tag{3.8}$$

where $\sigma(\cdot)$ denotes the spectrum of a given matrix.

Proof. By the well-known properties of the Kronecker product (see e.g. [24]), the eigenvalues of iteration matrix $G(\theta)$ have the following form

$$\nu = \left(\frac{\lambda - \theta}{\lambda + \theta} \right) \left(\frac{\mu + \theta}{\mu - \theta} \right), \quad \text{where } \lambda \in \sigma(B^{-1}A) \text{ and } \mu \in \sigma(\tau K^{-1} J).$$

It follows from Lemma 3 that $\Re(\mu) < 0$. In view of this and $\theta > 0$, one has that

$$\left| \frac{\mu + \theta}{\mu - \theta} \right| < 1.$$

This gives

$$\rho(G(\theta)) = \max_{\substack{\lambda \in \sigma(B^{-1}A) \\ \mu \in \sigma(\tau K^{-1} J)}} \left| \frac{\lambda - \theta}{\lambda + \theta} \right| \left| \frac{\mu + \theta}{\mu - \theta} \right| < \max_{\lambda \in \sigma(B^{-1}A)} \left| \frac{\lambda - \theta}{\lambda + \theta} \right|.$$

Also, since all eigenvalues of matrix $B^{-1}A$ have positive real part and $\theta > 0$, it is deduced that $\Psi(\theta) < 1$. This completes the proof. □

Theorem 1 gives a convergence criterion of the KPS iteration method for any initial guess $U^{(0)}$ and parameter $\theta > 0$. Naturally, it is to be asked whether there is an optimal choice of θ, which minimizes the spectral radius of the iteration matrix $G(\theta)$. If so, a KPS iteration method with the optimal convergence rate could be found. Though this problem is in general still open, the following result allows us to choose θ in order to minimize the upper bound $\Psi(\theta)$ defined in (3.8), instead of the spectral radius $\rho(G(\theta))$ itself.

Theorem 2 *Assume that λ^* is the eigenvalue of matrix $B^{-1}A$ with the maximum argument ϕ^* and all eigenvalues of $B^{-1}A$ have positive real part. Then, the optimal value of the parameter θ such that the upper bound $\Psi(\theta)$ of spectral radius $\rho(G(\theta))$ becomes minimum is given by*

$$\theta^* := \arg\min_{\theta} \left\{ \max_{\lambda \in \sigma(B^{-1}A)} \left| \frac{\lambda - \theta}{\lambda + \theta} \right| \right\} = |\lambda^*|, \tag{3.9}$$

and $\Psi(\theta^*) = \tan \frac{\phi^*}{2}$.

Proof. The proof is similar to that for Corollary 3.1 in [13]. □

Although, under suitable condition, the KPS iteration method (3.2) could speed up the convergence rate of the iterative solution of problem (2.16), we still need to solve a corresponding large-size linear system at each iteration. Hence, it is necessary to consider a further acceleration scheme to implement this method effectively. To this end, we consider solving (3.1) by using a preconditioned GMRES method (cf. [37]), with the *KPS preconditioner* $P(\theta)$ defined in (3.4)-(3.5). In other words, by (3.6) we at first transform (3.1) into the following linear system:

$$(I_{N(M-1)} - G(\theta))U \equiv P(\theta)^{-1}QU = P(\theta)^{-1}g, \tag{3.10}$$

and then we solve (3.10) by the GMRES method. In view of (3.6), when $\rho(G(\theta)) < 1$, all eigenvalues of $P(\theta)^{-1}Q$ are located in a circle centered at $(1,0)$ with radius smaller than one. Also, it is well-known that a clustered spectrum often translates in rapid convergence of the GMRES method (see e.g. [37]). In this way, the KPS iteration method takes full advantage from the use of the GMRES method.

