Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 3, No. 4, 2015, pp. 258-273

Solving large systems arising from fractional models by preconditioned methods

Reza Khoshsiar Ghaziani*
Faculty of Mathematical Sciences,
Shahrekord University,
P. O. Box 115, Shahrekord, Iran.
E-mail: Khoshsiar@sci.sku.ac.ir

Mojtaba Fardi
Faculty of Mathematical Sciences,
Shahrekord University,
P. O. Box 115, Shahrekord, Iran.
E-mail: fardimojtaba@yahoo.com

Mehdi Ghasemi
Faculty of Mathematical Sciences,
Shahrekord University,
P. O. Box 115, Shahrekord, Iran.
E-mail: meh.ghasemi@yahoo.com

Abstract This study develops and analyzes preconditioned Krylov subspace methods for solving discretization of the time-independent space-fractional models. First we apply a shifted Grnwald formulas to obtain a stable finite difference approximation to fractional advection-diffusion equations. Then, we apply two preconditioned iterative methods, namely, the preconditioned generalized minimal residual (preconditioned GMRES) method and the preconditioned conjugate gradient for normal residual(preconditioned CGN) method, to solve the corresponding discritized systems. We make comparisons between the preconditioners commonly used in the parallelization of the preconditioned Krylov subspace methods. The results suggest that preconditioning technique is a promising candidate for solving large-scale linear systems arising from fractional models.

Keywords. Krylov subspace methods, Preconditioning techniques, Fractional model.
2010 Mathematics Subject Classification. 65L05, 34K06, 34K28.

1. Introduction

Recent studies show that fractional advection-diffusion equations provide more adequate and accurate description of the movement of solute in an aquifer than the traditional second-order advectiondiffusion equations do [2, 3]. Fractional advection-diffusion equations are closely related to a continuous time random walk approach, which allows descriptions of particle motions with long-range correlations and is appropriate for the description of subsurface solute transport. The most significant

Received: 1 October 2015 ; Accepted: 7 November 2016.
∗ Corresponding author.

difference of fractional advection-diffusion equations from their integer analogue is that they generate numerical approximations with full coefficient matrices, which often cause extra numerical difficulties. This is in addition to the common numerical difficulties with their integer analogue. To put the present work in context, we begin by discussing some of the key numerical methods that have been proposed to solve various fractional partial differential equations (FPDEs). In recent years much progress has been made on the development of numerical methods for fractional advection-diffusion equations. For instance, in [10, 9] a shifted Grnwald formulas was used to obtain a stable finite difference approximation to fractional advection-diffusion equations. Numerical solution for boundary value problem of fractional order has been addressed in [12]. A method for solving space-time fractional differential equations is introduced in [5].

However, the most significant obstacle in the numerical methods for fractional partial differential equations is that these methods generate discrete systems with full coefficient matrices. Consequently $O(N^3)$ account of computations and $O(N^2)$ account of storage are required to solve a problem of size N.

Meerschaert and Tadjeran [9], showed that discretisation of the fractional derivatives using standard (non-shifted) Grnwald formulas led to unstable methods when the fractional order. To overcome this, they proposed a method utilising with shifted Grnwald formulas, which they showed to be stable, and first order accurate in space.

In more recent times, a number of authors have addressed the issue of high computational expense associated with the solution of FPDEs. Several different approaches have been explored, with many papers employing a mixture of these approaches in various fascinating ways. Krylov subspace methods have been a popular approach, owing to their ability to solve linear systems and compute matrix functions without the need to operate directly on dense matrices. Yang et al. [21, 22, 23] and Burrage et al. [4] used Krylov subspace methods for computing matrix functions to solve fractional Laplacian equations. Moroney and Yang [11] and Wang and Wang [20] used Krylov subspace methods to solve the two-sided space-fractional diffusion equation in one dimension, with the former authors considering nonlinear problems and the latter authors considering linear problems with an advection term.

Preconditioning has been a common theme in many of these papers, since it is well known that Krylov subspace methods generally require an effective preconditioner in order to perform satisfactorily. Yang et al. [21, 22, 23] developed preconditioners based on eigenvalue deflation. Burrage et al. [4] considered both algebraic multigrid and incomplete LU preconditioning. Moroney and Yang [11] developed a banded preconditioner. A numerical treatment of sparse indefinite systems of linear equations is given in [13].

In this paper we develop and apply preconditioned methods based on Krylov subspaces on linear systems arising from discretized time-independent space-fractional convection-diffusion to obtain the accurate and efficient solution of fractional advection-equations.

The rest of this paper is organized as follows. In Section 2 we present the fractional advection-diffusion equations and apply the implicit finite difference scheme to obtain corresponding linear systems. In Section 3 we briefly describe the two preconditioned

methods, namely, GMRES and CGN for solving large-scale linear systems arising from the fractional models. In Section 4 we carry out numerical experiment to investigate the performance of the developed algorithm. Finally, we draw our conclusions in Section 5.

2. The finite difference scheme: Structure of the coefficient matrix

In this section, the implicit finite difference scheme is applied to discretize the time-independent space-fractional convection-diffusion model and the fractional advection-dispersion model, which are obtained from the standard diffusion models by replacing the first order time derivative with a fractional derivative and can be written as the following forms

The first model: The time-independent space-fractional convection-diffusion model

$$\mathbf{P(D)}v(x,t) = k(x)\frac{\partial^{\vartheta_1} v(x,t)}{\partial x^{\vartheta_1}} + h(x)v(x,t) + z_1(x,t), \\ 0 < x < x_{U_1},\ 0 < t \le T_1,\ 1 < \vartheta_1 \le 2, \tag{2.1}$$

with the boundary conditions

$$v(0,t) = 0,\ v(x_{U_1},t) = 0,\ 0 < t \le T_1, \tag{2.2}$$

and the initial condition

$$v(x,0) = \phi(x),\ 0 \le x \le x_{U_1}, \tag{2.3}$$

where

$$\mathbf{P(D)} = \frac{\partial^{\omega_1} v}{\partial t^{\omega_1}},\ 0 < \omega_1 \le 1. \tag{2.4}$$

Here $k(x) \ge 0$ and $h(x) \le 0$ are continuous functions on $[0, x_{U_1}]$, $z_1(x,t)$ is a continuous function on $[0, x_{U_1}] \times [0, T_1]$.

