Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 8(12): 29474–29489.
DOI: 10.3934/math.20231509
Received: 02 June 2023
Revised: 07 October 2023
Accepted: 15 October 2023
Published: 31 October 2023

Research article

Investigation and analysis of the numerical approach to solve the multi-term time-fractional advection-diffusion model

Yones Esmaeelzade Aghdam[1,*], **Hamid Mesgarani**[1], **Zeinab Asadi**[1] **and Van Thinh Nguyen**[2,*]

[1] Department of Mathematics, Faculty of Science, Shahid Rajaee Teacher Training University, Tehran, 16785-136, Iran

[2] Department of Civil and Environmental Engineering, Seoul National University, Seoul, South Korea

* **Correspondence:** Email: yonesesmaeelzade@gmail.com; vnguyen@snu.ac.kr.

Abstract: In this paper, a methodical approach is presented to approximate the multi-term fractional advection-diffusion model (MT-FAD). The Lagrange squared interpolation is used to discretize temporal fractional derivatives, and Legendre polynomials are shifted as an operator to discretize the spatial fractional derivatives. The advantage of these numerical techniques lies in the orthogonality of Legendre polynomials and its matrix operations. A quadratic implicit design as well as its stability and convergence analysis are evaluated. It should be noted that the theoretical proof obtained from this study represents the first results for these numerical schemes. Finally, we provide three numerical examples to verify the validity of the proposed methods and demonstrate their accuracy and effectiveness in comparison with previous studies shown in [W. P. Bu, X. T. Liu, Y. F. Tang, J. Y. Yang, Finite element multigrid method for multi-term time fractional advection diffusion equations, *Int. J. Model. Simul. Sci. Comput.*, **6** (2015), 1540001].

Keywords: multi-term fractional advection-diffusion model; collocation method; Legendre polynomial; convergence analysis
Mathematics Subject Classification: 65L60, 65N12, 35R11

1. Introduction

Diffusion refers to the movement of substances in a physical system due to differences in the concentration of regions. The transfer of quantity due to the bulk movement of fluid is called advection [1, 2]. In combination of the advection and diffusion equations, the transfer of substance or quantity (particle, energy or chemical substances) in a medium is explained, namely the advection-diffusion equation. The advection-diffusion equation is a partial differential equation used to describe

physical phenomena, including two processes of advection and diffusion [3, 4].

In recent years, researchers observed that the classical advection-diffusion equation is insufficient in explaining some physical phenomena. For example, in the classical equation of motion, the diffusion process is assumed to be linear, while there are many media in which the transport motion is not linear [5]. Therefore, the fractional advection-diffusion equation considers the non-linearity of the transport movement in the media and the feature of the memory effects (the effect of the previous transport processes on the current state of media) for the system to fix the failures of the classical equation. Because the fractional equation includes all-around movements, it is more accurate than the classical model [6–9]. The fractional advection-diffusion equation was used for the first time in 1990 to describe the physical phenomenon of the anomalous diffusion of particles in disordered systems. After that, other researchers used the fractional equation to describe various physical phenomena until 2000 [5].

The fractional advection-diffusion equation is a powerful tool for modeling transport phenomena in complex systems. For example, one of the applications of the advection-diffusion equation is modeling the transport of passive tracers in groundwater in porous media [10]. Apprehending the advection-diffusion instrument is important for forecasting and overseeing the transport procedures of substances in different environments. However, there are many transport procedures in complex systems controlled by non-Brownian diffusion, which is well described by fractional calculus, and fractional advection-dispersion equations recently play an important role in the modeling of transport processes in various physical [11–19] and natural [20–23] systems. However, solving such equations is not always possible, because finding an answer for the fractional advection-diffusion equation is more difficult than the classical advection-diffusion equation. In addition, the fractional advection-diffusion equation may not be appropriate for all media, therefore, it is crucial to choose appropriate equations to represent advection-diffusion processes in particular systems [24, 25].

In general, various numerical methods can be employed to solve the fractional advection-diffusion equations, including finite difference, finite element, spectral, and Monte Carlo methods. However, the use of the Legendre-shifted polynomials [24] to approximate the fractional advection-diffusion equation is limited. The Legendre-shifted polynomials constitute a specific type of polynomials, which are suitable for approximating functions with special boundary conditions. The main feature of these polynomials is to calculate a more accurate solution using fewer terms than other methods [26, 27]. A type of multi-time fractional advection-diffusion (MT-FADE) equation with boundary and initial conditions is defined as follows:

$$\mathrm{P}(D_\tau)v(x,\tau) = K_1\frac{\partial^{2\beta_1}v(x,\tau)}{\partial|x|^{2\beta_1}} + K_2\frac{\partial^{2\beta_2}v(x,\tau)}{\partial|x|^{2\beta_2}} + q(x,\tau),\ 0 < x < 1,\ 0 < \tau < T, \tag{1.1}$$

with initial $v(x,0) = z(x),\ 0 \le x \le 1$ and boundary conditions $v(0,\tau) = r(\tau),\ v(1,\tau) = s(\tau),\ 0 \le \tau \le T$. In Eq (1.1), the fractional factor $\mathrm{P}(D_\tau)$ is described as $\mathrm{P}(D_\tau)v(x,\tau) = \sum_{e=0}^{\epsilon} b_e {}_0^C D_\tau^{\alpha_e} v(x,\tau),\ \epsilon \in \mathbb{N}$ with $0 < \alpha_e < \alpha_{e-1} < \ldots < \alpha_0 \le 1$ and $b_e \ge 0$. Terms $\frac{\partial^{2\beta_i}v(x,\tau)}{\partial|x|^{2\beta_i}},\ i = 1, 2$ are the Riesz fractional derivative with respect to x are described as

$$\frac{\partial^{2\beta_i}v(x,\tau)}{\partial|x|^{2\beta_i}} = \frac{-1}{2\cos(\beta_i\pi)}({}_xD_L^{2\beta_i}v(x,\tau) + {}_xD_R^{2\beta_i}v(x,\tau)),$$

where ${}_xD_L^{2\beta_i}v(x,\tau)$ and ${}_xD_R^{2\beta_i}v(x,\tau)$ describe the left and right fractional Riemann-Liouville derivatives specified as below:

$$
\begin{aligned}
{}_xD_L^{2\beta_i}v(x,\tau) &= \frac{1}{\Gamma(2-2\beta_i)}\Big(\frac{d}{dx}\Big)^n \int_0^x (x-\zeta)^{n-2\beta_i} v(\zeta,\tau)d\zeta,\\
{}_xD_R^{2\beta_i}v(x,\tau) &= \frac{1}{\Gamma(2-2\beta_i)}\Big(-\frac{d}{dx}\Big)^n \int_x^1 (\zeta-x)^{n-2\beta_i} v(\zeta,\tau)d\zeta,
\end{aligned}
$$

where $n-1 < 2\beta_i \leq n,\ n \in \mathbb{N}$.

It is noted that the well-known Eq (1.1) consists of some classic fractional instances. Using $e = 0$ in the fractional factor $\mathrm{P}(D_\tau)$ results in traditional advection-diffusion equation [28, 29]. Taking $K_1 = 0,\ \beta_2 = 1$, the multi-term time-fractional diffusion equations are obtained [30, 31]. Taking $\alpha_0 = 2,\ \alpha_1 = 1$ and $K_1 = 0$ in the case $1 < \alpha_e < \alpha_{e-1} < \ldots < \alpha_0 \leq 2$ results in a spatially fractional Telegraph equation [32].