4 A numerical illustration

In this section, we will show the computational advantages of quasi-compact boundary value methods with the KPS preconditioner, where the four BVMs: third-order GBDF (GBDF-3), fourth-order ETR_2 (ETR_2-4), fifth-order GAM (GAM-5) and sixth-order TOM (TOM-6) (cf. [4,6,7]) will be used as the time discretization methods, respectively. In the actual computational procedure, we will always choose the parameter θ as the optimal parameters θ^* defined in (3.9). For the various methods considered here and different values of the dimension N, some of the optimal parameters θ^* are listed in Table 1. Moreover, we shall use the stopping criterion for the iterative solution of (3.1):

$$\frac{\|g - QU^{(k)}\|_2}{\|g - QU^{(0)}\|_2} \leq 10^{-8}, \quad k \geq 1.$$

In order to give a numerical comparison, we will also consider the preconditioned GMRES method with Strang-type block circulant preconditioner S, which has the form (see e.g. [10,45]):

$$S = s(A) \otimes K - \tau s(B) \otimes J, \tag{4.1}$$

where

$$s(A)=\begin{bmatrix} \alpha_{k_1} & \cdots & \alpha_k & & & \alpha_0 & \cdots & \alpha_{k_1-1} \\ \vdots & \ddots & & \ddots & & & \ddots & \vdots \\ \alpha_0 & & \ddots & & \ddots & & & \alpha_0 \\ & \ddots & & \ddots & & \ddots & 0 & \\ & & \ddots & & \ddots & & \ddots & \\ & 0 & & \ddots & & \ddots & & \\ \alpha_k & & & \ddots & & \ddots & & \alpha_k \\ \vdots & \ddots & & & \ddots & & \ddots & \vdots \\ \alpha_{k_1+1} & \cdots & \alpha_k & & & \alpha_0 & \cdots & \alpha_{k_1} \end{bmatrix} \in \mathbb{R}^{N\times N},$$

with the α_i being the coefficients of the ρ polynomial of the main formula defining the considered BVM, and $s(B)$ similarly defined by replacing the α_i with the β_i. It is remarkable that $s(A)$ and $s(B)$ are circulant matrices and, thus, they can be diagonalized by the Fourier matrix F (cf. [19]), i.e.,

$$s(A)=F^*\Lambda_A F,\quad s(B)=F^*\Lambda_B F,$$

where Λ_A (resp. Λ_B) is a diagonal matrix whose diagonal entries are the eigenvalues of $s(A)$ (resp. $s(B)$). Hence, we have that

$$S^{-1}QU=(F^*\otimes I_{M-1})(\Lambda_A\otimes K-\tau\Lambda_B\otimes J)^{-1}(F\otimes I_{M-1})QU.$$

In this way, the fast Fourier transformation can be used to efficiently implement the Strang-type preconditioned GMRES method. Moreover, we shall consider a restarted version of the GMRES method after ℓ iteration, which is usual in the computational practice, which we shall denote GMRES(ℓ).

For convenience, we list several combinations of methods that we shall consider in the sequel:

- Method I: KPS iteration method (3.7);
- Method II: using $P(\theta^*)$ as a preconditioner for GMRES(20) method for solving (3.10);
- Method III: using S in (4.1) as a preconditioner for GMRES(20) method for solving (3.10);
- Method IV: using the unpreconditioned GMRES(20) method for solving (3.1).

Moreover, the linear subsystems in (3.7) and (3.10) will be solved by LU factorization.

By introducing function

$$\begin{aligned} f(x,t)=&-e^{-t}\left\{x^4(1-x)^4+0.6\left[\frac{24x^{4-\alpha}}{\Gamma(5-\alpha)}-\frac{480x^{5-\alpha}}{\Gamma(6-\alpha)}+\frac{4320x^{6-\alpha}}{\Gamma(7-\alpha)}-\frac{20160x^{7-\alpha}}{\Gamma(8-\alpha)}\right.\right.\\ &\left.+\frac{40320x^{8-\alpha}}{\Gamma(9-\alpha)}\right]+0.5\left[\frac{24(1-x)^{4-\alpha}}{\Gamma(5-\alpha)}-\frac{480(1-x)^{5-\alpha}}{\Gamma(6-\alpha)}+\frac{4320(1-x)^{6-\alpha}}{\Gamma(7-\alpha)}\right.\\ &\left.\left.-\frac{20160(1-x)^{7-\alpha}}{\Gamma(8-\alpha)}+\frac{40320(1-x)^{8-\alpha}}{\Gamma(9-\alpha)}\right]\right\}, \end{aligned}$$

Table 1 Optimal parameters θ^* and the corresponding spectral upper bounds $\Psi(\theta^*)$ for GBDF-3, ETR$_2$-4, GAM-5 and TOM-6.

