Filomat 38:32 (2024), 11453–11475
https://doi.org/10.2298/FIL2432453A

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

Optimal collocation method for time-fractional advection-dispersion equation using modified generalized Laguerre polynomials and particle swarm optimization algorithm

Alaa Hussein Braz Albohiwela[a], Safar Irandoust-pakchin[a,*], Aliasghar Jodayree Akbarfam[a]

[a] *Department of Applied Mathematics, Faculty of Mathematics, Statistics and Computer Sciences, University of Tabriz, Tabriz, Iran*

Abstract. In this article, time-fractional advection-dispersion equation is considered. Fractional derivative is in the Caputo sense and for approximating the first and second derivatives, the modified generalized Laguerre polynomials (*MGLPs*) have been used. The *MGLPs* ($L_n^{\alpha,\beta}(x)$) have two parameter $\alpha > -1$ and $\beta > 0$. These polynomials, orthogonal polynomial on the $[0, \infty)$ by weight function $\omega^{\alpha,\beta}(x) = x^{\alpha}e^{-\beta x}$. For solving these time-fractional advection-dispersion equation, we introduce optimal *MGLPs* collocation method. The error of the proposed algorithm depends on the parameters of α and β. The (*PSO*) algorithm is used to find the optimal parameters so that the error is minimized. In other words, we try to find the optimal value for parameters α and β so that the error of the method is minimized. In practice, since the exact solution is not available, we will face a problem to measure the error. To overcome this problem, the best parameters for approximating the function that describes the initial condition with the help of *PSO* algorithm are found and these parameters are used to better approximate the exact solution of the problem. A few numerical experiments are carried out to support the theoretical claims. The presented examples confirm that the optimal parameters for the initial condition can reduce the error of the method.

1. Introduction

In the last few decades, fractional differential equations have been studied by many researchers in some of physical phenomena and numerous areas. Due to the ability and flexibility of fractional partial differential equations (FPDEs) for describing scientific phenomena has made them a very good tool[1–3]. A fractional advection–dispersion equation (*FADE*) is a generalization of the classical (*ADE*) in which the order of derivatives is replaced with a fractional-order derivative.

In the last decade, the advection–dispersion equation in the fractional form has attracted the attention of researchers. Some of them are focused on solving the time fractional advection-dispersion equation(*TFADE*) [10, 13, 16, 18–21, 28, 29], others on space fractional advection-dispersion equation (*SFADE*) [11, 12, 26], and some of them are focused on the fractional time-space advection–dispersion equation(*TSFADE*)[23, 24, 27].

2020 *Mathematics Subject Classification*. Primary 26A33; Secondary 33E12, 34A08, 34K37, 35R11, 60G22.
Keywords. Advection-dispersion equation; Caputo fractional derivative; Modified generalized Laguerre polynomials; Particle swarm optimization algorithm.
Received: 11 September 2024; Accepted: 11 September 2024
Communicated by Maria Alessandra Ragusa
* Corresponding author: Safar Irandoust-pakchin
Email addresses: AlaaHussein6211@yahoo.com (Alaa Hussein Braz Albohiwela), s.irandoust@tabrizu.ac.ir (Safar Irandoust-pakchin), akbarfam@tabrizu.ac.ir (Aliasghar Jodayree Akbarfam)

To solve the equation (*FADE*), several methods have been used by researchers. Cao used the variable weights particle tracking method[29], Singh used the homotopy analysis method[28] and homotopy perturbation method[27], Saw using collocation method[26] and the fourth kind shifted chebyshev polynomials[11], Moghadam used radial basis functions[23], spline approximation method is used by Ravi Kanth[20**?**], Jacobi collocation method is used by Singh[22] and an RBF based meshless method is used by Bhardwaj[19].

In this article, the time-fractional advection–dispersion equation is considered in the bounded space-time variables as

$$\begin{cases} \frac{\partial^{\nu} u(x,t)}{\partial t^{\nu}} = \lambda \frac{\partial^2 u(x,t)}{\partial x^2} - \mu \frac{\partial u(x,t)}{\partial x} + f(x,t), \quad 0 \le t, \quad 0 \le x \le 1, \quad 0 < \nu \le 1, \\ B.C. : u(0,t) = 0, \quad u(1,t) = 0, \quad 0 \le t \le 1, \\ I.C : u(x,0) = g(x), \quad 0 \le x \le 1, \end{cases} \tag{1.1}$$

where the term $f(x,t)$ is known as the source term, λ and μ represent the dispersion coefficient and the average fluid velocity, respectively, and the fractional derivative is in the Caputo sense.

To approximate in the space domain, the modified generalized Laguerre polynomials (*MGLPs*) is used. In order to be able to use orthogonality of *MGLPs*, we assume that the function $g(x)$ is defined in the interval $x \in [0, \infty)$.

The paper is organized as follows. In Section 2, we give some definitions, notations and properties of modified generalized Laguerre polynomials. In Section 3, a numerical method based on *MGLPs* is presented. Then in Section 4, we give the analysis of stability and error estimates for the presented method. In Section 5, some numerical experiments for the introduced method are carried out.

2. preliminary

In this section, the required preliminary concepts are presented. First, the fractional derivative and some of its properties are presented, and then the important properties of Laguerre polynomials are studied. From a few decades ago until today, various definitions for fractional derivatives have been proposed. Some of them are limit based and others are defined based on integral. The fractional derivative presented by Caputo is still of interest to researchers, which is defined as follows:

Definition 2.1. *Caputo's fractional derivative of function* $f(t) \in C^{\lceil \nu \rceil}(\mathbb{R})$ *is defined as follows*

$${}_a^C D_t^{\nu} f(t) = \frac{1}{\Gamma(n-\nu)} \int_a^t \frac{f^{(n)}(\tau)}{(t-\tau)^{\nu+1-n}} d\tau, \quad n-1 < \nu < n, \tag{2.1}$$

Under natural conditions on the function $f(t)$, for $\nu \to n$ the Caputo derivative becomes a convctional n^{th} derivative of the function $f(t)$.[4] that is

$$\lim_{\nu \to n} {}_a^C D_t^{\nu} f(t) = f^{(n)}(t)$$

note that, ${}_a^C D_t^{\nu} C = 0$, where C is a constant and for $f(t) = t^{\eta}$, $\eta \in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$ we have

$${}_a^C D_t^{\nu} t^{\eta} = \begin{cases} \frac{\Gamma(\eta+1)}{\Gamma(\eta-\nu+1)} t^{\eta-\nu}, & \eta \in \mathbb{N}_0 \ \ and \ \ \eta \ge \lceil \nu \rceil, \\ 0 \qquad \eta \in \mathbb{N}_0 \ \ and \ \ \eta \le \lceil \nu \rceil. \end{cases} \tag{2.2}$$