The MT-FAD model describes the changes in a quantity on multiple time scales. It is a generalization of the classical advection-diffusion equation. We often use these equations to model physical phenomena such as heat conduction, diffusion, and wave propagation [30, 33]. There are two general analytical and numerical solutions to solve these equations. There are analytical ways for some MT-FAD model, although most of these answers are not accurate. In using numerical methods, we usually use finite difference methods [33, 34]. In this way, we first discretize time and space and then solve the resulting system of equations using numerical techniques. In most cases, finite difference methods are the only possible way to solve MT-FAD model [30]. Many authors discussed the solution of MT-FAD model with numerical and analytical methods [35, 36]. It is worth mentioning that Legendre-shifted polynomials are a specific type of polynomials, and they are suitable for approximating functions with special boundary conditions. The main feature of these polynomials is to calculate a solution more accurately using fewer terms than other methods. In this study, we proposed a novel approach to solve multi-term time-fractional advection-diffusion equation (1.1). We introduce Lagrange linear interpolation for temporal discretization and Legendre approximation for spatial discretization in Section 2. The highlights of the proposed numerical methods are unconditional stability and convergence, which are presented in Section 3. To demonstrate the effectiveness of the proposed numerical methods, we conducted some numerical experiments illustrated in Section 4. Eventually, the conclusion is summarized in Section 5.

2. Discretization methods

This section consults the discretization of relation Eq (1.1). To do so, let δh and $\delta\tau = \frac{T}{M}$ denote the sizes of the spatial and temporal discretization steps, respectively. We describe the partition in space as $\{x_i\}_{i=0}^M$ that is the root of the basis polynomials and the partition in time as $\tau_j = j\delta\tau,\ j = 0, 1, \ldots, M$. We denote the discrete values of the variables $v(x,\tau)$ and $q(x,\tau)$ as

$$
v(x,\tau_k) = v^k(x), \quad v(x_i,\tau_k) = v_i^k, \quad q(x_i,\tau_k) = q_i^k, \ \ k = 0, 1, \ldots, M.
$$

Considering the fact that $0 < \alpha_e \leq 1$, the Caputo fractional derivative can be utilized instead of the Riemann-Liouville.

2.1. Time-discretization scheme

To start off on discretization, we use S_2 method in paper [37]. First, we must approximate unknown function for the case that $j = 0$ in the intervals $[\tau_0, \tau_1]$ by Lagrange linear interpolation, then, for the rest of the nodes $j \geq 0$ in the interval $[\tau_{j-1}, \tau_j]$ we apply Lagrange square interpolation. Practically, we have three nodes to conduct the approximation.

$$ {}_0^C D_\tau^{\alpha_e} v^k(x) = \frac{\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \sum_{j=0}^{k} \mathcal{S}_{k,j}^{\alpha_e} v^j(x) + O(\delta\tau^{3-\alpha_e}), \tag{2.1} $$

where for $k = 1$ and $k = 2$ the coefficients $\mathcal{S}_{k,j}^{\alpha_e}$ is as

$$ \mathcal{S}_{1,j}^{\alpha_e} = \begin{cases} -\mathcal{A}_1, & j = 0, \\ \mathcal{A}_1, & j = 1, \end{cases} \qquad \mathcal{S}_{2,j}^{\alpha_e} = \begin{cases} -\mathcal{A}_2 + \mathcal{B}_{2,2}, & j = 0, \\ \mathcal{A}_2 + \mathcal{C}_{2,2}, & j = 1, \\ \mathcal{D}_{2,2}, & j = 2, \end{cases} $$

and for $k \geq 3$, we have

$$ \mathcal{S}_{k,j}^{\alpha_e} = \begin{cases} -\mathcal{A}_k + \mathcal{B}_{k,j+2}, & j = 0, \\ \mathcal{A}_k + \mathcal{B}_{k,j+2} + \mathcal{C}_{k,j+1}, & j = 1, \\ \mathcal{B}_{k,j+2} + \mathcal{C}_{k,j+1} + \mathcal{D}_{k,j}, & 2 \leq j \leq k-2, \\ \mathcal{C}_{k,j+1} + \mathcal{D}_{k,j}, & j = k-1, \\ \mathcal{D}_{k,j}, & j = k, \end{cases} $$

in which

$$ \begin{aligned} \mathcal{A}_k &= k^{1-\alpha_e} - (k-1)^{1-\alpha_e}, \\ \mathcal{B}_{k,j} &= \frac{1}{2-\alpha_e}[(k-j+1)^{1-\alpha_e}(k-j+\frac{\alpha_e}{2}) - (k-j)^{1-\alpha_e}(k-j-\frac{\alpha_e}{2}+1)], \\ \mathcal{C}_{k,j} &= \frac{2}{2-\alpha_e}[(k-j)^{1-\alpha_e}(k-j-\alpha_e+2) - (k-j+1)^{2-\alpha_e}], \\ \mathcal{D}_{k,j} &= \frac{1}{2-\alpha_e}[(k-j+1)^{1-\alpha_e}(k-j-\frac{\alpha_e}{2}+2) - (k-j)^{1-\alpha_e}(k-j-\frac{3\alpha_e}{2}+3)]. \end{aligned} $$

Applying the discretization of the relation (2.1) for the left side of Eq (1.1), we have

$$ \begin{aligned} &\sum_{e=0}^{\epsilon} \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k,k}^{\alpha_e} v^k(x) - K_1 \frac{\partial^{2\beta_1} v^k(x)}{\partial|x|^{2\beta_1}} - K_2 \frac{\partial^{2\beta_2} v^k(x)}{\partial|x|^{2\beta_2}} \\ &= \sum_{e=0}^{\epsilon} \sum_{j=0}^{k-1} \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k-1,j}^{\alpha_e} v^j(x) + q^k(x) + O(\delta\tau^{3-\max\alpha_e}),\ 0 < x < 1, \end{aligned} \tag{2.2} $$

Letting V_i^k as the approximate solution v_i^k in Eq (2.2), then we have

$$ \sum_{e=0}^{\epsilon} \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k,k}^{\alpha_e} V^k(x) - K_1 \frac{\partial^{2\beta_1} V^k(x)}{\partial|x|^{2\beta_1}} - K_2 \frac{\partial^{2\beta_2} V^k(x)}{\partial|x|^{2\beta_2}} = \sum_{e=0}^{\epsilon} \sum_{j=0}^{k-1} \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k-1,j}^{\alpha_e} V^j(x) + q^k(x). \tag{2.3} $$

2.2. Spatial discretization scheme with Legendre approximation

Now, to get the full discrete of Eq (2.3), we employ the following series:

$$V^k(x) = \sum_{i=0}^{n} \sigma_i^k \mathbb{L}_i^{\circledS}(x), \quad n = 0, 1, \dots N, \quad k = 1, 2, \dots M, \tag{2.4}$$

where N and M are the number point of spatial and temporal directions, respectively. Moreover, $\mathbb{L}_\iota^{\circledS}(x)$ is the shifted orthogonal polynomials in the domain [0, 1] that is defined in paper [38] as

$$\mathbb{L}_i^{\circledS}(x) = \sum_{r=0}^{\lfloor \frac{i}{2} \rfloor} \sum_{\iota=0}^{i-2r} N_{i,r,\iota} x^\iota, i = 0, 1, \dots, \tag{2.5}$$

where

$$N_{i,r,\iota} = \binom{i-2r}{\iota} \frac{(-1)^{i-r-\iota} 2^{\iota-i} (2i-2r)!}{r!(i-r)!(i-2r)!}.$$