The second model: The fractional advection-dispersion model

$$\mathbf{Q(D)}v(x,t) = \alpha_1 \frac{\partial^{\vartheta_2} v}{\partial x^{\vartheta_2}} + z_2(x,t),\ 0 < x < x_{U_2},\ 0 < t \le T_2,\ 1 < \vartheta_2 \le 2, \tag{2.5}$$

with the boundary conditions

$$v(0,t) = 0,\ v(x_{U_2},t) = 0,\ 0 < t \le T_2, \tag{2.6}$$

and the initial condition

$$v(x,0) = \phi(x),\ 0 \le x \le x_{U_2}, \tag{2.7}$$

where

$$\mathbf{Q(D)} = \frac{\partial v(x,t)}{\partial t} + \frac{\partial^{\omega_2} v(x,t)}{\partial t^{\omega_2}},\ 0 < \omega_2 \le 1. \tag{2.8}$$

Here $\alpha_1 \ge 0$ is a constant and $z_2(x,t)$ is a continuous function on $[0, x_{U_2}] \times [0, T_2]$. To illustrate the discretization process, denote $x_j = j\delta_x$, $t_n = n\delta_t$, $\Delta_{\delta_x} = \{x_j | 0 \le j \le J\}$, $\Delta_{\delta_t} = \{t_n | 0 \le n \le N\}$, $\Delta_{\delta_x}^{\delta_t} = \Delta_{\delta_x} \times \Delta_{\delta_t}$, where $\delta_x = \frac{x_{U_i}}{J}$, $i = 1, 2$, $\delta_t = \frac{T_i}{N}$, $i = 1, 2$, are uniform spacial and temporal mesh sizes respectively, and J, N are two positive integers. Suppose that $V = \{V_j^{(n)} = v(x_j, t_n) | 0 \le j \le J,\ 0 \le n \le$

$N\}$, $\partial_x^{\vartheta_i} V = \{\partial_x^{\vartheta_i} V_j^{(n)} = \frac{\partial^{\vartheta_i} v(x,t)}{\partial x^{\vartheta_i}}|_{(x,t)=(x_j,t_n)} | 0 \le j \le J,\ 0 \le n \le N, i = 1,2\}$ and $\partial_t^{\omega_i} V = \{\partial_t^{\omega_i} V_j^{(n)} = \frac{\partial^{\omega_i} v(x,t)}{\partial t^{\omega_i}}|_{(x,t)=(x_j,t_n)} | 0 \le j \le J,\ 0 \le n \le N,\ i = 1,2\}$ are three gird functions on $\Delta_{\delta_x}^{\delta_t}$.

We use the first order shifted Grünwald formula to descretize the Riemann-Liouville fractional order derivative $\frac{\partial^{\vartheta_i} v(x,t)}{\partial x^{\vartheta_i}},\ i = 1,2$ as the following (see Ref. [14])

$$\frac{\partial^{\vartheta_i} v(x_j,t)}{\partial x^{\vartheta_i}} = \frac{1}{\delta_x^{\vartheta_i}} \sum_{k=0}^{j+1} \mu_k^{(\vartheta_i)} v(x_{j+1-k},t) + O(\delta_x),\ i = 1,2, \tag{2.9}$$

where the coefficients $\mu_k^{(\vartheta_i)}$ can be evaluated recursively (see Ref. [14, 19])

$$\begin{cases} \mu_0^{(\vartheta_i)} = 1, \\ \mu_k^{(\vartheta_i)} = (1 - \frac{\vartheta_i + 1}{k}) \mu_{k-1}^{(\vartheta_i)},\ k > 0. \end{cases} \tag{2.10}$$

According to (2.10), the $\mu_k^{(\vartheta_i)}$ for $1 < \vartheta_i \le 2$, satisfy the following properties (see Ref. [14, 19])

$$\begin{cases} \mu_0^{(\vartheta_i)} = 1,\ \mu_1^{(\vartheta_i)} = -\vartheta_i < 0,\ \mu_2^{(\vartheta_i)} > \mu_3^{(\vartheta_i)} > ... > 0, \\ \sum_{k=0}^{\infty} \mu_k^{(\vartheta_i)} = 0,\ \sum_{k=0}^{m} \mu_k^{(\vartheta_i)} \le 0 \ for\ m \ge 1, \\ \mu_k^{(\vartheta_i)} = O(k^{-(\vartheta_i+1)}). \end{cases} \tag{2.11}$$

According to [6], we consider the following time difference formula to discretize the time-fractional derivative

$$\frac{\partial^{\omega_i} v(x,t_{n+1})}{\partial t^{\omega_i}} = \frac{\delta_t^{-\omega_i}}{\Gamma(2-\omega_i)} \sum_{k=0}^{n} a_k^{(\omega_i)} (v(x,t_{n+1-k}) - v(x,t_{n-k})) + r_{i,\delta_t}^{(n+1)},$$
$$i = 1,2 \tag{2.12}$$

where

$$a_k^{(\omega_i)} = (k+1)^{(1-\omega_i)} - k^{(1-\omega_i)},\ k \ge 0, \tag{2.13}$$

and the truncation error $r_{i,\delta_t}^{(n+1)}$ satisfies

$$r_{i,\delta_t}^{(n+1)} \le c_v \delta_t^{2-\omega_i}, \tag{2.14}$$

where c_v is a constant depending only on v.