	GBDF-3		ETR$_2$-4		GAM-5		TOM-6	
N	θ^*	$\Psi(\theta^*)$	θ^*	$\Psi(\theta^*)$	θ^*	$\Psi(\theta^*)$	θ^*	$\Psi(\theta^*)$
6	1.1075	0.7329	1.6585	0.4972	1.5716	0.6111	1.6556	0.7069
12	1.1196	0.7208	1.6663	0.4973	1.5520	0.6218	1.6740	0.7175
24	1.1215	0.7215	1.6663	0.4973	1.5514	0.6221	1.6747	0.7179
48	1.1215	0.7215	1.6663	0.4973	1.5514	0.6221	1.6747	0.7179
96	1.1187	0.7236	1.7096	0.5775	1.4792	0.7036	1.9017	0.7928

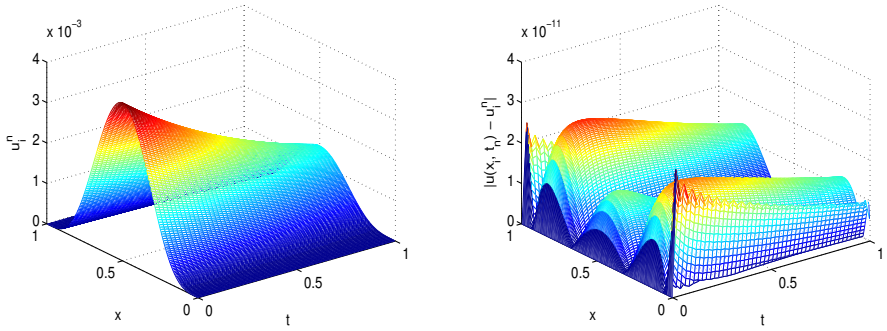
we consider the following initial-boundary problem of SFDE (cf. [27]):

$$\begin{cases} \dfrac{\partial u(x,t)}{\partial t} = 0.6\,{}_0D_x^{\alpha}u(x,t) + 0.5\,{}_xD_1^{\alpha}u(x,t) + f(x,t), \\ \qquad\qquad\qquad\qquad (x,t) \in (0,1)\times(0,1]; \\ u(x,0) = [x(1-x)]^4, \;\; x \in [0,1]; \quad u(0,t) = u(1,t) = 0, \;\; t \in [0,1], \end{cases} \tag{4.2}$$

whose exact solution $u(x,t) = e^{-t}[x(1-x)]^4$. For solving problem (4.2) with Methods I-II, in terms of Theorem 2, we first compute out the optimal parameters θ^* and the corresponding spectral upper bounds $\Psi(\theta^*)$ of GBDF-3, ETR$_2$-4, GAM-5 and TOM-6, which are listed in Table 1, where N denotes the number of the temporal partition.