Since we have $n = 1$ for the fractional derivative used in problem (1.1), so for simplicity, instead of writing ${}_0^C D_t^{\nu} u(x,t)$, we use the symbol $\partial_t^{\nu} u(x,t)$ for Caputo's fractional derivative. With this notation, we will have

$$\begin{cases} \partial_t^{\nu} u(x,t) = \lambda u_{xx}(x,t) - \mu u_x(x,t) + f(x,t), \ 0 \le t, \ 0 \le x \le 1, \ 0 < \nu \le 1, \\ B.C. : u(0,t) = 0, \quad u(1,t) = 0, \quad 0 \le t \le 1, \\ I.C : u(x,0) = g(x), \quad 0 \le x \le 1, \end{cases} \tag{2.3}$$

2.1. Modified generalized Laguerre polynomials

Consider the classical Laguerre differential equation

$$xy'' + (1 - x)y' + ny = 0. \tag{2.4}$$

Classical Laguerre polynomials(*CLPs*) are the solutions of the classical Laguerre equation(2.4). These polynomials are given by Rodrigues formula as

$$L_n(x) = \frac{1}{n!}e^x \partial_x^n (x^n e^{-x}), \tag{2.5}$$

and in the[42]the generalized Laguerre differential equation given by

$$xy'' + (\alpha + 1 - x)y' + ny = 0, \tag{2.6}$$

generalized Laguerre polynomials(*GLPs*) are the solutions of the generalized Laguerre equation(2.6). These polynomials are given by Rodrigues formula as

$$L_n^{(\alpha)}(x) = \frac{1}{n!}x^{-\alpha}e^x \partial_x^n (x^{n+\alpha} e^{-x}). \tag{2.7}$$

Finally, modified generalized Laguerre polynomials (*MGLPs*) are generated by [43]

$$L_n^{(\alpha,\beta)}(x) = \frac{1}{n!}x^{-\alpha}e^{\beta x} \partial_x^n (x^{n+\alpha} e^{-\beta x}), \tag{2.8}$$

they are the eigenfunctions of the Sturm–Liouville problem

$$(\omega_{\alpha+1,\beta}(x)y')' + (\beta n)\omega_{\alpha,\beta}(x)y = 0. \tag{2.9}$$

Also they can be generated by using the following recursive relation [37]

$$\begin{aligned} L_0^{(\alpha,\beta)}(x) &= 1, \\ L_1^{(\alpha,\beta)}(x) &= 1 + \alpha - \beta x, \\ L_{n+1}^{(\alpha,\beta)}(x) &= \frac{2n + 1 + \alpha - \beta x}{n + 1} L_n^{(\alpha,\beta)}(x) - \frac{n + \alpha}{n + 1} L_{n-1}^{(\alpha,\beta)}(x). \end{aligned} \tag{2.10}$$

one can write, the n-th degree *MGLPs* in the analytical form

$$L_n^{(\alpha,\beta)}(x) = \sum_{k=0}^{n} (-1)^k \frac{\beta^k \Gamma(n + \alpha + 1)}{\Gamma(k + \alpha + 1)(n - k)!k!} x^k, \tag{2.11}$$

and in the special case $\alpha = 0$ we have

$$L_n^{(0,\beta)}(x) = \sum_{k=0}^{n} (-1)^k \binom{n}{k} \frac{(\beta x)^k}{k!}. \tag{2.12}$$

Some other properties of the analytical form of *MGLPs* are presented as the following statements [37]

$$L_n^{(\alpha,\beta)}(0) = \frac{\Gamma(n + \alpha + 1)}{\Gamma(\alpha + 1)\Gamma(n + 1)}, \tag{2.13}$$

$$\frac{d^i}{dx^i}L_n^{(\alpha,\beta)}(x)=\begin{cases}\sum_{k=0}^{n}(-1)^k\frac{\beta^k\Gamma(n+\alpha+1)}{\Gamma(k+\alpha+1)\Gamma(k-i+1)(n-k)!}x^{k-i} & i\le n\\ 0 & i>n\end{cases}, \tag{2.14}$$

$$\frac{d}{dx}L_n^{(\alpha,\beta)}(x)=\frac{d}{dx}L_{n-1}^{(\alpha,\beta)}(x)-\beta L_{n-1}^{(\alpha,\beta)}(x). \tag{2.15}$$

It is possible to use property (2.15) and obtain the following important relationship

$$\begin{aligned}\frac{d}{dx}L_n^{(\alpha,\beta)}(x)&=\frac{d}{dx}L_{n-1}^{(\alpha,\beta)}(x)-\beta L_{n-1}^{(\alpha,\beta)}(x)\\&=\frac{d}{dx}L_{n-2}^{(\alpha,\beta)}(x)-\beta L_{n-2}^{(\alpha,\beta)}(x)-\beta L_{n-1}^{(\alpha,\beta)}(x)\\&=\frac{d}{dx}L_0^{(\alpha,\beta)}(x)-\beta L_0^{(\alpha,\beta)}(x)-\beta L_1^{(\alpha,\beta)}(x)-...-\beta L_{n-1}^{(\alpha,\beta)}(x)\\&=-\beta(L_0^{(\alpha,\beta)}(x)+L_1^{(\alpha,\beta)}(x)+...+L_{n-1}^{(\alpha,\beta)}(x)).\end{aligned} \tag{2.16}$$

Let's use the following notation

$$\mathcal{L}(x)=[L_0^{(\alpha,\beta)}(x),L_1^{(\alpha,\beta)}(x),...,L_N^{(\alpha,\beta)}(x)]^T,$$

in this case, property (2.15) can be written as follows

$$\frac{d}{dx}\mathcal{L}(x)=-\beta\mathcal{D}\mathcal{L}(x), \tag{2.17}$$

where

$$\mathcal{D}=\begin{bmatrix}0&0&0&\cdots&0\\1&0&0&\cdots&0\\1&1&0&\cdots&0\\\vdots&\vdots&\ddots&\ddots&\vdots\\1&\cdots&1&1&0\end{bmatrix}$$

and for k-th derivative we have

$$\frac{d^k}{dx^k}\mathcal{L}(x)=(-\beta)^k\mathcal{D}^k\mathcal{L}(x),\quad k=1,2,3,... \tag{2.18}$$