2.1. **Formulation of the finite difference scheme for the first model.** Let $Z_{1,j}^{(n)} = z_1(x_j,t_n)$, $k_j = k(x_j)$ and $h_j = h(x_j)$. By applying (2.9) and (2.12), and neglecting the small terms, the two-dimensional finite difference scheme for the problem (2.1) can be formulated as follows

$$\frac{\delta_t^{-\omega_1}}{\Gamma(2-\omega_1)} \sum_{i=0}^{n} a_i^{(\omega_1)} (V_j^{(n+1-i)} - V_j^{(n-i)}) = k_j \delta_x^{-\vartheta_1} \sum_{k=0}^{j+1} \mu_k^{(\vartheta_1)} V_{j-k+1}^{(n+1)} + h_j V_j^{(n+1)} + Z_{1,j}^{(n+1)} \tag{2.15}$$

We can rewrite (2.15) in following form

$$\begin{aligned}
&-k_j\delta_x^{-\vartheta_1}\gamma_{\delta_t}^{(1)}\mu_2^{(\vartheta_1)}V_{j-1}^{(n+1)}+(-k_j\delta_x^{-\vartheta_1}\gamma_{\delta_t}^{(1)}\mu_1^{(\vartheta_1)}-h_j\gamma_{\delta_t}^{(1)}+1)V_j^{(n+1)}-k_j\delta_x^{-\vartheta_1}\gamma_{\delta_t}^{(1)}V_{j+1}^{(n+1)}\\
&-k_j\delta_x^{-\vartheta_1}\gamma_{\delta_t}^{(1)}\textstyle\sum_{k=3}^{j+1}\mu_k^{(\vartheta_1)}V_{j-k+1}^{(n+1)}=-\sum_{i=1}^{n}a_i^{(\omega_1)}(V_j^{(n+1-i)}-V_j^{(n-i)})\\
&+V_j^{(n)}+\gamma_{\delta_t}^{(1)}Z_{1,j}^{(n+1)},\quad 1\le j\le J-1,\ 0\le n\le N-1,
\end{aligned}\tag{2.16}$$

where $\gamma_{\delta_t}^{(1)}=\Gamma(2-\omega_1)\delta_t^{\omega_1}$.
In addition, from (2.2) and (2.3) we have

$$\begin{cases} V_0^{(n)}=0,\ V_J^{(n)}=0,\ 1\le n\le N,\\ V_i^{(0)}=\phi(x_j),\ 0\le j\le J. \end{cases}\tag{2.17}$$

To develop an efficient solution technique, we express the above scheme into a matrix form. Let $\mathbf{V}^{(n)}=[V_1^{(n)},V_2^{(n)},...,V_{J-1}^{(n)}]^T$, $b^{(n)}=[b_1^{(n)},b_2^{(n)},...,b_{J-1}^{(n)}]^T$, $\mathbf{A}^{(n)}=[A_{ij}^{(n)}]_{i,j=1}^{J-1}$, $B^{(n)}=[B_{ij}^{(n)}]_{i,j=1}^{J-1}$ $\mathbf{Z}_1^{(n)}=[Z_{1,1}^{(n)},Z_{1,2}^{(n)},...,Z_{1,J-1}^{(n)}]^T$ and $\mathbf{I}_{J-1\times J-1}$ be the identity matrix with an appropriate size, then the finite difference scheme (2.16) can be written in the following matrix form

$$\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)}=(\mathbf{I}-\gamma_{\delta_t}^{(1)}\mathbf{A}^{(n+1)})\mathbf{V}^{(n+1)}=\mathbf{b}^{(n+1)}.\tag{2.18}$$

Here the entries of matrix $\mathbf{A}^{(n+1)}$ and vector $\mathbf{b}^{(n+1)}$ are given by

$$A_{ij}^{(n+1)}=\begin{cases} k_i\mu_{i-j+1}^{(\vartheta_1)}\delta_x^{-\vartheta_1},\ i\ge j+2,\\ k_i\mu_2^{(\vartheta_1)}\delta_x^{-\vartheta_1},\ i=j+1,\\ k_i\mu_1^{(\vartheta_1)}\delta_x^{-\vartheta_1}+h_i,\ i=j,\\ k_i\delta_x^{-\vartheta_1},\ i=j-1,\\ 0,\ i<j-1, \end{cases}\tag{2.19}$$

and

$$\mathbf{b}^{(n+1)}=-\sum_{i=1}^{n}a_i^{(\omega_1)}(\mathbf{V}^{(n+1-i)}-\mathbf{V}^{(n-i)})+\gamma_{\delta_t}^{(1)}\mathbf{V}^{(n)}+\gamma_{\delta_t}^{(1)}\mathbf{Z}_1^{(n+1)},\quad n=0,1,2,...N-1.\tag{2.20}$$

2.2. **Formulation of the finite difference scheme for the second model.** Now, we consider the finite difference scheme for problem (2.5) with the conditions (2.6)-(2.7). As usual, the first order temporal derivative can be approximated by the backward difference scheme

$$\frac{\partial v(x,t_{n+1})}{\partial t}=\frac{v(x,t_{n+1})-v(x,t_n)}{\delta t}+O(\delta t).\tag{2.21}$$

By applying (2.9), (2.11) and (2.21), and neglecting the truncation error of implicit finite difference scheme, the problem (2.5) can be formulated as follows

$$\begin{aligned}
&-\alpha_1\delta_x^{-\vartheta_2}\gamma_{\delta_t}^{(2)}\mu_2^{(\vartheta_2)}V_{j-1}^{(n+1)}+(\gamma_{\delta_t}^{(2)}+1-\alpha_1\delta_x^{-\vartheta_2}\gamma_{\delta_t}^{(2)}\mu_1^{(\vartheta_2)})V_j^{(n+1)}-\alpha_1\delta_x^{-\vartheta_2}V_{j+1}^{(n+1)}\\
&-\alpha_1\delta_x^{-\vartheta_2}B_{\delta_t}^{(1)}\textstyle\sum_{k=3}^{j+1}\mu_k^{(\vartheta_2)}V_{j+1-k}^{(n+1)}=(1+\delta_t^{-1}\gamma_{\delta_t}^{(2)})V_j^{(n)}\\
&-\textstyle\sum_{k=1}^{n}a_k^{(\omega_2)}(V_j^{(n+1-k)}-V_j^{(n-k)})+\gamma_{\delta_t}^{(2)}Z_{2,j}^{(n+1)},
\end{aligned}\tag{2.22}$$

where $Z_{2,j}^{(n+1)}=z_2(x_j,t_{n+1})$ and $\gamma_{\delta_t}^{(2)}=\Gamma(2-\omega_2)\delta_t^{\omega_2}$.
The initial and boundary conditions are discretized as follows

$$\begin{cases} V_0^{(n)}=0,\ V_J^{(n)}=0,\ 1\le n\le N,\\ V_i^{(0)}=\phi(x_j),\ 0\le j\le J. \end{cases}\tag{2.23}$$