Taking the numbers of temporal and spatial partitions: $N = 12\times 2^i$ $(i = 0,1,2)$, $M = 32\times 2^j$ $(j = 0,1,2,3)$, then applying Methods I-IV to the corresponding linear systems generated by the use of GBDF-3, ETR$_2$-4, GAM-5 and TOM-6 for the time discretization of problem (4.2), respectively, we can obtain a series of efficient numerical solutions. As an example, we apply Method II with $N = 48$, $M = 256$ and TOM-6 to problem (4.2) with $\alpha = 1.5$, whose numerical solutions u_i^n and their errors $|u(x_i,t_n) - u_i^n|$ $(0 \le i \le 256, 0 \le n \le 48)$ are plotted in Figure 1. Moreover, in Tables 2-3, we list the numbers of iterations (Iter) and CPU times (CPU) of Methods I-IV with TOM-6 for problem (4.2) with $\alpha = 1.5$. The results in Table 2 are derived from the tests whose initial guesses in the iterations are given by the initial condition $u(x,0) = [x(1-x)]^4$. Whereas, the results in Table 3 come from tests whose initial guesses in the iterations are given by the random initial guess $U^{(0)}$, which is a vector with normally distributed entries of mean zero and variance one. These numerical results verify Theorem 1, and show that Methods I-III converge faster than Method IV, Method II can reduce the iteration numbers and CPU times of Method I and Method II needs fewer CPU times than Method III although the latter requires less iterations than the former in some situations. According to these comparisons, we conclude that Method II is the best one among the considered methods. This is further confirmed by Figures 2 (a)-(b), where the spectrums of matrices Q and $P^{-1}(\theta^*)Q$ with TOM-6 for problem (4.2) with $N = 48, M = 256$ and $\alpha = 1.5$ are plotted, respectively. As is clear, the eigenvalues of preconditioned matrix $P^{-1}(\theta^*)Q$ of Method II closely cluster around the point (1,0), whereas those of Method IV are not so.

According to the above analysis, in the following, we shall always use Method II to solve the generated linear systems. In order to give an insight into the computa-



(a) (b)

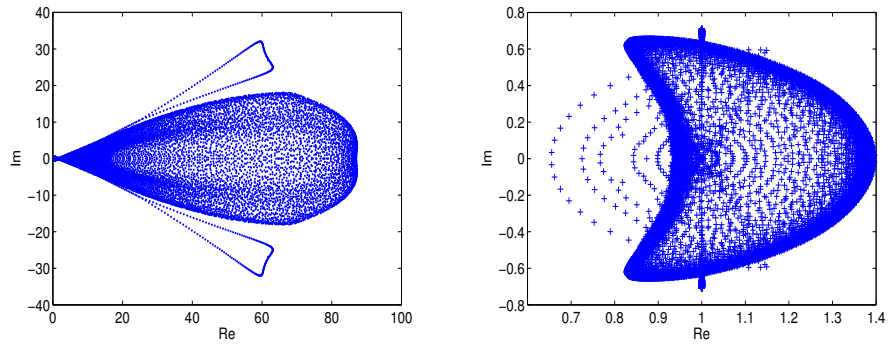
Fig. 1 (a) Numerical solutions derived by Method II with: $N = 48$, $M = 256$ and TOM-6 for problem (4.2) with $\alpha = 1.5$; (b) Errors of numerical solutions derived by Method II with: $N = 48$, $M = 256$ and TOM-6 for problem (4.2) with $\alpha = 1.5$.

Table 2 Iteration numbers and CPU times (in second) of Methods I-IV with TOM-6 for problem (4.2) with $\alpha = 1.5$, where initial guesses in the iterations are given by the initial condition $u(x,0) = [x(1-x)]^4$.

		Method I		Method II		Method III		Method IV	
N	M	Iter	CPU	Iter	CPU	Iter	CPU	Iter	CPU
12	32	38	0.0084	19	0.0076	18	0.0272	80	0.0420
	64	44	0.0166	19	0.0138	18	0.0588	147	0.1569
	128	46	0.0473	19	0.0347	23	0.1996	272	0.6515
	256	48	0.1620	19	0.1224	38	1.0787	501	4.9252
24	32	57	0.0172	28	0.0167	19	0.0504	85	0.0595
	64	57	0.0407	30	0.0387	20	0.1271	156	0.2279
	128	62	0.1299	30	0.1114	20	0.3580	284	1.5972
	256	64	0.4725	30	0.4132	22	1.4104	522	9.4939
48	32	101	0.0481	50	0.0480	17	0.0897	98	0.1090
	64	101	0.1260	52	0.1257	19	0.2422	179	0.6570
	128	101	0.4277	53	0.4085	19	0.6779	328	3.7880
	256	101	1.5842	53	1.5206	20	2.5774	603	21.9218

Table 3 Iterations numbers and CPU times (in second) of Methods I-IV with TOM-6 for problem (4.2) with $\alpha = 1.5$, where initial guesses in the iterations are given by the random initial guess $U^{(0)}$.