The *MGLPs* are orthogonal polynomials on $[0,\infty)$ respect to weight function

$$\omega^{(\alpha,\beta)}(x)=x^\alpha e^{-\beta x},$$

in other words

$$\begin{aligned}\left\langle L_m^{(\alpha,\beta)}(x),L_n^{(\alpha,\beta)}(x)\right\rangle_{\omega^{(\alpha,\beta)}(x)}&=\int_0^\infty L_m^{(\alpha,\beta)}(x)L_n^{(\alpha,\beta)}(x)\omega^{(\alpha,\beta)}(x)dx\\&=\frac{\Gamma(n+\alpha+1)}{\beta^{\alpha+1}n!}\delta_{nm},\quad n,m=0,1,2,...,\end{aligned} \tag{2.19}$$

where δ_{nm} is the Kronecker function and $L^2_{\omega^{(\alpha,\beta)}}([0,\infty))$ is a Hilbert space. Note that, all the zeros of modifed generalized Laguerre polynomials are simple and in the $(0,\infty)$. Therefore, $L_n^{(\alpha,\beta)}(x)$ has exactly n zeros, all

of which are simple, real and positive. When N tends to ∞, then the largest root tends to ∞.

Suppose $\mathcal{B}_N = \{L_n^{\alpha,\beta}(x)\}_{n=0}^N$ and we want to use a finite numbers terms for expanssion as

$$u(x,t_j) \cong \tilde{u}(x,t_j) = \tilde{u}^j(x) = \sum_{n=0}^{N} c_n L_n^{\alpha,\beta}(x) = C^T \mathcal{L}(x) = \mathcal{L}(x)^T C, \tag{2.20}$$

where

$$C = [c_0, c_1, c_2, \cdots, c_N]^T,$$

then we have

$$\frac{d^k}{dx^k}\tilde{u}^j(x) = C^T \frac{d^k}{dx^k}\mathcal{L}(x) = (-\beta)^k C^T D^k \mathcal{L}(x) = (-\beta)^k (D^k \mathcal{L}(x))^T C. \tag{2.21}$$

To convince the reader that the approximation error depends on the values of α and β, we write the approximation of $f(t) = e^{-t}$ for $N = 1$. that is

$$f(t) = c_0 L_0^{\alpha,\beta}(t) + c_1 L_1^{\alpha,\beta}(t) = c_0 + c_1((1+\alpha) - \beta t).$$

It can be written by using the orthogonality property

$$\begin{aligned}
c_0 =& \frac{\int_0^\infty f(t) L_0^{\alpha,\beta}(t)\omega^{\alpha,\beta}(t)dt}{\int_0^\infty L_0^{\alpha,\beta}(t) L_0^{\alpha,\beta}(t)\omega^{\alpha,\beta}(t)dt} \\
=& \frac{\beta^{\alpha+1}0!}{\Gamma(\alpha+1)} \int_0^\infty t^\alpha e^{-(\beta+1)t} dt \\
=& \frac{\beta^{\alpha+1}}{\Gamma(\alpha+1)} \int_0^\infty (\frac{\tau}{\beta+1})^\alpha e^{-\tau} \frac{d\tau}{\beta+1} \\
=& \frac{\beta^{\alpha+1}}{\Gamma(\alpha+1)} \frac{1}{(\beta+1)^{(\alpha+1)}} \Gamma(\alpha+1) \\
=& (\frac{\beta}{\beta+1})^{(\alpha+1)},
\end{aligned}$$

where $(\beta+1)t = \tau$ and

$$\begin{aligned}
c_1 =& \frac{\int_0^\infty f(t) L_1^{\alpha,\beta}(t)\omega^{\alpha,\beta}(t)dt}{\int_0^\infty L_1^{\alpha,\beta}(t) L_1^{\alpha,\beta}(t)\omega^{\alpha,\beta}(t)dt} \\
=& \frac{\beta^{\alpha+1}1!}{\Gamma(\alpha+2)} \int_0^\infty [(1+\alpha) - \beta t] t^\alpha e^{-(\beta+1)t} dt \\
=& \frac{\beta^{\alpha+1}}{\Gamma(\alpha+2)} \{(1+\alpha) \int_0^\infty t^\alpha e^{-(\beta+1)t} dt - \beta \int_0^\infty t^{\alpha+1} e^{-(\beta+1)t} dt\} \\
=& \frac{\beta^{\alpha+1}}{\Gamma(\alpha+2)} \{(1+\alpha) \frac{1}{(\beta+1)^{(\alpha+1)}} \Gamma(\alpha+1) - \beta \frac{1}{(\beta+1)^{(\alpha+2)}} \Gamma(\alpha+2)\} \\
=& (\frac{\beta}{\beta+1})^{(\alpha+1)} - (\frac{\beta}{\beta+1})^{(\alpha+2)},
\end{aligned}$$

then we have

$$e^{-t} \approx (\frac{\beta}{\beta+1})^{(\alpha+1)} + ((\frac{\beta}{\beta+1})^{(\alpha+1)} - (\frac{\beta}{\beta+1})^{(\alpha+2)})[(1+\alpha) - \beta t]. \tag{2.22}$$

In other words, (2.22) is the best approximation of the function $g(t) = e^{-t}$ in the

$$Span\{L_0^{\alpha,\beta}(t), L_1^{\alpha,\beta}(t)\}.$$

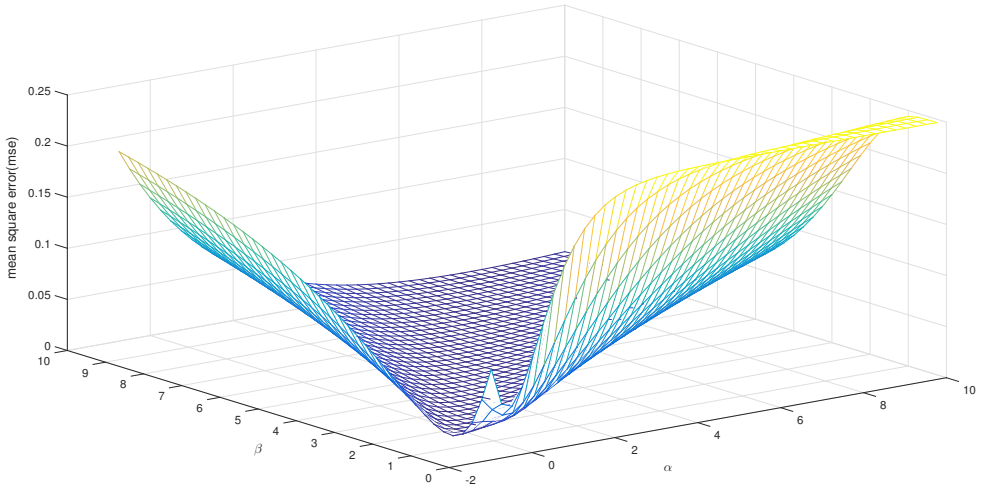

Figure 1: The *MSE* for $g(t) = e^{-t}$ when $N = 1$. The minimum error is 3.6831×10^{-3} which is achieve for $\alpha = 0.75$ and $\beta = 2.5$ and The maximum error is 0.2467 which is achieve for $\alpha = 10$ and $\beta = 0.25$.