Suppose that $\mathbf{I}_{J-1\times J-1}$ is an identity matrix and $\mathbf{V}^{(n)}$, $\mathbf{b}^{(n)}$ and $\mathbf{Z}_2^{(n)}$ are vectors given by

$$\begin{cases} \mathbf{V}^{(n)}=[V_1^{(n)},V_2^{(n)},...,V_{J-1}^{(n)}]^T,\\ \mathbf{b}^{(n)}=[b_1^{(n)},b_2^{(n)},...,b_{J-1}^{(n)}]^T,\\ \mathbf{Z}_2^{(n)}=[Z_{2,1}^{(n)},Z_{2,2}^{(n)},...,Z_{2,J-1}^{(n)}]^T, \end{cases}\tag{2.24}$$

then (2.22) can be written in a matrix form

$$\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)}=(\mathbf{I}+\gamma_{\delta_t}^{(2)}\mathbf{A}^{(n+1)})\mathbf{V}^{(n+1)}=\mathbf{b}^{(n+1)},\tag{2.25}$$

where the entries of matrix $\mathbf{A}^{(n+1)}$ and vector $\mathbf{b}^{(n+1)}$ are defined as follows

$$A_{ij}^{(n+1)}=\begin{cases} -\alpha_1\delta_x^{-\vartheta_2}\gamma_{\delta_t}^{(2)}\mu_{i-j+1}^{(\vartheta_2)},\ i\ge j+2,\\ -\alpha_1\delta_x^{-\vartheta_2}\mu_2^{(\vartheta_2)},\ i=j+1,\\ 1-\alpha_1\delta_x^{-\vartheta_2}\mu_1^{(\vartheta_2)},\ i=j,\\ -\alpha_1\delta_x^{-\vartheta_2},\ i=j-1,\\ 0,\ i<j-1, \end{cases}\tag{2.26}$$

and

$$\mathbf{b}^{(n+1)}=(1+\delta_t^{-1}\gamma_{\delta_t}^{(2)})\mathbf{V}^{(n)}-\sum_{k=1}^{n}a_k^{(\omega_2)}(\mathbf{V}^{(n+1-k)}-\mathbf{V}^{(n-k)})+\gamma_{\delta_t}^{(2)}\mathbf{Z}_2^{(n+1)},$$
$$n=0,1,2,...N-1\ .\tag{2.27}$$

2.3. **Stability and convergence results.** Applying the similar techniques as in [7], we state the analogous results for models (2.1) and (2.5).
Remark 3.2.1 The implicit finite difference schemes defined by (2.16) and (2.22) are unconditionally stable.
Theorem 3.2.1 Let V_j^n be the numerical solution computed by using the implicit

finite difference schemes (2.16) and (2.22), and $v(x,t)$ be the exact solution of (2.1)-(2.3) and (2.5)-(2.7), then a positive constant C_v exists, such that

$$\|V_j^n - v(x_j,t_n)\| \leq C_v(\delta_t + \delta_x),\ j = 1,2,,...,J,\ n = 0,1,2,...N. \tag{2.28}$$

2.4. **Full coefficient matrices.** By the Gerschgorin theorem the eigenvalues of the coefficient matrix $\mathbf{B}^{(n+1)}$ defined in (2.19) lie in the disks centered $B_{ii}^{(n+1)} = 1 - \gamma_{\delta_t}^{(1)} k_i \mu_1^{(\vartheta_1)} \delta_x^{-\vartheta_1} - \gamma_{\delta_t}^{(1)} h_i$ with radius $R_i = \sum_{j=1_{j\neq i}}^{J-1} |B_{ij}^{(n+1)}|$.
Now by (2.11), we have

$$R_i = \gamma_{\delta_t}^{(1)} k_i \delta_x^{-\vartheta_1} \sum_{j=0_{j\neq 1}}^{i+1} \mu_j^{(\vartheta_1)} \leq -\gamma_{\delta_t}^{(1)} k_i \delta_x^{-\vartheta_1} \mu_1^{(\vartheta_1)}. \tag{2.29}$$

Since $h_i \leq 0$ and $k_i \geq 0$, then each eigenvalue λ of $\mathbf{B}^{(n+1)}$ defined in (2.19) satisfies the following inequality

$$|\lambda| \geq B_{ii}^{(n+1)} - R_i \geq 1 - \gamma_{\delta_t}^{(1)} h_i \geq 1. \tag{2.30}$$

Therefore the coefficient matrix $\mathbf{B}^{(n+1)}$ defined in (2.19) is a nonsingular matrix.
Using the similar argument, we can show that the coefficient matrix $\mathbf{B}^{(n+1)}$ defined in (2.25) is nonsingular.

3. Overview of Preconditioned GMRES and Preconditioned CGN

In this section, we briefly describe the background of the preconditioned GMRES and preconditioned CGN for solving large-scale linear systems arising from fractional models in the previous sections. These methods are powerful tools for solving huge systems of linear algebraic equations. The significant advantages of these methods such as low memory requirements and good approximation properties make them very popular.

3.1. **Preconditioned GMRES.** The Generalized Minimum Residual method (GMRES) has the property of minimizing the norm of the residual at each step over a Krylov subspace. The algorithm GMRES is derived from the Arnoldi process for constructing an orthogonal basis of Krylov subspace (see [15]). Since the residual norm is minimized at each step, we would expect GMRES to be convergent for sufficiently large step. The algorithm of standard GMRES method is formulated in [15].
Now, suppose that P_i be the space of all polynomials of *degree* $\leq i$, the following result was proved in [16] for the GMRES method.