		Method I		Method II		Method III		Method IV	
N	M	Iter	CPU	Iter	CPU	Iter	CPU	Iter	CPU
12	32	55	0.0081	23	0.0073	24	0.0345	84	0.0487
	64	63	0.0185	22	0.0137	22	0.0718	151	0.1590
	128	65	0.0547	22	0.0377	27	0.2320	259	0.5915
	256	67	0.1837	21	0.1219	43	1.4258	462	4.2261
24	32	57	0.0175	33	0.0171	30	0.0755	90	0.0614
	64	73	0.0452	35	0.0384	30	0.1790	160	0.2385
	128	81	0.1356	35	0.1157	26	0.4553	283	1.5727
	256	85	0.4887	34	0.4123	27	1.9367	503	8.8423
48	32	101	0.0484	53	0.0472	27	0.1307	102	0.1278
	64	101	0.1264	57	0.1259	33	0.3850	185	0.7069
	128	109	0.4213	59	0.4144	34	1.1393	334	3.9042
	256	118	1.5839	59	1.5324	30	3.8278	598	21.2995



(a) (b)

Fig. 2 (a) The spectrum of matrix Q with TOM-6 for problem (4.2) with $N = 48, M = 256$ and $\alpha = 1.5$; (b) The spectrum of matrix $P^{-1}(\theta^*)Q$ with TOM-6 for problem (4.2) with $N = 48, M = 256$ and $\alpha = 1.5$.

tional accuracy of the methods, we introduce the notations:

$$E(h,\tau) = \max_{0\le n\le N}\ \max_{1\le i\le M-1} |u(x_i,t_n) - u_i^n|, \quad h = 1/N,\ \tau = 1/M,$$

$$\text{Order}_1 = \log_2\left[\frac{E(h,\tau)}{E(h/2,\tau)}\right], \quad \text{Order}_2 = \log_2\left[\frac{E(h,\tau)}{E(h,\tau/2)}\right]$$

to denote the global error, and temporal and spatial convergence orders, respectively. For testing the spatial convergence orders of the methods, we take a sufficiently small time-stepsize $\tau = \frac{1}{200}$ and the space-stepsizes $h = \frac{1}{5\times 2^j}$ $(j = 0,1,2,3)$ and then apply Method II to solve the linear systems (3.1) derived respectively by GBDF-3, ETR_2-4, GAM-5, and TOM-6 for problem (4.2) ($\alpha = 1.1, 1.5, 1.9$). The obtained global errors and spatial convergence orders are listed in Table 4. We can find from Table 4 that the used methods are quite effective and their spatial convergence orders are about 4, which is consistent to the local order of the methods in space. Next, we continue to test the temporal convergence orders of the methods. Taking the time-stepsize $\tau = \frac{1}{5\times 2^i}$ $(i = 0,1,2,3)$ and the corresponding spatial-stepsize $h = 1/\lfloor (1/\tau)^{\frac{p}{4}} \rfloor$ (where p is the local order of the used BVM), and then applying Method II to the linear systems (3.1) generated respectively by GBDF-3, ETR_2-4, GAM-5 and TOM-6 for problem (4.2) ($\alpha = 1.1, 1.5, 1.9$), we can numerically estimate the temporal convergence orders of the methods, which are shown in Table 5. As one may infer from the listed results, they are in good agreement with the local orders of the used BVMs.

Table 4 Global errors and spatial convergence orders of GBDF-3, ETR_2-4, GAM-5 and TOM-6 to problem (4.2) with time stepsize $\tau = 1/200$.