Then the unknown coefficients in Eq (2.4) are defined below:

$$\sigma_i^k = (2i+1) \int_0^1 \mathbb{L}_i^{\circledS}(x) V^k(x) dx.$$

Here is a formula for the approximation of the fractional derivative $\mathbb{L}_i^{\circledS}$ is carried out that we define it with the symbol $\mathbb{L}_i^{\gamma,\circledS}$. Suppose $\gamma > 0$, by using the Caputo linearity property, we get

$$\mathbb{L}_i^{\gamma,\circledS}(x) = \sum_{r=0}^{\lfloor \frac{i}{2} \rfloor} \sum_{\iota=\lceil \gamma \rceil}^{i-2r} N_{i,r,\iota}^{\gamma} x^{\iota-\gamma}, i = 0, 1, \dots, \tag{2.6}$$

in which $\lfloor \gamma \rfloor$ and $\lceil \gamma \rceil$ are the floor and ceiling of the fractional term γ and

$$N_{i,r,\iota}^{\gamma} = \binom{i-2r}{\iota} \frac{(-1)^{i-r-\iota} 2^{\iota-i} (2i-2r)! \Gamma(\iota+1)}{r!(i-r)!(i-2r)! \Gamma(\iota-\gamma+1)}.$$

Notice that for $0 \le i < \lceil \gamma \rceil$ we have $\mathbb{L}_i^{\gamma,\circledS}(x) = 0$. Substituting the operators (2.5) and (2.6) in (2.3), we approximate $V^k(x)$ with respect to x as below:

$$\sum_{i=0}^{n} \sigma_i^k A_{e,i}(x) = \sum_{j=0}^{n-1} \sum_{i=0}^{j} \sigma_i^j B_{e,i}(x) + Q(x), \tag{2.7}$$

where

$$\begin{aligned} A_{e,i}(x) = &\sum_{e=0}^{\epsilon} \sum_{r=0}^{\lfloor \frac{i}{2} \rfloor} \sum_{\iota=0}^{i-2r} \frac{b_e \delta \tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k,k}^{\alpha_e} N_{i,r,\iota} x^\iota + \frac{K_1}{2\cos(\beta_1 \pi)} \sum_{r=0}^{\lfloor \frac{i}{2} \rfloor} \sum_{\iota=\lceil \beta_1 \rceil}^{i-2r} \left(N_{i,r,\iota}^{\beta_1} x^{\iota-\beta_1} + N_{i,r,\iota}^{\beta_1} (1-x)^{\iota-\beta_1} \right) \\ &+ \frac{K_2}{2\cos(\beta_2 \pi)} \sum_{r=0}^{\lfloor \frac{i}{2} \rfloor} \sum_{\iota=\lceil \beta_2 \rceil}^{i-2r} \left(N_{i,r,\iota}^{\beta_2} x^{\iota-\beta_2} + N_{i,r,\iota}^{\beta_2} (1-x)^{\iota-\beta_2} \right), \\ B_{e,i}(x) = &\sum_{e=0}^{\epsilon} \sum_{r=0}^{\lfloor \frac{i}{2} \rfloor} \sum_{\iota=0}^{i-2r} \frac{b_e \delta \tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k-1,j}^{\alpha_e} N_{i,r,\iota} x^\iota, \quad Q(x) = q^k(x). \end{aligned}$$

We use the collocation manner to obtain the coefficients σ_i^k of relation (2.7). For this objective, we get the roots of the shifted Legendre polynomials, $\mathbb{L}_i^{\circledS}(x)$, as the collocation points and substitute them in Eq (2.7) to obtain linear equations at each time step k. $\mathbb{L}_{N-1}^{\circledS}(x)$ has $N-1$ roots, which needs two more conditions to obtain a system of linear equations with $N+1$ equations, which can be obtained using the following boundary conditions for $i = 0, 1, \ldots N$ and $k = 1, 2, \ldots M$.

$$v(0,t) = \sum_{i=0}^{n} \sigma_i^k \mathbb{L}_i^{\circledS}(0) = \sum_{i=0}^{n} \sigma_i^k (-1)^i = r(\tau_k), \qquad v(1,t) = \sum_{i=0}^{n} \sigma_i^k \mathbb{L}_i^{\circledS}(1) = \sum_{i=0}^{n} \sigma_i^k = s(\tau_k).$$

To start the iterative method, we need the initial condition as

$$\sum_{i=0}^{n} \sigma_i^k \mathbb{L}_i^{\circledS}(x) = z(x),$$

where

$$\sigma_i^k = (2i+1) \int_0^1 \mathbb{L}_i^{\circledS}(x) z(x) dx.$$

Below, we briefly describe the step-by-step numerical algorithm.

Step 1. Generate the basis functions, $\{\mathbb{L}_\iota^{\circledS}(x)\}_{i=0}^N$, select the time step, $\delta\tau$, and produce a set of roots of $\mathbb{L}_\iota^{\circledS}(x)$,$\{x_i\}_{i=0}^N$, as the collocation points.

Step 2. Define the approximate fractional time derivative (2.1) and construct the matrix $A_{e,i}(x)$, the vectors matrix $B_{e,i}(x)$ and $Q(x)$ in relation (2.7).

Step 3. Compute the coefficient matrices, σ_i^k, in relation (2.7) with considering the initial and boundary conditions.

Step 4. Finally, by substituting these coefficients in relation (2.4), one estimates the approximate solution.

3. Study of the stability and convergence approach

This section investigates the order of convergence and the stability of the numerical method. Using the weak design (2.3) and defining the following function space in Hilbert space $L^2(\Omega)$ in Ω and standard norm $\|\theta(x)\|_2^2 = \langle \theta(x), \theta(x) \rangle$, we prove two theorems that describe the precision and efficiency of the numerical strategy explained in the earlier section.

$$H_\Omega^n(\theta) = \{\theta \in L^2(\Omega), D^\alpha \theta \in L^2(\Omega), \forall |\alpha| \leq n\},$$

where D^α is the fractional derivative.

Let $V^k(x)$ and $\overline{V}^k(x)$ be the exact and approximate solution of Eq (2.3), respectively. Multiplying $\varepsilon^k(x)$ and integrating on Ω in the relation (2.3) and denoting $\varepsilon^k(x) = \overline{V}^k(x) - V^k(x)$, we get the weak form of the relation (2.3) as

$$\begin{aligned} &\sum_{e=0}^{\epsilon} \frac{b_e \delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k,k}^{\alpha_e} \langle \varepsilon^k(x), \varepsilon^k(x) \rangle - K_1 \langle \frac{\partial^{2\beta_1} \varepsilon^k(x)}{\partial |x|^{2\beta_1}}, \varepsilon^k(x) \rangle - K_2 \langle \frac{\partial^{2\beta_2} \varepsilon^k(x)}{\partial |x|^{2\beta_2}}, \varepsilon^k(x) \rangle \\ &= \sum_{e=0}^{\epsilon} \sum_{j=0}^{k-1} \frac{b_e \delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k-1,j}^{\alpha_e} \langle \varepsilon^j(x), \varepsilon^k(x) \rangle. \end{aligned} \tag{3.1}$$

Before giving the stability and convergence of the weak scheme (3.1), we first give some lemmas.