Theorem 2 At step i of the GMRES iteration, the residual $\mathbf{r}_i^{(n+1)} = \mathbf{b}^{(n+1)} - \mathbf{B}^{(n+1)}\mathbf{V}_i^{(n+1)}$ satisfies

$$\frac{\|\mathbf{r}_i^{(n+1)}\|}{\|\mathbf{b}^{(n+1)}\|} \leq \kappa(\mathbf{W}^{(n+1)}) \inf_{p_i\in P_i, p_i(0)=1} \sup |p_i(\lambda)|_{\lambda\in\Lambda(\mathbf{B}^{(n+1)})}, \tag{3.1}$$

where $\Lambda(\mathbf{B}^{(n+1)})$ is the spectrum of $\mathbf{B}^{(n+1)}$, $\mathbf{W}^{(n+1)}$ is a nonsingular matrix of eigenvectors and $\kappa(W)$ is the condition number of $\mathbf{W}^{(n+1)}$.
Therefore the rate of convergence of the GMRES method depends on the distribution

of the eigenvalues of $\mathbf{B}^{(n+1)}$ in the complex plane, and if properly normalized degree i polynomials can be found whose size on the $\Lambda(\mathbf{B}^{(n+1)})$ decreases quickly with i, then our method based on GMRES converges quickly.
If $\mathbf{B}^{(n+1)}$ is not too far from normal in the sense that $\kappa(\mathbf{W}^{(n+1)})$ is not too large, the convergence speed of GMRES method is determined by the value

$$\inf_{p_i\in P_i, p_i(0)=1} \sup |p_i(\lambda)|_{\lambda\in\Lambda(\mathbf{B}^{(n+1)})}. \tag{3.2}$$

Assume that there are s eigenvalues $\lambda_1, \lambda_2, ..., \lambda_s$ of $\mathbf{B}^{(n+1)}$ with non-positive real parts and let the other eigenvalues be enclosed in a disk on complex plane with radius $r > 0$ and center at $c > 0$. Then the polynomial $p_i(z) = (1-\frac{z}{\lambda_1})(1-\frac{z}{\lambda_2})...(1-\frac{z}{\lambda_s})(\frac{z-c}{c})^{J-1-s}$ can be used to show that

$$\inf_{p_i\in P_i, p_i(0)=1} \sup |p_i(\lambda)|_{\lambda\in\Lambda(B^{(n+1)})} \leq (\frac{\max_{i=1,...,s,\ j=i+1,...,J-1}\{|\lambda_i-\lambda_j|\}}{\min_{i=1,...,s}\{|\lambda_i|\}})^s(\frac{r}{c})^{J-1-s}. \tag{3.3}$$

Therefore, when $\kappa(\mathbf{W}^{(n+1)})$ is not too large, the right hand side of (3.1) represents a good convergence bound.
If $\mathbf{W}^{(n+1)}$ is far from normal, the bound (3.1) may fail to provide any reasonable information about the GMRES convergence. We note that in the nonnormal case the GMRES convergence behavior is significantly more difficult to analyze than in the normal case. For more details refer to [8].
Preconditioning is a technique that can accelerate the convergence of iterative methods. We can apply this technique to speed up the convergence rate of GMRES method (see [16]). In order to justify the idea of this technique note that the convergence rate of GMRES method for solving the linear system $\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)} = \mathbf{b}^{(n+1)}$ depends on the spectral properties of $\mathbf{B}^{(n+1)}$. For ill-conditioned problems GMRES is slowly convergent or even in some of cases is divergent. For preconditioning the linear system $\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)} = \mathbf{b}^{(n+1)}$, it is often preferable to transform it into an equivalent one

$$\textit{Left preconditioning}: \ {\mathbf{M}^{(n+1)}}^{-1}\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)} = {\mathbf{M}^{(n+1)}}^{-1}\mathbf{b}^{(n+1)}.$$

$$\textit{Right preconditioning}: \ \mathbf{B}^{(n+1)}{\mathbf{M}^{(n+1)}}^{-1}\mathbf{U}^{(n+1)} = \mathbf{b}^{(n+1)}, \ \mathbf{U}^{(n+1)} = \mathbf{M}\mathbf{V}^{(n+1)},$$

or

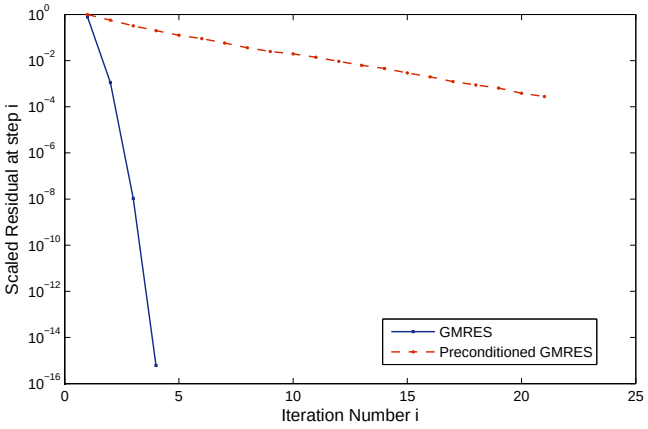
$$\textit{Split preconditioning}: \quad {\mathbf{L}^{(n+1)}}^{-1}\mathbf{B}^{(n+1)}{\mathbf{U}^{(n+1)}}^{-1}\mathbf{Z}^{(n+1)} = L^{-1}{\mathbf{L}^{(n+1)}}^{-1}\mathbf{b}^{(n+1)},$$
$${\mathbf{U}^{(n+1)}}^{-1}\mathbf{Z}^{(n+1)} = \mathbf{V}^{(n+1)}.$$

In practice, the preconditioner $\mathbf{M}^{(n+1)}$ should satisfy the following properties

- A preconditioner matrix ${\mathbf{M}^{(n+1)}}^{-1}\mathbf{B}^{(n+1)}$ must be better conditioned than $\mathbf{B}^{(n+1)}$ (see [16]),
- The cost of constructing a preconditioner should also be cheap to make the preconditioned system easy to be solved (see [16]). A preconditioning matrice $\mathbf{M}^{(n+1)}$ can be computed in $O(i\log_2 i)$ operations,

thus the linear system ${\mathbf{M}^{(n+1)}}^{-1}\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)} = \mathbf{b}^{(n+1)}$ can be expected to converge quickly.

FIGURE 1. Convergence histories of GMRES for preconditioning technique.



The detailed comments concerning the preconditioned GMRES algorithm (PGMRES) can be found in [17].