		GBDF-3		ETR_2-4		GAM-5		TOM-6	
α	h	$E(h,\tau)$	Order_1	$E(h,\tau)$	Order_1	$E(h,\tau)$	Order_1	$E(h,\tau)$	Order_1
1.1	1/5	4.3639e–5	–	4.3639e–5	–	4.3639e–5	–	4.3639e–5	–
	1/10	4.5567e–6	3.2596	4.5567e–6	3.2596	4.5567e–6	3.2596	4.5567e–6	3.2596
	1/20	3.3244e–7	3.7768	3.3244e–7	3.7768	3.3244e–7	3.7768	3.3244e–7	3.7768
	1/40	3.1655e–8	3.3926	3.1655e–8	3.3926	3.1655e–8	3.3926	3.1655e–8	3.3926
1.5	1/5	9.8875e–5	–	9.8874e–5	–	9.8874e–5	–	9.8874e–5	–
	1/10	8.4639e–6	3.5462	8.4638e–6	3.5462	8.4638e–6	3.5462	8.4638e–6	3.5462
	1/20	6.6453e–7	3.6709	6.6453e–7	3.6709	6.6453e–7	3.6709	6.6453e–7	3.6709
	1/40	4.4224e–8	3.9095	4.4222e–8	3.9095	4.4222e–8	3.9095	4.4222e–8	3.9095
1.9	1/5	1.5014e–4	–	1.5014e–4	–	1.5014e–4	–	1.5014e–4	–
	1/10	1.1488e–5	3.7081	1.1488e–5	3.7081	1.1488e–5	3.7081	1.1488e–5	3.7081
	1/20	8.0714e–7	3.8312	8.0714e–7	3.8312	8.0714e–7	3.8312	8.0714e–7	3.8312
	1/40	5.2118e–8	3.9530	5.2117e–8	3.9530	5.2117e–8	3.9530	5.2117e–8	3.9530

Table 5 Global errors and temporal convergence orders of GBDF-3, ETR_2-4, GAM-5 and TOM-6 to problem (4.2) with spatial stepsizes $h = 1/\lfloor (1/\tau)^{\frac{p}{4}} \rfloor$.

		GBDF-3		ETR_2-4		GAM-5		TOM-6	
α	τ	$E(h,\tau)$	Order_2	$E(h,\tau)$	Order_2	$E(h,\tau)$	Order_2	$E(h,\tau)$	Order_2
1.1	1/5	9.7096e–5	–	4.3636e–5	–	1.0499e–5	–	2.1151e–6	–
	1/10	2.3675e–5	2.0361	4.5458e–6	3.2629	4.5770e–7	4.5197	6.9787e–8	4.9216
	1/20	4.5575e–6	2.3770	3.3227e–7	3.7741	2.4313e–8	4.2346	1.5411e–9	5.5010
	1/40	7.0091e–7	2.7009	3.1655e–8	3.3919	9.9174e–10	4.6156	2.7836e–11	5.7908
1.5	1/5	1.6574e–4	–	1.0976e–4	–	2.1496e–5	–	4.9788e–6	–
	1/10	5.6123e–5	1.5623	8.3262e–6	3.7206	9.8191e–7	4.4524	1.0583e–7	5.5559
	1/20	8.4889e–6	2.7249	6.6219e–7	3.6523	3.3383e–8	4.8784	1.7477e–9	5.9202
	1/40	1.5250e–6	2.4768	4.4192e–8	3.9054	1.1028e–9	4.9199	3.2975e–11	5.7279
1.9	1/5	2.4642e–4	–	1.8205e–4	–	3.1362e–5	–	6.6881e–6	–
	1/10	8.0785e–5	1.6089	1.2743e–5	3.8366	1.3680e–6	4.5189	1.3882e–7	5.5904
	1/20	1.1501e–5	2.8123	8.3652e–7	3.9292	3.9705e–8	5.1066	2.0793e–9	6.0610
	1/40	1.9141e–6	2.5871	5.2135e–8	4.0041	1.2929e–9	4.9407	3.2439e–11	6.0022