Lemma 3.1. (See [39]) For $\forall f, g \in H_{\Omega}^{n}$ and $\forall x \in \mathbb{R}$, we have

$$\langle {}_xD_L^{\alpha} f(x), {}_xD_R^{\alpha} f(x)\rangle = \cos(\alpha\pi)\|{}_xD_L^{\alpha} f(x)\|^2 = \cos(\alpha\pi)\|{}_xD_R^{\alpha} f(x)\|^2, \quad \forall \alpha > 0,$$
$$\langle {}_xD_L^{\alpha} f(x), g(x)\rangle = \langle {}_xD_L^{\frac{\alpha}{2}} f(x), {}_xD_R^{\frac{\alpha}{2}} g(x)\rangle, \quad \langle {}_xD_R^{\alpha} f(x), g(x)\rangle = \langle {}_xD_R^{\frac{\alpha}{2}} f(x), {}_xD_L^{\frac{\alpha}{2}} g(x)\rangle, \quad \forall \alpha \in (1,2).$$

Lemma 3.2. For $\forall \varepsilon \in H_{\Omega}^{n}$, we have

$$\langle \frac{\partial^{2\beta_i}\varepsilon^k(x)}{\partial|x|^{2\beta_i}}, \varepsilon^k(x)\rangle = -\|{}_xD_L^{\beta_i}\varepsilon^k(x)\|^2, \quad \forall i = 1,2.$$

Proof. Using Lemma 3.1 and features of the interior product, we get

$$\begin{aligned}
\langle \frac{\partial^{2\beta_i}\varepsilon^k(x)}{\partial|x|^{2\beta_i}}, \varepsilon^k(x)\rangle &= \langle \frac{-1}{2\cos(\beta_i\pi)}\big({}_xD_L^{2\beta_i}\varepsilon^k(x) + {}_xD_R^{2\beta_i}\varepsilon^k(x)\big), \varepsilon^k(x)\rangle \\
&= \frac{-1}{2\cos(\beta_i\pi)}\Big(\langle {}_xD_L^{2\beta_i}\varepsilon^k(x), \varepsilon^k(x)\rangle + \langle {}_xD_R^{2\beta_i}\varepsilon^k(x), \varepsilon^k(x)\rangle\Big) \\
&= \frac{-1}{2\cos(\beta_i\pi)}\Big(\langle {}_xD_L^{\beta_i}\varepsilon^k(x), {}_xD_R^{\beta_i}\varepsilon^k(x)\rangle + \langle {}_xD_R^{\beta_i}\varepsilon^k(x), {}_xD_L^{\beta_i}\varepsilon^k(x)\rangle\Big) \\
&= -\|{}_xD_L^{\beta_i}\varepsilon^k(x)\|^2.
\end{aligned}$$

Theorem 3.1. When $0 < \alpha_e, \beta_i < 1$, $e = 0, 1, \ldots, \varepsilon$, $i = 1, 2$, the scheme (2.2) is unconditionally stable.

Proof. According to the Lemmas 3.1 and 3.2, the second and third terms of Eq (3.1) can be removed, then we gain

$$\sum_{e=0}^{\epsilon} \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)}\mathcal{S}_{k,k}^{\alpha_e}\langle \varepsilon^k(x), \varepsilon^k(x)\rangle \le \sum_{e=0}^{\epsilon}\sum_{j=0}^{k-1} \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)}\mathcal{S}_{k-1,j}^{\alpha_e}\langle \varepsilon^j(x), \varepsilon^k(x)\rangle,$$

and using the Cauchy Schwarz inequality, one get

$$\sum_{e=0}^{\epsilon} \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)}\mathcal{S}_{k,k}^{\alpha_e}\|\varepsilon^k(x)\| \le \sum_{e=0}^{\epsilon}\sum_{j=0}^{k-1} \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)}\mathcal{S}_{k-1,j}^{\alpha_e}\|\varepsilon^j(x)\|.$$

Since $\frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} > 0$ and $1 < \mathcal{S}_{k,k}^{\alpha_e} \le \frac{3}{2}$ from lemma of paper [40], we denote the above relation as form

$$\|\varepsilon^k(x)\| \le \epsilon\mathcal{S}_{k,k}^{\alpha_e}\|\varepsilon^k(x)\| \le \sum_{e=0}^{\epsilon}\mathcal{S}_{k,k}^{\alpha_e}\|\varepsilon^k(x)\| \le \sum_{e=0}^{\epsilon}\sum_{j=0}^{k-1}\Gamma_e\mathcal{S}_{k-1,j}^{\alpha_e}\|\varepsilon^j(x)\|,$$

where $\Gamma_e = \frac{b_e\delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)}$. Applying the theory of induction applies to all k, we know that the above relation satisfies as below:

$$\|\varepsilon^k(x)\| \le \sum_{e=0}^{\epsilon}\Gamma_e\sum_{j=0}^{k-1}\mathcal{S}_{k-1,j}^{\alpha_e}\|\varepsilon^0(x)\|.$$

Using the properties of the expansion coefficients in the lemma of paper [40] that describe the characteristics of approximation coefficient $1 < \mathcal{D}_{k,k} \leq \frac{3}{2}$ and $\sum_{j=0}^{k-1} \mathcal{S}_{k,j} = -\mathcal{D}_{k-1,k-1}$, it can be written as

$$\|\varepsilon^k(x)\| \leq \sum_{e=0}^{\epsilon} \Gamma_e |\mathcal{D}_{k,k}| \|\varepsilon^0(x)\| = c\|\varepsilon^0(x)\|,$$

where $c = \sum_{e=0}^{\epsilon} \Gamma_e |\mathcal{D}_{k,k}|$ is the positive constant. Then it shows that the discredited scheme (2.3) is unconditionally stable.

The convergence of the scheme (2.3) is given in the following theorem.

Theorem 3.2. Let $v^k(x) \in H_\Omega^2$ be the exact solution of (2.2), $V^k(x) \in H_\Omega^2$ be the solution of the semi-scheme (2.3). Then there exists a positive constant C such that

$$\|\xi^M(x)\| \leq CO(\delta\tau^{3-\max\alpha_e}).$$

Proof. Subtracting (2.2) from (2.3) and denoting $\xi^k(x) = v^k(x) - V^k(x), k = 1, 2, \ldots, M$, we can obtain that

$$\sum_{e=0}^{\epsilon} \frac{b_e \delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k,k}^{\alpha_e} \xi^k(x) - K_1 \frac{\partial^{2\beta_1} \xi^k(x)}{\partial |x|^{2\beta_1}} - K_2 \frac{\partial^{2\beta_2} \xi^k(x)}{\partial |x|^{2\beta_2}} = \sum_{e=0}^{\epsilon} \sum_{j=0}^{k-1} \frac{b_e \delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k-1,j}^{\alpha_e} \xi^j(x) + CR^k,$$

where $R^k = O(\delta\tau^{3-\max\alpha_e})$. From Lemmas 3.1 and 3.2, we know that the second and third terms of the above relation are negative, thus it can be rewritten as

$$\sum_{e=0}^{\epsilon} \frac{b_e \delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k,k}^{\alpha_e} \|\xi^k(x)\| \leq \sum_{e=0}^{\epsilon} \sum_{j=0}^{k-1} \frac{b_e \delta\tau^{-\alpha_e}}{\Gamma(2-\alpha_e)} \mathcal{S}_{k-1,j}^{\alpha_e} \|\xi^j(x)\| + C\|R^k\|.$$

By the method employed to prove the earlier theorem, we know that there exists a positive fixed term c such that

$$\|\xi^k(x)\| \leq c\|\xi^0(x)\| + C\|R^k\|.$$

Since $\|\xi^0(x)\| = 0$, then

$$\|\xi^M(x)\| \leq CO(\delta\tau^{3-\max\alpha_e}).$$

This concludes the proof.