3.2. **Preconditioned CGN.** One of the simplest methods for solving nonsymmetric linear system $\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)} = \mathbf{b}^{(n+1)}$ is to apply the CG iterations to the normal equations $\mathbf{B}^{(n+1)^T}\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)} = \mathbf{B}^{(n+1)^T}\mathbf{b}^{(n+1)}$ which is called the CGN method [16]. We note that the convergence of CGN is controlled by the eigenvalue of $\mathbf{B}^{(n+1)^T}\mathbf{B}^{(n+1)}$. Thus the convergence of CGN is determined by the singular value of $\mathbf{B}^{(n+1)}$. Now, suppose that the CG iterations be applied to the normal equations $\mathbf{B}^{(n+1)^T}\mathbf{B}^{(n+1)}\mathbf{V}^{(n+1)} = \mathbf{B}^{(n+1)^T}\mathbf{b}^{(n+1)}$, where $\mathbf{B}^{(n+1)^T}\mathbf{B}^{(n+1)}$ has 2-norm condition number κ. Then the A-norm of the errors satisfy

$$\frac{\|\mathbf{r}_i^{(n+1)}\|_2}{\|\mathbf{r}_0^{(n+1)}\|_2} \leq 2(\frac{\kappa-1}{\kappa+1})^i, \tag{3.4}$$

which implies if κ is reseanblely large, convergence to a specified tolerance can be expected in $O(\kappa)$ iteration. A disadvantage of the CGN method is that the convergence rate can be slow. Therefore, preconditioner technique could be exploited to accelerate the convergence rate of the CGN method. The preconditioned CGN (PCGN) algorithm is given in [18].

4. NUMERICAL EXPERIMENTS

In this section, we report the results of numerical experiments to compare the performance and efficiency of preconditioned methods that introduced in the previous section.
All programs run in MATLAB R2012a. The tests have been performed on a computer with the configuration:

- Intel(R) Core(TM) i3-3217U @ 1.80 GHz 1.80 GHz CPU

FIGURE 2. Left hand figure: Eigenvalues distribution of nopreconditioning ($J = 200$, $N = 200$). Right hand figure: Eigenvalues distribution of sparse band and diagonal preconditioning ($J = 200$, $N = 200$).

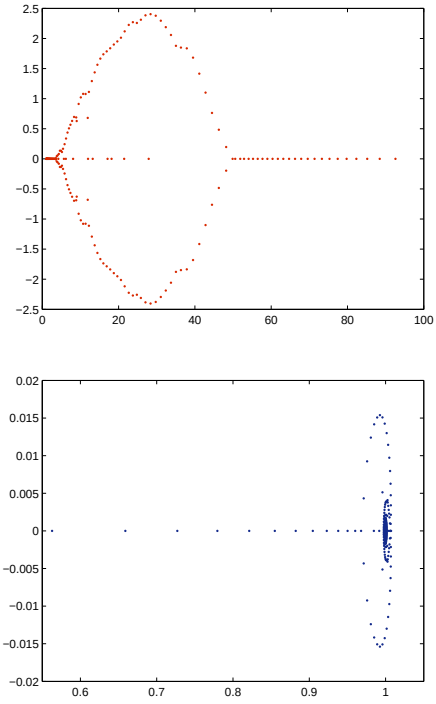


FIGURE 3. Convergence histories of CGN for preconditioning technique.

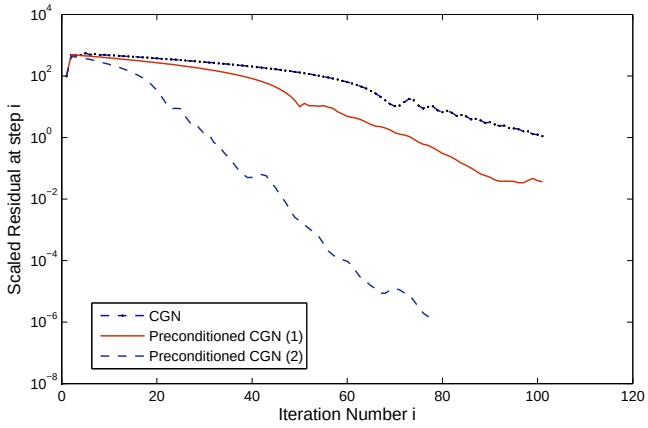
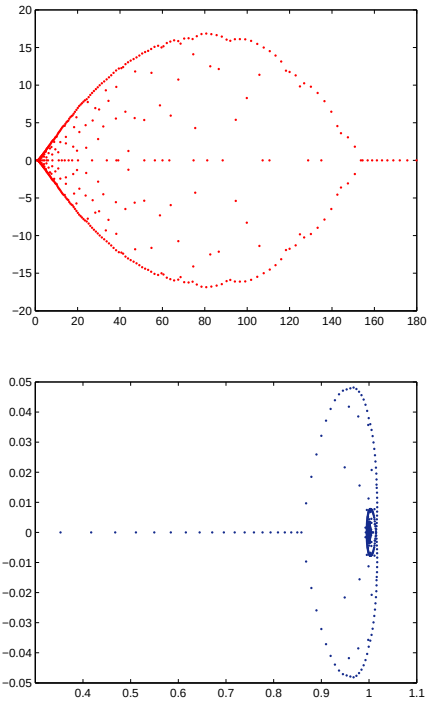


FIGURE 4. Left hand figure: Eigenvalues distribution of noprecon-ditioning ($J = 500$, $N = 500$) Right hand figure: Eigenvalues distribution of ILU preconditioning ($luinc(\mathbf{B}^{(n+1)^T}\mathbf{B}^{(n+1)}, 0.01)$, $J = 500$, $N = 500$.



- 4GB RAM memory under Windows seven

The stopping criterion is

$$\frac{\|\mathbf{r}_i^{(n+1)}\|_2}{\|\mathbf{b}^{(n+1)}\|_2} < \epsilon, \tag{4.1}$$

where $\mathbf{r}_i^{(n+1)}$ is the residual vector after i iterations.

Example 1. In this example, we consider the time-independent space-fractional convection-diffusion model with $\omega_1 = 0.8$, $\vartheta_1 = 1.5$, $k(x) = x^2$, $h(x) = -1$,$\phi(x) = x^2(1-x)$ and

$$\begin{aligned} z_1(x,t) &= 1.815207368t^{\frac{6}{5}}x^2 - 1.815207368t^{\frac{6}{5}}x^4 - 2.256758334\sqrt{t}x^4 \\ &+ 2.256758334x^6\sqrt{t} + x^2 + x^2t^2 - x^4 - x^4t^2,\ 0 < x < 1,\ 0 < t \leq 1. \end{aligned} \tag{4.2}$$

In order to solve the linear system arising from (4.2) two strategies are proposed:

TABLE 1. The average number of iterations of GMRES and preconditioned GMRES.