References

1. Amodio, P., Brugnano, L.: The Conditioning of Toeplitz Banded Matrices. Math. Comput. Modelling **23**, 29–42 (1996)
2. Bai, J., Feng, X.: Fractional-order anisotropic diffusion for image denoising. IEEE Trans. Image Process. **16**, 2492–2502 (2007)
3. Benson, D.A., Wheatcraft, S.W., Meerschaert, M.M.: The fractional-order governing equation of Lévy motion. Water Resour. Res. **36**, 1413–1423 (2000)
4. Brugnano, L.: Essentially symplectic boundary value methods for linear Hamiltonian systems. J. Comput. Math. **15**, 233–252 (1997)
5. Brugnano, L., Trigiante, D.: Tridiagonal Matrices: Invertibility and Conditioning. Linear Algebra Appl. **166**, 131–150 (1992)
6. Brugnano, L., Trigiante, D.: Convergence and stability of boundary value methods for ordinary differential equations. J. Comput. Appl. Math. **66**, 97–109 (1996)
7. Brugnano, L., Trigiante, D.: Solving Differential Equations by Multistep Initial and Boundary Value Methods. Gordan and Breach, Amsterdam (1998)
8. Brugnano, L., Trigiante, D.: Boundary value methods: the third way between linear multistep and Runge-Kutta methods. Comput. Math. Appl. **36**, 269–284 (1998)

9. Brugnano, L., Zhang, C., Li, D.: A class of energy-conserving Hamiltonian boundary value methods for nonlinear Schrödinger equation with wave operator. Comm. Nonlinear Sci. Numer. Simmu. **60**, 33–49 (2018)
10. Chan, R.H., Ng, M.K., Jin, X.: Strang-type preconditioners for systems of LMF-based ODE codes. IMA J. Numer. Anal. **21**, 451–462 (2001)
11. Chen, H., Zhang, C.: Boundary value methods for Volterra integral and integro-differential equations. Appl. Math. Comput. **218**, 2619–2630 (2011)
12. Chen, H., Zhang, C.: Convergence and stability of extended block boundary value methods for Volterra delay integro-differential equations. Appl. Numer. Math. **62**, 141–154 (2012)
13. Chen, H.: A splitting preconditioner for the iterative solution of implicit Runge-Kutta and boundary value methods. BIT **54**, 607–621 (2014)
14. Chen, H.: Kronecker product splitting preconditioners for implicit Runge-Kutta discretizations of viscous wave equations. Appl. Math. Model. **40**, 4429–4440 (2016)
15. Chen, H.: A splitting preconditioner for implicit Runge-Kutta discretizations of a partial differential-algebraic equation. Numer. Algor. **73**, 1037–1054 (2016)
16. Chen, H., Zhang, T., Lv, W.: Block preconditioning strategies for time-space fractional diffusion equations. Appl. Math. Comput. **337**, 41–53 (2018)
17. Chen, M., Deng, W.: Fourth order accurate scheme for the space fractional diffusion equations. SIAM J. Numer. Anal. **52**, 1418–1438 (2014)
18. Chen, M., Deng, W.: Fourth order difference approximations for space Riemann-Liouville derivatives based on weighted and shifted Lubich difference operators. Comm. Comput. Phys. **16**, 516–540 (2014)
19. Davis, P.: Circulant Matrices. AMS Chelsea Publishing, Rhode Island (1994)
20. Gal, N., Weihs, D.: Experimental evidence of strong anomalous diffusion in living cells. Phys. Rev. E **81**, 020903 (2010)
21. Ganti, V., Meerschaert, M.M., Georgiou, E.F., Viparelli, E., Parker, G.: Normal and anomalous diffusion of gravel tracer particles in rivers. J. Geophys. Res. **115**, F00A07 (2010)
22. Gu, X., Huang, T., Zhao, X., Li, H., Li, L.: Strang-type preconditioners for solving fractional diffusion equations by boundary value methods. J. Comput. Appl. Math. **277**, 73–86 (2015)
23. Hao, Z., Sun, Z., Cao, W.: A fourth-order approximation of fractional derivatives with its applications. J. Comput. Phys. **281**, 787–805 (2015)
24. Horn, R.A., Johnson, C.R.: Topics in Matrix Analysis. Cambridge University Press, Cambridge (1991)
25. Iavernaro, F., Mazzia, F.: Convergence and stability of multistep methods solving nonlinear initial value problems. SIAM J. Sci. Comput. **18**, 270–285 (1997)
26. Iavernaro, F., Mazzia, F.: Block-boundary value methods for the solution of ordinary differential equations. SIAM J. Sci. Comput. **21**, 323–339 (1999).
27. Lei, S., Huang, Y.: Fast algorithms for high-order numerical methods for space fractional diffusion equations. Int. J. Comput. Math. **94**, 1062–1078 (2017)
28. Lei, S., Sun, H.: A circulant preconditioner for fractional diffusion equations. J. Comput. Phys. **242**, 715–725 (2013)
29. Li, C., Zhang, C.: Block boundary value methods applied to functional differential equations with piecewise continuous arguments. Appl. Numer. Math. **115**, 214–224 (2017)
30. Li, C., Zhang, C.: The extended generalized Störmer-Cowell methods for second-order delay boundary value problems. Appl. Math. Comput. **294**, 87–95 (2017)
31. Lubich, C.: Discretized fractional calculus. SIAM J. Math. Anal. **17**, 704–719 (1986)
32. Meerschaert, M.M., Tadjeran, C.: Finite difference approximations for fractional advection-dispersion flow equations. J. Comput. Appl. Math. **172**, 65–77 (2004)
33. Oldham, K.B.: Fractional differential equations in electrochemistry. Adv. Engrg. Software **41**, 9–12 (2010)
34. Pang, H., Sun, H.: Multigrid method for fractional diffusion equations. J. Comput. Phys. **231**, 693–703 (2012)
35. Podlubny, I.: Fractional Differential Equations. Academic Press, New York (1999)
36. Raberto, M., Scalas, E., Mainardi, F.: Waiting-times and returns in high-frequency financial data: an empirical study. Physica A **314**, 749–755 (2002)
37. Saad, Y., Schultz, M.H.: GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **7**, 856–869 (1986)
38. Samko, S.G., Kilbas, A.A., Marichev, O.I.: Fractional Integrals and Derivatives: Theory and Applications. Gordon and Breach, New York (1993)