4. Numerical experiments

We have conducted some experiments to demonstrate the effectiveness of our proposed methods. The numerical calculation is carried out on a Dell Inspiron personal computer equipped with an Intel (R) Core i72630QM 2.00GHz processor and 8GB of memory, using Wolfram Mathematica version 11. Our unique approach helped to address various issues, and we evaluated the maximum norm error L_∞-norm and L_2-norm error between the numerical solution and the exact solution at $T = 1$ as below:

$$L_\infty - norm = E_\infty(\delta h, \delta\tau) = \max_{0 \leq i \leq N} |v_i^M - \widehat{v}_i^M|, \qquad L_2 - norm = E_2(\delta h, \delta\tau) = \sqrt{\frac{1}{N} \sum_{i=0}^{N} |v_i^M - \widehat{v}_i^M|^2},$$

respectively. In this case, v_i^M is the exact solution and $\widehat{v}_i^M$ is the numerical solution with mesh step size δh and grid point (x_i, τ_j), where $i = 0, 1, \ldots, N$ and $j = 0, 1, \ldots, M$. Moreover, the convergence rate is calculated as follows:

$$Rate_\infty = \log_2(\frac{E_\infty(\delta h, 2\delta\tau)}{E_\infty(\delta h, \delta\tau)}), \quad Rate_2 = \log_2(\frac{E_2(\delta h, 2\delta\tau)}{E_2(\delta h, \delta\tau)}).$$

This formula measures how the error between the exact and numerical solutions decreases as the mesh step size δh is decreased. A higher convergence rate indicates that the numerical solution is converging to the exact solution more rapidly.

The numerical experiments conducted to evaluate the precision of the new approach for discretizing equations. Equation (1.1) is presented in the tables. The convergence rate $Rate_\infty$ and maximal norm error $E_\infty(\delta h, \delta\tau)$ are demonstrated for various mesh step sizes δh. The findings suggest that the innovative approach can achieve temporal convergence accuracy of $O(3 - max(\alpha_i))$, where $i = 1, 2$. Furthermore, the accuracy of the outcomes was compared to a previous study [33], and the proposed method demonstrated higher accuracy with lower errors than the method used in [33].

Example 4.1. Consider the following model with the diffusion coefficients are given $K_1 = K_2 = 2$ that the source term $q(x, \tau)$ is obtained from the exact solution $v(x, \tau) = \exp(-\tau)\sin(x)(x^2 - x + 1)$.

$${}_0^C D_\tau^{0.3} v(x, \tau) + {}_0^C D_\tau^{0.8} v(x, \tau) = 2\frac{\partial^{1.5} v(x, \tau)}{\partial |x|^{1.5}} + 2\frac{\partial^{1.7} v(x, \tau)}{\partial |x|^{1.7}} + q(x, \tau),\ 0 < x < 1,\ 0 < \tau < 1,$$

where the initial condition is $v(x, 0) = \sin(x)(x^2 - x + 1)$, and the boundary conditions are $v(0, \tau) = 0$ and $v(1, \tau) = \sin(1)$. The computing results of this model are shown in Table 1, wherein the maximal norm and E_2-norm errors have been provided; and the convergence rate has been found that it is equal to $O(3 - \max(\alpha_i))$, $i = 1, 2$ for all temporal discretization levels. From this table, we can see that the convergence rate is 2.2. Then the largest numbers of the parameters α_i are required to determine the convergence rate.

Table 1. Maximal norm, E_2-norm error and the convergence rate for Example 4.1 at $T = 1$.

$\delta\tau$	$E_\infty(\delta h, \delta\tau)$	$Rate_\infty$	$E_2(\delta h, \delta\tau)$	$Rate_2$
$\frac{1}{15}$	$4.67736E{-}4$	–	$1.84571E{-}3$	–
$\frac{1}{30}$	$1.02080E{-}4$	2.19600	$4.04321E{-}4$	2.19060
$\frac{1}{60}$	$2.22318E{-}5$	2.19900	$8.85524E{-}5$	2.19090
$\frac{1}{120}$	$4.83179E{-}6$	2.20199	$1.93902E{-}5$	2.1912.
$\frac{1}{240}$	$1.04796E{-}6$	2.20497	$4.24498E{-}6$	2.19150
$\frac{1}{480}$	$2.26841E{-}7$	2.20783	$9.29134E{-}7$	2.19180

Example 4.2. We first consider the one-dimensional Eq (1.1) in the domain $(x, \tau) \in [0, 1] \times [0, 1]$, with $T = 1, K_1 = K_2 = 1$, and source term

$$
\begin{aligned}
q(x,t) =& 200(x^2-x^3)\Big(\frac{t^{2-\alpha_0}}{\Gamma(3-\alpha_0)}+\frac{t^{2-\alpha_1}}{\Gamma(3-\alpha_1)}\Big)\\
&+\frac{50K_1(t^2+1)}{\cos(\beta_1\pi)}\Big(\frac{2(x^{2-2\beta_1}+(1-x)^{2-2\beta_1})}{\Gamma(3-2\beta_1)}-\frac{6(x^{3-2\beta_1}+(1-x)^{3-2\beta_1})}{\Gamma(4-2\beta_1)}\Big)\\
&+\frac{50K_2(t^2+1)}{\cos(\beta_2\pi)}\Big(\frac{2(x^{2-2\beta_2}+(1-x)^{2-2\beta_2})}{\Gamma(3-2\beta_2)}-\frac{6(x^{3-2\beta_2}+(1-x)^{3-2\beta_2})}{\Gamma(4-2\beta_2)}\Big).
\end{aligned}
$$

To be specific, the initial condition is $v(x,0) = 100(x^2 - x^3)$, and the boundary conditions are $v(0,\tau) = v(1,\tau) = 0$. The precise solution for this issue can be expressed as $v(x,\tau) = 100(1+\tau^2)(x^2 - x^3)$. We have carried out numerical experiments and presented the results in Tables 2 and 3. Table 2 shows the maximum norm and $E_2 - norm$ errors for the new method, which uses parameters $\beta_1 = 0.2$ and $\beta_2 = 0.7$. Based on our analysis, we have determined that the convergence rate can be expressed as $O(3 - max(\alpha_i))$.

Table 3 shows cases of a comparison between the error and convergence order of our novel method and the technique outlined in [33]. We used parameters $\alpha_0 = 0.5,\ \alpha_1 = 0.2,\ \beta_0 = 0.3$ and $\beta_1 = 0.8$. Our results demonstrate that our new method is more efficient than the method presented in [33]. When $\alpha_0 = 0.5$, the convergence order for our new method is 2.5, while the convergence order for the method presented in [33] is only 1.5. Additionally, the error of our new method is significantly lower than the error of the method presented in [33]. In conclusion, our new method outperforms the method presented in [33] in terms of efficiency and accuracy.

Table 2. Maximal norm, E_2-norm error and the convergence rate with $\beta_1 = 0.2,\ \beta_2 = 0.7$ and $N = 5$ for Example 4.2 at $T = 1$.

	with $\alpha_1 = \alpha_2 = 0.5$				with $\alpha_1 = 0.7, \alpha_2 = 0.3$			
$\delta\tau$	$E_\infty(\delta h,\delta\tau)$	$Rate_\infty$	$E_2(\delta h,\delta\tau)$	$Rate_2$	$E_\infty(\delta h,\delta\tau)$	$Rate_\infty$	$E_2(\delta h,\delta\tau)$	$Rate_2$
$\frac{1}{20}$	$2.88003E{-3}$	–	$8.36760E{-3}$	–	$4.65992E{-3}$	–	$1.35337E{-2}$	–
$\frac{1}{40}$	$5.04560E{-4}$	2.51299	$1.46594E{-3}$	2.51298	$9.12647E{-4}$	2.35218	$2.65059E{-3}$	2.35217
$\frac{1}{80}$	$8.88040E{-5}$	2.50633	$2.58011E{-4}$	2.50633	$1.80816E{-4}$	2.33553	$5.25143E{-5}$	2.33553
$\frac{1}{160}$	$1.56647E{-5}$	2.50311	$4.55123E{-5}$	2.50310	$3.60866E{-5}$	2.32499	$1.04806E{-6}$	2.32499
$\frac{1}{320}$	$2.76789E{-6}$	2.50066	$8.04183E{-6}$	2.50066	$7.23087E{-6}$	2.31926	$2.10001E{-5}$	2.31926

Table 3. Comparing the error and convergence order of method [33] with the new method for Example 4.2 with $\alpha_0 = 0.5,\ \alpha_1 = 0.2,\ \beta_0 = 0.3$ and $\beta_1 = 0.8$.