Figure 1 shows that the mean squared error(*MSE*) changes with variety in parameters α and β. In other words, the best approximation obtained in $Span\{L_0^{\alpha,\beta}(t), L_1^{\alpha,\beta}(t)\}$ changes with the variety of these two parameters. How to find the best of best approximation among the best approximations?

3. Derivation of the method

In this section, a spectral method based on *MGLPs* with specific parameters α and β is presented. Then, we use the *PSO* algorithm to determine the best parameters α^* and β^*.

3.1. Selection the best parameters

For this purpose, first, it obtains the best approximation for the describing function of the initial condition, $g(t)$, in $Span\{L_n^{\alpha,\beta}(t)\}_{n=0}^{N}$ space. That is

$$g_N^{\alpha,\beta}(t) = \sum_{n=0}^{N} c_n L_n^{\alpha,\beta}(t),$$

where

$$\begin{aligned} c_n &= \frac{\left\langle L_n^{\alpha,\beta}(t), g(t)\right\rangle_{\omega^{(\alpha,\beta)}(t)}}{\left\langle L_n^{\alpha,\beta}(t), L_n^{\alpha,\beta}(t)\right\rangle_{\omega^{(\alpha,\beta)}(t)}} \\ &= \frac{\beta^n n!}{\Gamma(n+\alpha+1)}\left\langle L_n^{\alpha,\beta}(t), g(t)\right\rangle_{\omega^{(\alpha,\beta)}(t)} \\ &= \frac{\beta^n n!}{\Gamma(n+\alpha+1)}\int_0^{\infty} g(t) L_n^{(\alpha,\beta)}(t)\omega^{(\alpha,\beta)}(t)dt, \end{aligned}$$

then minimize the error

$$E(\alpha,\beta) = \|g_N^{\alpha,\beta}(t) - g(t)\|,$$

that is

$$\min_{\alpha,\beta} E(\alpha,\beta). \tag{3.23}$$

In order to get the desired results, it is better to introduce a boundary searching interval for α and β. Let $\alpha \in [\alpha_l, \alpha_u]$ and $\beta \in [\beta_l, \beta_u]$. For minimizing the error (3.23) we use the *PSO* algorithm. After finding the best parameters α^* and β^*, we solve the problem by these best parameters.

3.2. Discretization

For discretization we use the following nodes

$$\begin{cases} x_i = i\Delta_x, i = 0, 1, ..., M, \Delta_x = 1/M, \\ t_j = j\Delta_t, j = 0, 1, ..., N, \Delta_t = 1/N, \end{cases} \tag{3.24}$$

where the Δ_x is step size of x and Δ_t is step size of t. In the [13], time fractional derivative in the $j + 1/2$ time level, is discretized as following

$$\begin{aligned} \frac{\partial^\nu u\left(x, t_{j+1/2}\right)}{\partial t^\nu} = \; & \omega_1 u^j(x) + \sum_{k=1}^{\mathrm{j}-1} \left(\omega_{j-k+1} - \omega_{j-k}\right) u^k(x) - \omega_{\mathrm{j}} u^0(x) \\ & + \sigma \frac{u^{j+1}(x) - u^j(x)}{2^{1-\nu}} + \mathrm{O}\left(\Delta_{\mathrm{t}}^{2-\nu}\right), \end{aligned} \tag{3.25}$$

where

$$\sigma = \frac{\Delta_t^{-\nu}}{\Gamma(2-\nu)}, \quad \omega_j = \sigma\left[(j+1/2)^{1-\nu} - (j-1/2)^{1-\nu}\right],$$

or by using the following notation

$$\begin{aligned} & d_0^j = -\omega_j, \\ & d_k^j = \omega_{j-k+1} - \omega_{j-k}, \quad k = 1, 2, ..., j-1, \\ & d_j^j = \omega_1 - \frac{\sigma}{2^{1-\nu}} = \sigma[(\frac{3}{2})^{1-\nu} - (\frac{1}{2})^{1-\nu}] - \frac{\sigma}{2^{1-\nu}}, \\ & d_{j+1}^j = \frac{\sigma}{2^{1-\nu}}, \end{aligned}$$

the equation (3.25) can be rewritten as follows

$$\begin{aligned} \frac{\partial^\nu u\left(x, t^{j+1/2}\right)}{\partial t^\nu} &= \sum_{k=0}^{\mathrm{j}+1} d_k^j u^k(x) + O(\Delta_{\mathrm{t}}^{2-\nu}) \\ &= \sum_{k=0}^{\mathrm{j}+1} d_k^j \mathcal{L}(x)^T C^k + O(\Delta_{\mathrm{t}}^{2-\nu}). \end{aligned} \tag{3.26}$$

And for the discretization of spatial derivatives, it can be written

$$\begin{aligned}\lambda u_{xx}(x,t_{j+1/2})-\mu u_x(x,t_{j+1/2}) &= \frac{\lambda}{2}(u_{xx}(x,t_{j+1})+u_{xx}(x,t_j))\\ &-\frac{\mu}{2}(u_x(x,t_{j+1})+u_x(x,t_j))\\ &\cong \frac{\lambda}{2}(\frac{d^2}{dx^2}\tilde{u}^{j+1}(x)+\frac{d^2}{dx^2}\tilde{u}^{j}(x))\\ &-\frac{\mu}{2}(\frac{d}{dx}\tilde{u}^{j+1}(x)+\frac{d}{dx}\tilde{u}^{j}(x))\\ &=\frac{\lambda\beta^2}{2}(D^2\mathcal{L}(x))^T(C^{j+1}+C^j)\\ &+\frac{\mu\beta}{2}(D\mathcal{L}(x))^T(C^{j+1}+C^j).\end{aligned} \tag{3.27}$$