39. Scher, H., Montroll, E.W.: Anomalous transit-time dispersion in amorphous solids. Phys. Rev. B **12**, 2455–2477 (1975)
40. Wang, H., Du, N.: A super fast-preconditioned iterative method for steady-state space-fractional diffusion equations. J. Comput. Phys. **240**, 49–57 (2013)
41. Wang, H., Wang, K., Sircar, T.: A direct $\mathcal{O}(N\log_2 N)$ finite difference method for fractional diffusion equations. J. Comput. Phys. **229**, 8095–8104 (2010)
42. Wang, H., Zhang, C., Zhou, Y.: A class of compact boundary value methods applied to semi-linear reaction-diffusion equations. Appl. Math. Comput. **325**, 69–81 (2018)
43. Zhang, C., Chen, H.: Block boundary value methods for delay differential equations. Appl. Numer. Math. **60**, 915–923 (2010)
44. Zhang, C., Chen, H.: Asymptotic stability of block boundary value methods for delay differential-algebraic equations. Math. Comput. Simul. **81**, 100–108 (2010)
45. Zhang, C., Chen, H., Wang, L.: Strang-type preconditioners applied to ordinary and neutral differential-algebraic equations. Numer. Linear Alge. Appl. **18**, 843–855 (2011)
46. Zhang, C., Li, C.: Generalized Störmer-Cowell methods for nonlinear BVPs of second-order delay-integro-differential equations. J. Sci. Comput. **74**, 1221–1240 (2018)
47. Zhou, Y., Zhang, C., Wang, H.: The extended boundary value methods for a class of fractional differential equations with Caputo derivatives, submitted.
48. Zhou, Y., Zhang, C.: Convergence and stability of block boundary value methods applied to nonlinear fractional differential equations with Caputo derivatives. Appl. Numer. Math. **135**, 367–380 (2019)