	Error and the convergence order of paper [33]		Error and the convergence order of the new method with $N = 5$			
$\delta\tau$	$E_\infty(\delta h,\delta\tau)$	$Rate_\infty$	$E_\infty(\delta h,\delta\tau)$	$Rate_\infty$	$E_2(\delta h,\delta\tau)$	$Rate_2$
$\frac{1}{64}$	$2.9976E{-2}$	–	$3.91334E{-5}$	–	$1.08214E{-4}$	–
$\frac{1}{128}$	$9.4587E{-3}$	1.6641	$6.72919E{-6}$	2.53990	$1.86054E{-5}$	2.54009
$\frac{1}{256}$	$3.1683E{-3}$	1.5779	$1.16248E{-6}$	2.53323	$3.21374E{-6}$	2.53340
$\frac{1}{512}$	$1.5910E{-3}$	1.5435	$1.80841E{-7}$	2.68442	$4.99025E{-7}$	2.68707
$\frac{1}{1024}$	$8.3916E{-4}$	1.5777	$1.11978E{-8}$	2.64325	$3.13224E{-8}$	2.65001

The stiffness matrix's condition numbers obtained for space partitions $N = 5$ and $N = 7$ with different values of β_1 and β_2 are shown in Figure 1. The two curves indicate that the new algorithm is robust in discretizing the equation to obtain the resulting systems. The condition numbers decrease as the number of discretization points increases. In Figure 2, the absolute error is shown with $N = 5$, $\alpha_1 = 0.5$, $\alpha_2 = \beta_1 = 0.2$ and $\beta_2 = 0.7$ parameters, while M values are varied. As the number of discretization points increases, the absolute error decreases. In conclusion, the new algorithm effectively discretizes the equation and the resulting systems. When you increase the number of discretization points, it leads to a decrease in both the condition numbers and absolute error.

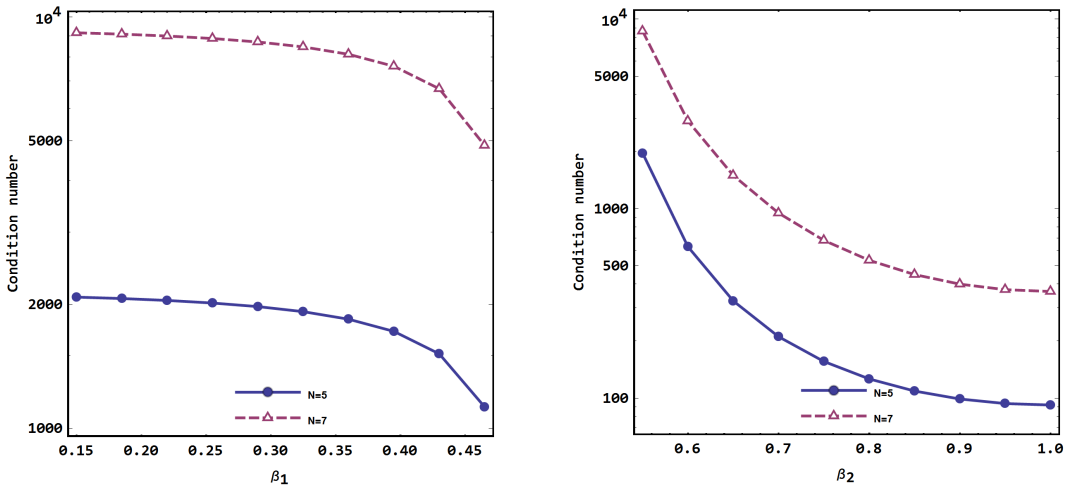

Figure 1. Condition numbers of the stiffness matrix obtained for the space partitions $N = 5$ and $N = 7$ with the various values β_1 (left panel) and β_2 (right panel) for Example 4.2.

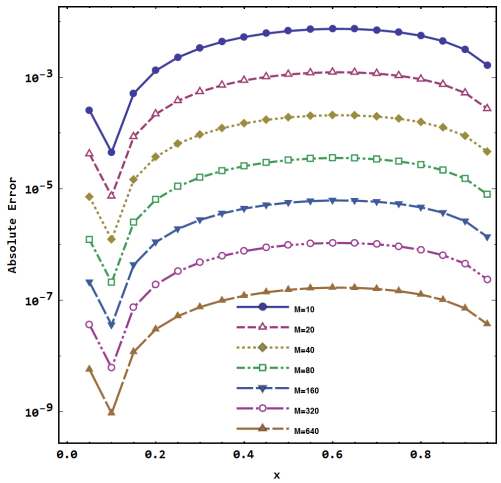

Figure 2. Absolute error for the first example with parameter $N = 5$, $\alpha_1 = 0.5$, $\alpha_2 = \beta_1 = 0.2$ and $\beta_2 = 0.7$.

Example 4.3. The second example is similar to the first one, with the same accurate resolution of $v(x, \tau) = 100(1 + \tau^2)x^2(1 - x)^2$ and diffusion coefficients $K_1 = K_2 = 1$. The source term for this example is

$$
\begin{aligned}
q(x,t) =& 200x^2(1-x)^2\Big(\frac{t^{2-\alpha_0}}{\Gamma(3-\alpha_0)} + \frac{t^{2-\alpha_1}}{\Gamma(3-\alpha_1)}\Big) + \frac{100K_1(t^2+1)}{\cos(\beta_1\pi)}\Big(\frac{(x^{2-2\beta_1}+(1-x)^{2-2\beta_1})}{\Gamma(3-2\beta_1)} \\
&- \frac{6(x^{3-2\beta_1}+(1-x)^{3-2\beta_1})}{\Gamma(4-2\beta_1)} + \frac{12(x^{4-2\beta_1}+12(1-x)^{4-2\beta_1})}{\Gamma(5-2\beta_1)}\Big) \\
&+ \frac{100K_2(t^2+1)}{\cos(\beta_2\pi)}\Big(\frac{(x^{2-2\beta_2}+(1-x)^{2-2\beta_2})}{\Gamma(3-2\beta_2)} - \frac{6(x^{3-2\beta_2}+(1-x)^{3-2\beta_2})}{\Gamma(4-2\beta_2)} + \frac{12(x^{4-2\beta_2}+(1-x)^{4-2\beta_2})}{\Gamma(5-2\beta_2)}\Big).
\end{aligned}
$$

Tables 4 and 5 present the results of numerical experiments conducted on a new method. In Table 4, wherein the maximal norm and E_2-norm errors have been provided with parameters $\beta_1 = 0.45$, $\beta_2 = 0.7$, and the convergence rate has been found that it is equal to $O(3-\max(\alpha_i))$, $i = 1, 2$ for all temporal discretization levels. From this table, we can see that the convergence rate is 2.5 when $\alpha_1 = \alpha_2 = 0.5$ but it is 2.35 for $\alpha_1 = 0.2$, $\alpha_2 = 0.65$. Then, the largest numbers of the parameters α_i are required to determine the convergence rate. Comparing the error and convergence order of the method [33] with the new method is represented in Table 5 with the parameters $\alpha_0 = 0.7$, $\alpha_1 = 0.4$, $\beta_0 = 0.3$ and $\beta_1 = 0.85$. From this table, we can see that the proposed method is much more efficient than method of [33]. Moreover, the error of the new method is much less compared to the paper of [33].