3.3. Derivation of the method

Replacing (3.26) and (3.27) in (2.3) conclude that

$$\begin{aligned}\sum_{k=0}^{j+1} d_k^j\mathcal{L}(x)^TC^k &= \frac{\lambda\beta^2}{2}(D^2\mathcal{L}(x))^T(C^{j+1}+C^j)\\ &+\frac{\mu\beta}{2}(D\mathcal{L}(x))^T(C^{j+1}+C^j)+f^{j+1/2}(x)\end{aligned} \tag{3.28}$$

According to the collocation technique, we force $\mathcal{L}(x)$ to satisfy the following collocation conditions:

$$\begin{aligned}\sum_{k=0}^{j+1} d_k^j\mathcal{L}(x_i)^TC^k &= \frac{\lambda\beta^2}{2}(D^2\mathcal{L}(x_i))^T(C^{j+1}+C^j)\\ &+\frac{\mu\beta}{2}(D\mathcal{L}(x_i))^T(C^{j+1}+C^j)\\ &+f^{j+1/2}(x_i),\quad i=1,2,...,M-1.\end{aligned} \tag{3.29}$$

Or

$$\begin{aligned}\mathcal{L}_i^T(-\frac{\lambda\beta^2}{2}D^2-\frac{\mu\beta}{2}D+d_{j+1}^jI)^TC^{j+1} &= \mathcal{L}_i^T(\frac{\lambda\beta^2}{2}D^2+\frac{\mu\beta}{2}D-d_j^jI)^TC^j\\ &-\mathcal{L}_i^T\sum_{k=0}^{j-1}d_k^jC^k+f_i^{j+1/2},\\ &i=1,2,...,M-1,\end{aligned} \tag{3.30}$$

or

$$\mathcal{L}_i^TD_1^TC^{j+1}=\mathcal{L}_i^TD_2^TC^j-\mathcal{L}_i^T\sum_{k=0}^{j-1}d_k^jC^k+f_i^{j+1/2}\quad i=1,2,...,M-1. \tag{3.31}$$

Equations (3.31) for $1\le i\le M-1$, yields a linear system of $M-1$ equations in $M+1$ unknowns $c_0^j, c_1^j, ..., c_M^j$. In order to close this system we need two equations. For this purpose, we use the initial conditions

$$u^{j+1}(0) = u^{j+1}(1) = 0$$

$$\begin{aligned} \tilde{u}^{j+1}(0) &= \mathcal{L}(0)^T C^{j+1} = 0, \\ \tilde{u}^{j+1}(1) &= \mathcal{L}(1)^T C^{j+1} = 0. \end{aligned} \tag{3.32}$$

Equations (3.31) for $1 \le i \le M-1$ and (3.32) yields a linear system of $M+1$ equations in $M+1$ unknowns. Suppose that $\mathbf{L_1}, \mathbf{L_2}$ are $(M+1) \times (M+1)$ matrices as follows

$$\begin{aligned} \mathbf{L} &= [\mathcal{L}_0, \mathcal{L}_1, ..., \mathcal{L}_{M-1}, \mathcal{L}_M]^T, \\ \mathbf{L}_0 &= [\mathbf{0}, \mathcal{L}_1, ..., \mathcal{L}_{M-1}, \mathbf{0}]^T, \\ \bar{\mathbf{L}}_1 &= [\mathbf{D}_1^{-1}\mathcal{L}_0, \mathcal{L}_1, ..., \mathcal{L}_{M-1}, \mathbf{D}_1^{-1}\mathcal{L}_M]^T, \\ \mathbf{L}_1 &= [\mathcal{L}_0, \mathbf{D}_1\mathcal{L}_1, ..., \mathbf{D}_1\mathcal{L}_{M-1}, \mathcal{L}_M]^T = \mathbf{D}_1\bar{\mathbf{L}}_1, \\ \mathbf{L}_2 &= [\mathbf{0}, \mathbf{D}_2\mathcal{L}_1, ..., \mathbf{D}_2\mathcal{L}_{M-1}, \mathbf{0}]^T = D_2\mathbf{L}_0. \end{aligned} \tag{3.33}$$

In this case, (3.31) and (3.32) can be rewritten as follows

$$\mathbf{L}_1 C^{j+1} = \mathbf{L}_2 C^j - \mathbf{L}_0 \sum_{k=0}^{j-1} d_k^j C^k + F^{j+1/2}, \tag{3.34}$$

or

$$C^{j+1} = \mathbf{L}_1^{-1}(\mathbf{L}_2 C^j - \mathbf{L}_0 \sum_{k=0}^{j-1} d_k^j C^k + F^{j+1/2}), \tag{3.35}$$

where

$$\begin{aligned} F^{j+1/2} &= [0, f_1^{j+1/2}, f_2^{j+1/2}, ..., f_{M-1}^{j+1/2}]^T, 0]^T, \\ \mathbf{D_1} &= (-\frac{\lambda\beta^2}{2}D^2 - \frac{\mu\beta}{2}D + d_{j+1}^j I)^T, \\ \mathbf{D_2} &= (\frac{\lambda\beta^2}{2}D^2 + \frac{\mu\beta}{2}D - d_j^j I)^T. \end{aligned}$$

The proposed algorithm needs C^0 to start. Vector C^0, is the Vector C^0, is the expansion coefficients of the $g(x)$ which is calculated by (3.23).

3.4. Solvability, stability and convergence

First of all, we need to show that (3.35) is solvable. For this purpose, the invertibility of $\mathbf{L}_1$ must be proved. We start the proof from the following proposition

Proposition 3.1. *The matrix* $\mathbf{L}$ *in* (3.33) *is invertible.*

Proof. Assume that $\mathbf{L}$ singular. Then there is a vector $\mathbf{v} = [v_0, v_1, ..., v_N]^T \neq \mathbf{0}$ with $\mathbf{Lv} = 0$. The polynomial

$$p(x) = \mathbf{v}^T\mathcal{L}(x) = \sum_{n=0}^{N} v_n L_n^{\alpha,\beta}(x),$$

with $degree(p) = N$ has the $N+1$ distinct roots $x_0, x_1, ..., x_N$ and must vanish identically. Let m be the largest index with $v_m \neq 0$. Then

$$L_m^{\alpha,\beta}(x) = -\frac{1}{v_m}\sum_{n=0}^{m-1} v_n L_n^{\alpha,\beta}(x).$$

This is a contradiction, since the polynomial to the right has the *degree* $= m-1$ but the polynomial to the left has the *degree* $= m$. □

The non-singularity of $\bar{\mathbf{L}}$ results from the non-singularity of $\mathbf{L}$ and the non-singularity of $\mathbf{L}_1$ results from the non-singularity of $\bar{\mathbf{L}}$. Therefore, $\mathbf{L}_1$ is invertible and (3.35) is solvable.