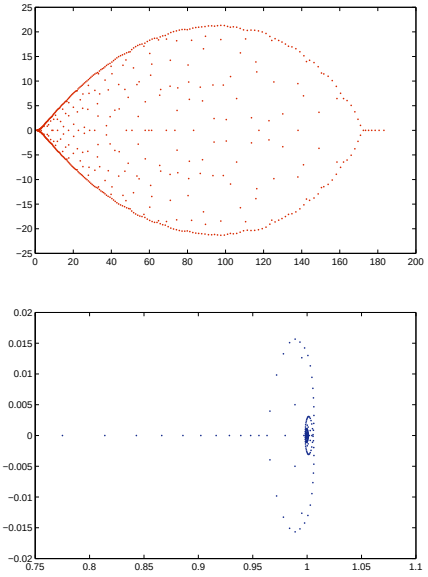
method	*nopre*	*SOR*	*SSOR*
$GMRES(J = 600, N = 600)$	41	5	4
$GMRES(J = 700, N = 700)$	58	6	4

a): GMRES with band and diagonal preconditioning and without preconditioning ($\epsilon = 10^{-17}$).
Band and Diagonal Preconditioner: A simple preconditioner would be sparse band and diagonal matrix constructed by considering the special structure of the original matrix. The toolbox of *MATLAB* supplies a function *spdiags*, that can be used to produce sparse band and diagonal preconditioner. In our tests, we use
$d = [-10 : 10]$;
$\mathbf{C}^{(n+1)} = spdiags(\mathbf{B}^{(n+1)}, d)$;
$[m, m] = size(\mathbf{B}^{(n+1)})$;
$\mathbf{M}^{(n+1)} = spdiags(\mathbf{C}^{(n+1)}, d, m, m)$;

b): CGN with incomplete Cholesky preconditioning and without preconditioning ($\epsilon = 10^{-8}$).
ILU preconditioner: Incomplete LU (ILU) factorisations constitute one of the best known classes of general-purpose preconditioners. It is known that ILU preconditioners tend to cluster eigenvalues. A general ILU factorization computes a sparse approximation of the LU factorization. Incomplete LU factorization is often used as a preconditioner (see [17]). The toolbox of *Matlab* supplies a function *luinc*, incomplete lu factorization, that can be used to produce ILU preconditioners. The idea behind *luinc* is simple: start a normal LU factorization but if any entry in the factorization is small, set it to zero. In the above-mentioned example we used ILU preconditioner where:
a) Preconditioner (1): $[\mathbf{L}^{(n+1)}, \mathbf{U}^{(n+1)}] = luinc(\mathbf{B}^{(n+1)^T}\mathbf{B}^{(n+1)}, '0')$ is no fill ILU decomposition
b) Preconditioner (2): $[\mathbf{L}^{(n+1)}, \mathbf{U}^{(n+1)}] = luinc(\mathbf{B}^{(n+1)^T}\mathbf{B}^{(n+1)}, \tau)$ is an ILU decomposition of A with threshold $\tau = 0.01$.

We choose $J = 200$, $N = 200$. Figure 1 shows convergence curve corresponding to 23 steps of the GMRES and a preconditioned GMRES iterations. As Figure 1 illustrates, the GMRES iterations for this linear system converges slowly. The condition numbers of $\mathbf{B}^{(n+1)}$ and $\mathbf{W}^{(n+1)}$ are 98.9 and 3.27, respectively, so the deterioration in convergence can be explained by conditioning alone. When an iterative method stagnates like this, it is time to look for a better preconditioner. Distributions of the eigenvalues of matrix $\mathbf{B}^{(n+1)}$ and matrix $\mathbf{M}^{(n+1)^{-1}}\mathbf{B}^{(n+1)}$ are shown in Figure 2. It is clear that the eigenvalues of the system are very scattered and such distribution is not favorable to the rapid convergence of GMRES. From Figure 2, it can be seen

FIGURE 5. Left hand figure: Eigenvalues distribution of nopreconditioning ($J = 600$, $N = 600$). Right hand figure: Eigenvalues distribution of sparse band and diagonal preconditioning ($J = 600$, $N = 600$).



that the eigenvalues of preconditioning are clustered around 1. Therefore, one goal of preconditioning is to improve this distribution by grouping the eigenvalues into a few small clusters and around 1 as approximate as possible.

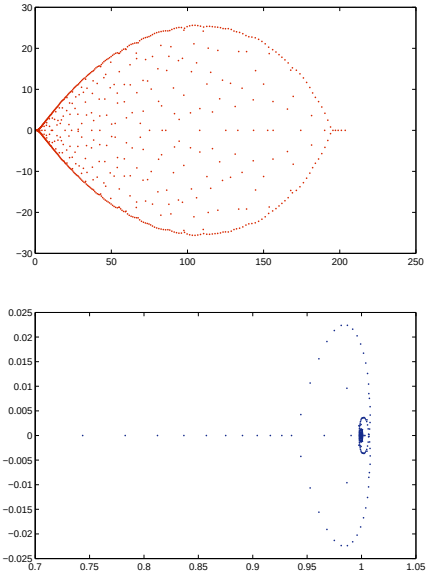
For second strategy, we choose $J = 500$, $N = 500$. The CGN algorithm, without preconditioning, converges very badly as shown in Figure 3. This can be explained by looking to the eigenvalues. As can be seen in Figure 4, most of the eigenvalues are close to zero, which leads to a bad convergence. The eigenvalues of the ILU preconditioning are shown in Figure 4. The difference in convergence speed between a preconditioned matrix with eigenvalues close to 1 and an unpreconditioned matrix with the eigenvalues close to zero is shown in Figure 4.