Table 4. Maximal norm, E_2-norm error and convergence rate with $\beta_1 = 0.45$, $\beta_2 = 0.7$ and $N = 5$ for Example 4.3 at $T = 1$.

	with $\alpha_1 = 0.2, \alpha_2 = 0.65$				with $\alpha_1 = \alpha_2 = 0.5$			
$\delta\tau$	$E_\infty(\delta h, \delta\tau)$	$Rate_\infty$	$E_2(\delta h, \delta\tau)$	$Rate_2$	$E_\infty(\delta h, \delta\tau)$	$Rate_\infty$	$E_2(\delta h, \delta\tau)$	$Rate_2$
$\frac{1}{10}$	$1.30566E{-3}$	–	$4.49206E{-7}$	–	$1.52349E{-4}$	–	$4.64827E{-4}$	–
$\frac{1}{20}$	$2.20088E{-5}$	2.56862	$2.30350E{-6}$	2.56742	$2.31367E{-5}$	2.71912	$7.06426E{-5}$	2.71808
$\frac{1}{40}$	$3.97720E{-6}$	2.46826	$1.21600E{-5}$	2.46729	$3.68642E{-6}$	2.64989	$1.12627E{-5}$	2.64899
$\frac{1}{80}$	$7.53043E{-7}$	2.40095	$6.72457E{-5}$	2.40025	$6.06790E{-7}$	2.60295	$1.85482E{-6}$	2.60220
$\frac{1}{160}$	$1.46802E{-7}$	2.35886	$3.98597E{-4}$	2.35838	$1.02142E{-7}$	2.57063	$3.12350E{-7}$	2.57004

Table 5. Comparing the error and the CPU time of the method [33] with the new method for example 4.3 with $\alpha_0 = 0.7$, $\alpha_1 = 0.4$, $\beta_0 = 0.3$ and $\beta_1 = 0.85$.

	Error and the CPU time of paper [33]				Error and the CPU time *of the new method with* $N = 5$		
$\delta\tau$	$E_\infty(2\delta\tau, \delta\tau)$	$CPU(s)$			$E_\infty(\delta h, \delta\tau)$	$CPU(s)$	$Rate_\infty$
		Gauss	CGNR	MG		*New method*	
$\frac{1}{128}$	$1.9108E{-3}$	0.02	0.02	0.04	$1.98616E{-5}$	0.0189	–
$\frac{1}{256}$	$4.7880E{-4}$	0.24	0.38	0.19	$3.36372E{-6}$	0.2154	2.56186
$\frac{1}{512}$	$1.2254E{-5}$	3.19	3.14	0.84	$6.06274E{-7}$	1.0259	2.47202
$\frac{1}{1024}$	$2.7918E{-5}$	40.15	34.59	4.20	$1.13894E{-7}$	3.2468	2.41227
$\frac{1}{2048}$	$8.5361E{-6}$	698.80	693.64	22.54	$2.20280E{-8}$	8.5496	2.37028

Figure 3 illustrates the absolute error at different values of M, with parameters $N = 5$, $\alpha_1 = 0.3$, $\alpha_2 = 0.6$, $\beta_1 = 0.4$ and $\beta_2 = 0.8$. The figure shows that increasing the number of discretization points decreases the absolute error. In conclusion, the new technique is more efficient and accurate than the method from [33].

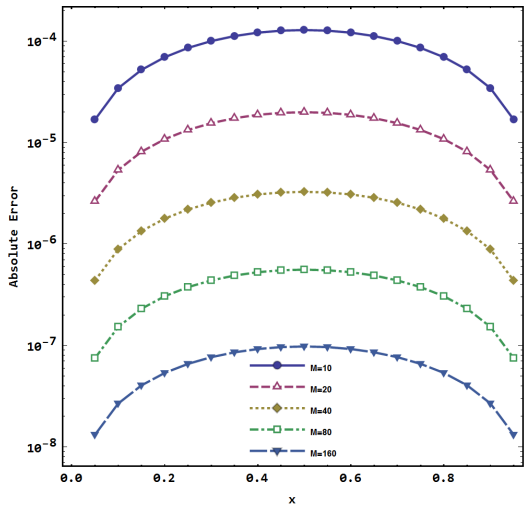

Figure 3. Absolute error with parameters $N = 5$, $\alpha_1 = 0.3$, $\alpha_2 = 0.6$, $\beta_1 = 0.4$ and $\beta_2 = 0.8$ for Example 4.3.

5. Conclusions

In this study, we have introduced a novel approach for solving multi-term time-fractional advection-diffusion equations that utilizes a square interpolation method to discretize the temporal fractional derivative and a spectral method and employs Legendre polynomials to discretize the spatial derivatives. We then construct a matrix-approximate preconditioner to accelerate the solution of the discretized system. Our approach has been proven to be stable without any conditions and has a convergence rate of $O(3 - max(\alpha_i))$. To the best of our knowledge, the theoretical proof obtained from this study represents the first results for these numerical schemes. We also presented experimental results that demonstrate the effectiveness of our proposed method in comparison with the methods shown in [33]. However, the proposed method is inherently a limitation, wherein the coefficient matrix of the linear system obtained from the discretization methods is not a sparse matrix, and this may increase a significant amount of memory and slow down the processing of that data for a large matrix of three-dimensional and large scale problems. In the future, we aim to expand our proposed approach to address variable coefficients in higher-order fractional differential equations and also plan to develop some fast numerical methods for these equations.

Use of AI tools declaration

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The authors would like to thank the support from the National Research Foundation of Korea under grant number NRF-2020K1A3A1A05101625 and from the research support for foreign professors of the College of Engineering at Seoul National University.

Conflict of interest

The authors declare no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

1. S. Chandrasekar, Stochastic problems in physics and astronomy, *Rev. Mod. Phys.*, **15** (1943), 1–89. https://doi.org/10.1103/RevModPhys.15.1
2. C. E. Baukal Jr., V. Y. Gershtein, X. M. Li, *Computational fluid dynamics in industrial combustion*, CRC press, 2000.
3. M. Dehghan, Weighted finite difference techniques for the one-dimensional advection-diffusion equation, *Appl. Math. Comput.*, **147** (2004), 307–319. https://doi.org/10.1016/S0096-3003(02)00667-7
4. J. D. Seymour, J. P. Gage, S. L. Codd, R. Gerlach, Anomalous fluid transport in porous media induced by biofilm growth, *Phys. Rev. Lett.*, **93** (2004), 198103. https://doi.org/10.1103/PhysRevLett.93.198103
5. J. P. Bouchaud, A. Georges, Anomalous diffusion in disordered media: Statistical mechanisms, models and physical applications, *Phys. Rep.*, **195** (1990), 127–293. https://doi.org/10.1016/0370-1573(90)90099-N
6. G. A. Maugin, On the thermomechanics of continuous media with diffusion and/or weak nonlocality, *Arch. Appl. Mech.*, **75** (2006), 723–738. https://doi.org/10.1007/s00419-006-0062-4
7. H. D. Qu, X. Liu, X. Lu, M. ur Rahman, Z. H. She, Neural network method for solving nonlinear fractional advection-diffusion equation with spatiotemporal variable-order, *Chaos Solitons Fract.*, **156** (2022), 111856. https://doi.org/10.1016/j.chaos.2022.111856
8. X. W. Jiang, J. H. Li, B. Li, W. Yin, L. Sun, X. Y. Chen, Bifurcation, chaos, and circuit realisation of a new four-dimensional memristor system, *Int. J. Nonlinear Sci. Numer. Simul.*, 2022. https://doi.org/10.1515/ijnsns-2021-0393
9. J. Li, Y. L. Cheng, Barycentric rational interpolation method for solving time-dependent fractional convection-diffusion equation, *Electron. Res. Arch.*, **31** (2023), 4034–4056. https://doi.org/10.3934/era.2023205
10. A. Atangana, A. Kılıçman, The use of Sumudu transform for solving certain nonlinear fractional heat-like equations, *Abstr. Appl. Anal.*, **2013** (2013), 1–12. https://doi.org/10.1155/2013/737481
11. E. Barkai, R. Metzler, J. Klafter, From continuous time random walks to the fractional Fokker-Planck equation, *Phys. Rev. E*, **61** (2000), 132–138. https://doi.org/10.1103/PhysRevE.61.132