The matrices $\mathbf{D}_1$ and $\mathbf{D}_2$ are lower triangular matrix with fix diagonal entries $\frac{\sigma}{2^{1-\nu}}$, $\omega_1 - \frac{\sigma}{2^{1-\nu}}$ respectively. Then *spectral*$(\mathbf{D}_1) = \frac{\sigma}{2^{1-\nu}}$, *spectral*$(\mathbf{D}_2) = \omega_1 - \frac{\sigma}{2^{1-\nu}}$ and the relations in the following hold

Proposition 3.2. $\mathbf{D}_2 = (2\frac{\sigma}{2^{1-\nu}} - \omega_1)I - \mathbf{D}_1$ *and* ${\mathbf{D}_1}^{-1}\mathbf{D}_2 = (2\frac{1}{2^{1-\nu}} - \omega_1){\mathbf{D}_1}^{-1} - I$.

Proof. By (3.35), $\mathbf{D}_1 + \mathbf{D}_2 = (d_{j+1} - d_j)I$. But

$$\begin{aligned}(d_{j+1} - d_j) &= \frac{\sigma}{2^{1-\nu}} - (\omega_1 - \frac{\sigma}{2^{1-\nu}}) \\ &= 2\frac{\sigma}{2^{1-\nu}} - \omega_1.\end{aligned}$$

So we obtain $\mathbf{D}_2 = (2\frac{\sigma}{2^{1-\nu}} - \omega_1)I - \mathbf{D}_1$. To prove the next part of the claim, it can be written

$$\begin{aligned}{\mathbf{D}_1}^{-1}\mathbf{D}_2 &= {\mathbf{D}_1}^{-1}[(2\frac{\sigma}{2^{1-\nu}} - \omega_1)I - \mathbf{D}_1] \\ &= (2\frac{\sigma}{2^{1-\nu}} - \omega_1){\mathbf{D}_1}^{-1} - I.\end{aligned}$$

□

Proposition 3.3. $\mathbf{L}_1^{-1}\mathbf{L}_2 = \bar{\mathbf{L}}_1^{-1}[(2\frac{\sigma}{2^{1-\nu}} - \omega_1){\mathbf{D}_1}^{-1} - I]\mathbf{L}_0$.

Proof. By (3.33), we have $\mathbf{L}_1 = \mathbf{D}_1\bar{\mathbf{L}}_1$ and $\mathbf{L}_2 = \mathbf{D}_2\mathbf{L}_0$. Then ${\mathbf{L}_1}^{-1} = \bar{\mathbf{L}}_1^{-1}{\mathbf{D}_1}^{-1}$ and

$$\begin{aligned}{\mathbf{L}_1}^{-1}\mathbf{L}_2 &= \bar{\mathbf{L}}_1^{-1}{\mathbf{D}_1}^{-1}\mathbf{D}_2\mathbf{L}_0 \\ &= \bar{\mathbf{L}}_1^{-1}[(2\frac{\sigma}{2^{1-\nu}} - \omega_1){\mathbf{D}_1}^{-1} - I]\mathbf{L}_0.\end{aligned}$$

□

Proposition 3.4. *Spectral*$([(2\frac{\sigma}{2^{1-\nu}} - \omega_1){\mathbf{D}_1}^{-1} - I]) < 1$ *if and only if* $\nu < 1$.

Proof.

$$\begin{aligned}Spectral([(2\frac{\sigma}{2^{1-\nu}} - \omega_1){\mathbf{D}_1}^{-1} - I]) &= (2\frac{\sigma}{2^{1-\nu}} - \omega_1)\frac{1}{\frac{\sigma}{2^{1-\nu}}} - 1 \\ &= 1 - \frac{\omega_1}{\frac{\sigma}{2^{1-\nu}}} \\ &= 1 - \frac{1.5^{1-\nu} - 0.5^{1-\nu}}{0.5^{1-\nu}} \\ &= 2 - 3^{1-\nu}.\end{aligned}$$

Therefore, to complete the proof we have to show $2 - 3^{1-\nu} < 1$ or $1 < 3^{1-\nu}$. But this inequality is true because ν is the order of fractional derivative in the problem (1.1) and it is in the interval $\nu \in (0, 1)$. □

Theorem 3.1. *The proposed algorithm* (3.35) *is stabile if*

$$\Delta_t^{\nu} < \frac{\|\mathbf{L}_0\|}{\Gamma(2-\nu)(\|\mathbf{L}_2\| - \|\mathbf{L}_1\|)}$$

Proof. Let $\epsilon_i^j = u(x_i, t_j) - \tilde{u}(x_i, t_j)$. We can write $u(x_i, t_j) = \mathcal{L}(x_i)^T C^j$ and $\tilde{u}(x_i, t_j) = \mathcal{L}(x_i)^T \tilde{C}^j$ then

$$\begin{aligned}
\|\epsilon_i^j\| &= \|u(x_i, t_j) - \tilde{u}(x_i, t_j)\| \\
&= \|\mathcal{L}(x_i)^T (C^j - \tilde{C}^j)\| \\
&= \|\mathcal{L}(x_i)^T\|.\|(C^j - \tilde{C}^j)\| \\
&= \|\mathcal{L}(x_i)^T\|.\|\zeta^j\|,
\end{aligned}$$

where $\zeta^j = (C^j - \tilde{C}^j)$. To prove stability, we must show $\|\zeta^j\| \leq \|\zeta^0\|$. For this purpose, we use mathematical induction. First, it can be seen that the assertion is true for j=0

$$\begin{aligned}
\|\zeta^1\| &= \|\mathbf{L}_1^{-1}\mathbf{L}_2\zeta^0\| \\
&= \|\mathbf{L}_1^{-1}\mathbf{L}_2\|.\|\zeta^0\| \\
&\leq \|\zeta^0\|
\end{aligned} \tag{3.36}$$