Example 2. As second example, we consider the fractional advection-dispersion model with $\omega_2 = 0.5$, $\vartheta_2 = 1.5$, $\alpha_1 = -1$, $\phi(x) = x^2$ and

$$z_2(x,t) = x^2 + 1.128379167\sqrt{t}x^2 + 2.256758334x^{\frac{3}{2}} + 2.256758334x^{\frac{3}{2}}t, \quad 0 < x < 1,\ 0 < t \leq 1. \tag{4.3}$$

The results of the previous example show that the CGN in composition preconditioners don't work very well but by using GMRES method in composition preconditioners, we get less iteration number than CGN method. In order to solve the linear system arising (4.3) a strategy is proposed:

FIGURE 6. Left hand figure: Eigenvalues distribution of nopreconditioning ($J = 700$, $N = 700$). Right hand figure: Eigenvalues distribution of sparse band and diagonal preconditioning ($J = 700$, $N = 700$).



a: GMRES in combination of the SOR and SSOR preconditioners and without preconditioning ($\epsilon = 10^{-7}$).
Preconditioner based on relaxation technique: Suppose that $\mathbf{B}^{(n+1)} = \mathbf{D}^{(n+1)} - \mathbf{E}^{(n+1)} - \mathbf{F}^{(n+1)}$, in which $\mathbf{D}^{(n+1)}$ is the diagonal of A, $-\mathbf{E}^{(n+1)}$ its strict lower part, and $-\mathbf{F}^{(n+1)}$ its strict upper part, as was seen in [17], the SOR and SSOR preconditioner are defined by a) SOR Preconditioning: $\mathbf{M}^{(n+1)} = \frac{1}{\omega}(\mathbf{D}^{(n+1)} - \omega\mathbf{E}^{(n+1)})$,
b) SSOR Preconditioning: $\mathbf{M}^{(n+1)} = \frac{1}{\omega(2-\omega)}(\mathbf{D}^{(n+1)} - \omega\mathbf{E}^{(n+1)})\mathbf{D}^{(n+1)^{-1}}(\mathbf{D}^{(n+1)} + \omega\mathbf{F}^{(n+1)})$,
where ω is called the relaxation parameter. The more explanations of the preconditioners can be found in [17].

Figure 4 and Figure 5 depict the spectral distribution of the original coefficient matrices and the preconditioned matrices with the SSOR preconditioner. The eigenvalues of the preconditioned matrices are well clustered around 1 while the eigenvalues of the original coefficient matrices lie in a much larger range. The numbers of iterations required for convergence of the GMRES in combination of various preconditioning is

shown in Table 1. We can see that the average number of iterations required by the preconditioned GMRES method is less than GMRES method.

5. Concluding remarks

We present two preconditioned iterative methods to solve linear systems arising from the discretization of fractional advection-diffusion equations. Numerical experiments confirm that the methods perform very well, easily obtaining the solution of large systems that are infeasible to solve using traditional methods.

References

[1] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R . Pozo, C. Romine. and H. Van der Vorst, *Templates for the solution of linear systems: Building blocks for iterative methods*, SIAM, 1994.

[2] D. Benson, R. Schumer, M. M. Meerschaert and S. W. Wheatcraft, *Fractional dispersion, Lvy motion, and the MADE tracer tests*, Transport Porous Media *42* (2001), 21140.

[3] D. Benson D, S. W. Wheatcraft, and M. M. Meerschaert, *The fractional-order governing equation of Lvy motion*. Water Resour Res., *36* (2000), 141323.

[4] K. Burrage, N. Hale, and D. Kay, *An Efficient Implicit FEM Scheme for Fractional-in-Space ReactionDiffusion Equations*, SIAM J. Sci. Comput. *34* (4) (2012), 21452172.

[5] M. Ekici, A. Sonmezoglu, and E. M. Zayed, *A new fractional sub-equation method for solving the space-time fractional differential equations in mathematical physics, Computational Methods for Differential Equations*, *34* (2014), 153-170.

[6] Y. Lin and C. Xu, *Finite difference / spectral approximations for the time-fractional diffusion equation*, Journal of Computional Physics, *225* (2007), 1533-1552.

[7] F. Liu, P. Zhuang, V. Anh, I. Turner and K. Burrage, *Stability and convergence of the difference methods for the space-time fractional advection-diffusion equation*, Applied Mathematics and Computation, *191* (2007), 12-20.

[8] J. Liesen and P. Tich Y, *The worst-case GMRES for normal matrices*, BIT, 44 (2004), 7998.

[9] M. M. Meerschaert and C. Tadjeran, *Finite difference approximations for fractional advectiondispersion flow equations*, J Comput Appl Math *172* (2004), 6577.

[10] M. M. Meerschaert and C. Tadjeran, *Finite difference approximations for two-sided space-fractional partial differential equations*, Appl Numer Math *56* (2006) 8090.

[11] T. Moroney and Q. Yang, *A banded preconditioner for the two-sided, nonlinear space-fractional diffusion equation*, Computers and Mathematics with Applications *5* (2013), 659-667.

[12] A. A. Neamaty, B. Agheli, and M. Adabitabar. *Numerical solution for boundary value problem of fractional order with approximate Integral and derivative*, Computational Methods for Differential Equations *3* (2014) 195-204.

[13] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM journal on numerical analysis *12* (1975), 617-629.

[14] I. Podlubny, *Fractional Differential Equations, Academic Press*, New York, 1974.

[15] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. *7* (1986) 856-869.

[16] L. N. Trefehen and D. Bau, *Numerical linear algebra*, SIAM, Philadelphoa, 1997.

[17] Y. Saad, *Iterative methods for sparse linear systems*, SIAM, Philadelphoa, 2003.

[18] Y. Saad, *A flexible inner-outer preconditioned gmres algorithm*, SIAM J. Sci. Comput. *14* (2) (1993) 461469.

[19] H. Wang, K. Wang, T. Sircar, *Adirect* $O(N \log^2 N)$ *finite difference method for fractional diffusion equations*, Journal of Computional Physics, *229* (2010), 8095-8104.

[20] K. Wang and H. Wang, *A fast characteristic finite difference method for fractional advection-diffusion equations*, Advances in water resources *34*7 (2011), 810-816.

[21] Q. Yang, I. Turner, F. Liu, M. Ilic , *Novel numerical methods for solving the time-space fractional diffusion equation in 2D*, SIAM J. Sci. Comput. *33* (3) (2011), 11591180.

[22] Q. Yang, T. Moroney, K. Burrage, I. Turner, and F. Liu, *Novel numerical methods for time-space fractional reaction diffusion equations in two dimensions*, Aust. New Zealand Ind. Appl. Math. J. *52* (2011), 395409.

[23] Q. Yang, T.J. Moroney, F. Liu, and I. Turner, *Computationally efficient methods for solving time-variable-order timespace fractional reaction-diffusion equation*, in: Proceedings of the Fifth IFAC Symposium on Fractional Differentiation and its Applications.