12. A. Blumen, G. Zumofen, J. Klafter, Transport aspects in anomalous diffusion: Lévy walks, *Phys. Rev. A*, **40** (1989), 3964–3973. https://doi.org/10.1103/PhysRevA.40.3964
13. J. P. Bouchaud, A. Georges, Anomalous diffusion in disordered media: Statistical mechanisms, models and physical applications, *Phys. Rep.*, **195** (1990), 127–293. https://doi.org/10.1016/0370-1573(90)90099-N
14. A. S. Chaves, A fractional diffusion equation to describe Lévy flights, *Phys. Lett. A*, **239** (1998), 13–16. https://doi.org/10.1016/S0375-9601(97)00947-X
15. J. Klafter, A. Blumen, M. F. Shlesinger, Stochastic pathway to anomalous diffusion, *Phys. Rev. A*, **35** (1987), 3081–3085. https://doi.org/10.1103/PhysRevA.35.3081
16. B. Baeumer, M. M. Meerschaert, Stochastic solutions for fractional Cauchy problems, *Fract. Calc. Appl. Anal.*, **4** (2001), 481–500.
17. M. M. Meerschaert, D. A. Benson, H. P. Scheffler, B. Baeumer, Stochastic solution of space-time fractional diffusion equations, *Phys. Rev. E*, **65** (2002), 041103. https://doi.org/10.1103/PhysRevE.65.041103
18. A. I. Saichev, G. M. Zaslavsky, Fractional kinetic equations: Solutions and applications, *Chaos*, **7** (1997), 753–764. https://doi.org/10.1063/1.166272
19. G. M. Zaslavsky, Fractional kinetic equation for Hamiltonian chaos, *Phys. D*, **76** (1994), 110–122. https://doi.org/10.1016/0167-2789(94)90254-2
20. B. Baeumer, D. A. Benson, M. M. Meerschaert, S. W. Wheatcraft, Subordinated advection-dispersion equation for contaminant transport, *Water Resour. Res.*, **37** (2001), 1543–1550. https://doi.org/10.1029/2000WR900409
21. D. A. Benson, R. Schumer, M. M. Meerschaert, S. W. Wheatcraft, Fractional dispersion, Lévy motion, and the MADE tracer tests, *Transport Porous Med.*, **42** (2001), 211–240. https://doi.org/10.1023/A:1006733002131
22. D. A. Benson, S. W. Wheatcraft, M. M. Meerschaert, The fractional-order governing equation of Lévy motion, *Water Resour. Res.*, **36** (2000), 1413–1423. https://doi.org/10.1029/2000WR900032
23. R. Schumer, D. A. Benson, M. M. Meerschaert, B. Baeumer, Multiscaling fractional advection-dispersion equations and their solutions, *Water Resour. Res.*, **39** (2003), 1022–1032. https://doi.org/10.1029/2001WR001229
24. Y. Povstenko, Generalized boundary conditions for the time-fractional advection diffusion equation, *Entropy*, **17** (2015), 4028–4039. https://doi.org/10.3390/e17064028
25. Q. Rubbab, I. A. Mirza, M. Z. A. Qureshi, Analytical solutions to the fractional advection-diffusion equation with time-dependent pulses on the boundary, *AIP Adv.*, **6** (2016), 075318. https://doi.org/10.1063/1.4960108
26. F. S. Md Nasrudin, C. Phang, A. Kanwal, Fractal-fractional advection-diffusion-reaction equations by Ritz approximation approach, *Open Phys.*, **21** (2023), 20220221. https://doi.org/10.1515/phys-2022-0221
27. A. A. El-Sayed, P. Agarwal, Numerical solution of multiterm variable-order fractional differential equations via shifted Legendre polynomials, *Math. Methods Appl. Sci.*, **42** (2019), 3978–3991. https://doi.org/10.1002/mma.5627

28. Q. Yang, F. Liu, I. Turner, Numerical methods for fractional partial differential equations with Riesz space fractional derivatives, *Appl. Math. Model.*, **34** (2010), 200–218. https://doi.org/10.1016/j.apm.2009.04.006
29. S. J. Shen, F. W. Liu, V. Anh, Numerical approximations and solution techniques for the space-time Riesz-Caputo fractional advection-diffusion equation, *Numer. Algorithms*, **56** (2011), 383–403. https://doi.org/10.1007/s11075-010-9393-x
30. F. W. Liu, M. M. Meerschaert, R. J. McGough, P. H. Zhuang, Q. X. Liu, Numerical methods for solving the multi-term time-fractional wave-diffusion equation, *Fract. Calc. Appl. Anal.*, **16** (2013), 9–25. https://doi.org/10.2478/s13540-013-0002-2
31. A. V. Pskhu, Multi-time fractional diffusion equation, *Eur. Phys. J. Spec. Top.*, **222** (2013), 1939–1950. https://doi.org/10.1140/epjst/e2013-01975-y
32. M. Garg, P. Manohar, S. L. Kalla, Generalized differential transform method to space-time fractional telegraph equation, *Int. J. Differ. Equ.*, **2011** (2011), 1–9. https://doi.org/10.1155/2011/548982
33. W. P. Bu, X. T. Liu, Y. F. Tang, J. Y. Yang, Finite element multigrid method for multi-term time fractional advection diffusion equations, *Int. J. Model. Simul. Sci. Comput.*, **6** (2015), 1540001. https://doi.org/10.1142/S1793962315400012
34. H. Jiang, F. Liu, I. Turner, K. Burrage, Analytical solutions for the multi-term time-space Caputo-Riesz fractional advection-diffusion equations on a finite domain, *J. Math. Anal. Appl.*, **389** (2012), 1117–1127. https://doi.org/10.1016/j.jmaa.2011.12.055
35. V. V. Uchaĭkin, R. Sibatov, *Fractional kinetics in solids: Anomalous charge transport in semiconductors, dielectrics, and nanosystems*, World Scientific, 2013.
36. X. L. Ding, Y. L. Jiang, Analytical solutions for the multi-term time-space fractional advection-diffusion equations with mixed boundary conditions, *Nonlinear Anal. Real World Appl.*, **14** (2013), 1026–1033. https://doi.org/10.1016/j.nonrwa.2012.08.014
37. K. Kumar, R. K. Pandey, S. Sharma, Comparative study of three numerical schemes for fractional integro-differential equations, *J. Comput. Appl. Math.*, **315** (2017), 287–302. https://doi.org/10.1016/j.cam.2016.11.013
38. H. Mesgarani, A. Adl, Y. E. Aghdam, Approximate price of the option under discretization by applying quadratic interpolation and Legendre polynomials, *Math. Sci.*, **17** (2023), 51–58. https://doi.org/10.1007/s40096-021-00439-9
39. V. J. Ervin, J. P. Roop, Variational solution of fractional advection dispersion equations on bounded domains in $\mathbb{R}^d$, *Numer. Methods Partial Differ. Equ.*, **23** (2007), 256–281. https://doi.org/10.1002/num.20169
40. Y. E. Aghdam, H. Mesgarani, G. M. Moremedi, M. Khoshkhahtinat, High-accuracy numerical scheme for solving the space-time fractional advection-diffusion equation with convergence analysis, *Alex. Eng. J.*, **61** (2022), 217–225. https://doi.org/10.1016/j.aej.2021.04.092

AIMS Press

© 2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0)