Suppose the induction assertion holds for $n \leq j$ (induction hypothesis) and we try to prove the induction assertion for $j + 1$

$$\begin{aligned}
\|\zeta^{j+1}\| &= \|\mathbf{L}_1^{-1}(\mathbf{L}_2\zeta^j - \mathbf{L}_0 \sum_{k=0}^{j-1} d_k^j \zeta^k)\| \\
&\leq \|\mathbf{L}_1^{-1}\mathbf{L}_2\|.\|\zeta^j\| - \|\mathbf{L}_1^{-1}\mathbf{L}_0\|.\left|\sum_{k=0}^{j-1} d_k^j\right|.\|\zeta^k\| \\
&\leq \|\mathbf{L}_1^{-1}\mathbf{L}_2\|.\|\zeta^j\| - \|\mathbf{L}_1^{-1}\mathbf{L}_0\|.\left|\sum_{k=0}^{j-1} d_k^j\right|.\|\zeta^j\| \\
&= \{\|\mathbf{L}_1^{-1}\mathbf{L}_2\| - \|\mathbf{L}_1^{-1}\mathbf{L}_0\|.\left|\sum_{k=0}^{j-1} d_k^j\}\right|.\|\zeta^j\| \\
&= \{\|\mathbf{L}_1^{-1}\mathbf{L}_2\| - \|\mathbf{L}_1^{-1}\mathbf{L}_0\|\sigma\}\|\zeta^j\| \\
&\leq \|\zeta^j\|
\end{aligned} \tag{3.37}$$

The last inequality is true by assumption of theorem. By hypothesis of induction we can write

$$\|\zeta^{j+1}\| \leq \|\zeta^j\| \leq \|\zeta^0\| \tag{3.38}$$

Therefore, the proposed algorithm is conditionally stable. □

Theorem 3.2. *Spectral*$(\mathbf{L}_1{}^{-1}\mathbf{L}_2) < 1$ *so the proposed algorithm* (3.35) *is convergent.*

Proof. By Prop.(3.2) and Prop.(3.3), $\mathbf{L}_1{}^{-1}\mathbf{L}_2 = \bar{\mathbf{L}}_1^{-1}[(2\frac{\sigma}{2^{1-\nu}} - \omega_1)\mathbf{D}_1{}^{-1} - I]\mathbf{L}_0$. So,

$$\begin{aligned}
\textit{Spectral}(\mathbf{L}_1{}^{-1}\mathbf{L}_2) &= \textit{Spectral}(\bar{\mathbf{L}}_1^{-1}[(2\frac{\sigma}{2^{1-\nu}} - \omega_1)\mathbf{D}_1{}^{-1} - I]\mathbf{L}_0) \\
&\leq \textit{Spectral}([(2\frac{\sigma}{2^{1-\nu}} - \omega_1)\mathbf{D}_1{}^{-1} - I]) \\
&= 2 - 3^{1-v}.
\end{aligned}$$

therefore, when $\nu < 1$ we have $2 - 3^{1-v} < 1$. Consequently *Spectral*$(\mathbf{L}_1{}^{-1}\mathbf{L}_2) < 1$. But $\|\mathbf{L}_1{}^{-1}\mathbf{L}_2\|_2 =$ *Spectral*$(\mathbf{L}_1{}^{-1}\mathbf{L}_2)$ then proposed algorithm (3.35) is convergent. □

4. Numerical results

In this section, to show the accuracy of the proposed algorithm, we solve some problems for example. In the problems, we setting $\Delta_x = 1/M, \Delta_t = 1/N$ and $M = 5, N = 1000$. For Implementation, the matlab software are used and the errors are calculated in the end of interval ,$t = 1$ by the formula

$$error = \|u(x,1) - \tilde{u}(x,1)\|_2. \tag{4.39}$$

Since the accuracy of the g approximation is very effective (as the first step in the accuracy of the algorithm), we try to find the best parameters α^{best_g} and β^{best_g} for the approximation of $g(x)$. For this purpose, we use two different search methods to find the optimal parameters(α, β) to approximate g. The search operation is performed in the $\alpha \in [-1, 10]$ and $\beta \in [0, 10]$ intervals. In the step-by-step ($S.b.S$) searching we set $\Delta_\alpha = \Delta_\beta = 0.25$.

Example 4.1. *Consider the time-fractional advection-diffusion equation as*

$$\begin{cases} \partial_t^{0.5} u = u_{xx} + \frac{6}{\Gamma(3.5)} t^{2.5} x(1-x) + 2(1+t^3), \\ I.C.: \quad u(x,0) = x(1-x), \\ B.C.: \quad u(0,t) = u(1,t) = 0, \end{cases} \tag{4.40}$$

The exact solution for this problem is $u(x,t) = x(1-x)(1+t^3)$. With the help of PSO algorithm, the best parameters for approximating of $g(x)$ are obtained as

$$(\alpha_{pso}^{best_g} = 1.6061, \beta_{pso}^{best_g} = 6.4396) \quad e_{pso}^{best_g} = 2.2204 \times 10^{-16},$$

Figure 2 shows the steps of this search. By using the step-by-step search, the best parameters for approximating of $g(x)$ are obtained as

$$(\alpha_{S.b.S}^{best_g} = 0, \beta_{S.b.S}^{best_g} = 4.25) \quad e_{S.b.S}^{best_g} = 1.5203 \times 10^{-11}.$$

Figure 3 shows the steps of this search.

When the exact solution of the problem is available, the error of the proposed algorithm can be calculated. Figure 4 shows the errors of the proposed algorithm at $t = 1$ for different parameters. As can be seen, improper selection of these parameters can increase the error of the algorithm by tens of times. From PSO algorithm for maximizing the error of $u(x, t = 1)$ approximation by using the proposed method, the worst parameters obtained as

$$(\alpha_{pso}^{worst_u} = 5.4347, \beta_{pso}^{worst_u} = 0.0500) \quad e_{pso}^{worst_u} = 1.5634 \times 10^{-2}.$$

To compare the error of the proposed algorithm with the best and worst parameters, see Figure 5. As can be seen, the solution obtained with the best parameters is matched with the exact solution with considerable accuracy, while the solution obtained from the worst parameter has a non-negligible difference with the exact solution.

Example 4.2. *Consider the time-fractional advection-diffusion equation as*

$$\begin{cases} \partial_t^{0.5} u = 4u_{xx} - 3u_x + \dfrac{6}{\Gamma(3.5)} t^{2.5} sin(\pi x) + 4\pi^2(1+t^3) sin(\pi x) \\ \qquad\qquad + 3\pi(1+t^3) cos(\pi x), \\ B.C.: \quad u(0,t) = u(1,t) = 0, \\ I.C.: \quad u(x,0) = sin(\pi x), \end{cases} \tag{4.41}$$

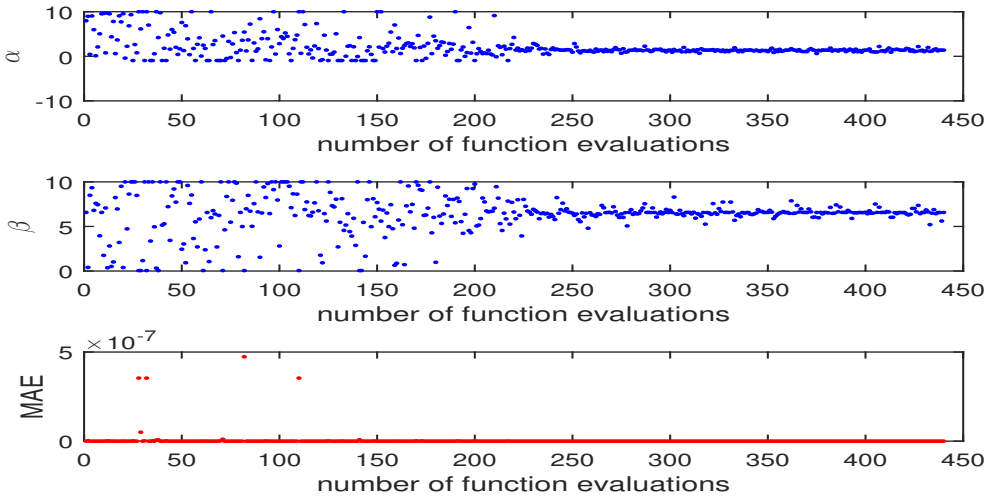

Figure 2: Minimizing the maximum absolute error for g when $M = 5$ by using *PSO* algorithm for Example 4.1.

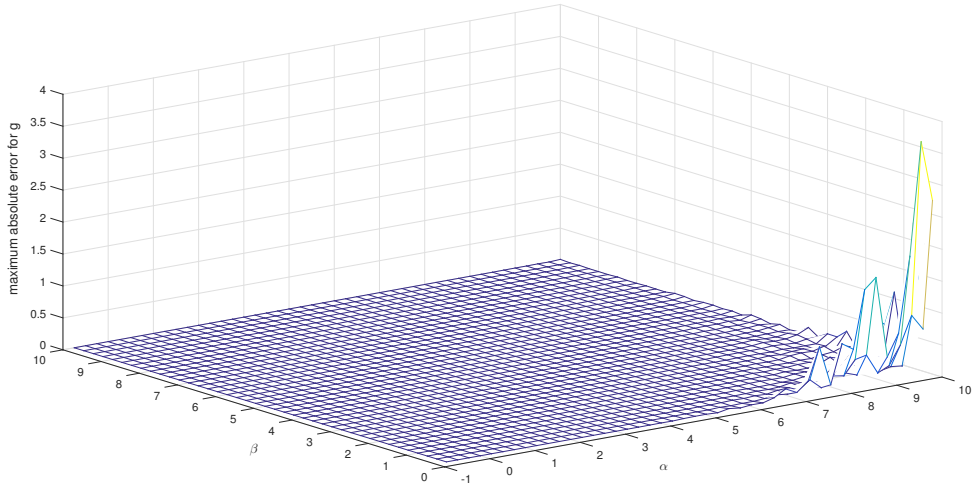

Figure 3: The maximum absolute error for g when $M = 5$ by using *S.b.S* method for Example 4.1

note that, $\lambda = 4, \mu = 3, \alpha = 0.5, g(x) = sin(\pi x)$ *and* $f(x,t) = \frac{6}{\Gamma(3.5)}t^{2.5}sin(\pi x)+4\pi^2(1+t^3)sin(\pi x)+3\pi(1+t^3)cos(\pi x)$. *The exact solution for this problem is* $u(x,t) = (1+t^3)sin(\pi x)$. *With the help of PSO algorithm, the best parameters for approximating of* $g(x)$ *are obtained as*

$$(\alpha_{pso}^{best_g} = 0.9496, \beta_{pso}^{best_g} = 10) \quad e_{pso}^{best_g} = 6.1472 \times 10^{-3},$$

Figure 6 shows the steps of this search. By using the step-by-step search, the best parameters for approximating of $g(x)$ *are obtained as*

$$(\alpha_{S.b.S}^{best_g} = 5, \beta_{S.b.S}^{best_g} = 10) \quad e_{S.b.S}^{best_g} = 6.4396 \times 10^{-3},$$

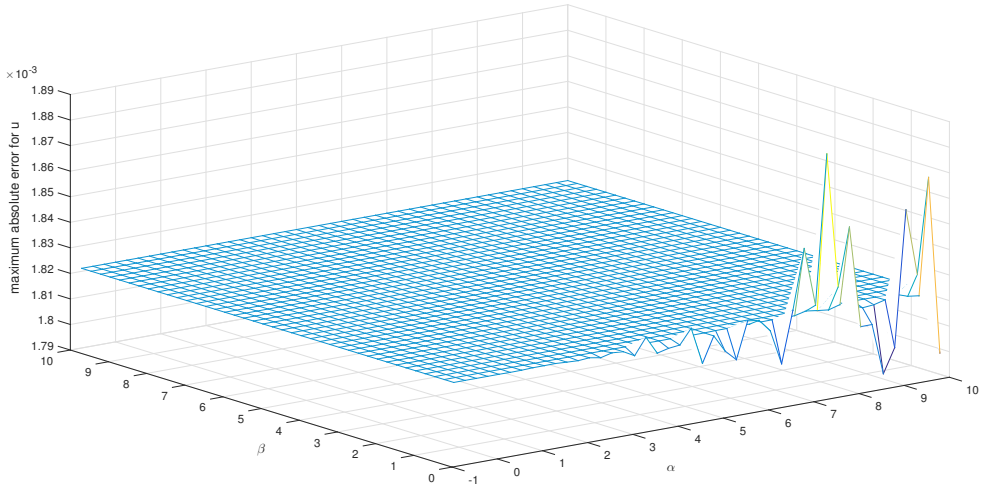

Figure 4: The maximum absolute error for u when $M = 5$ and $N = 1000$ for Example 4.1.

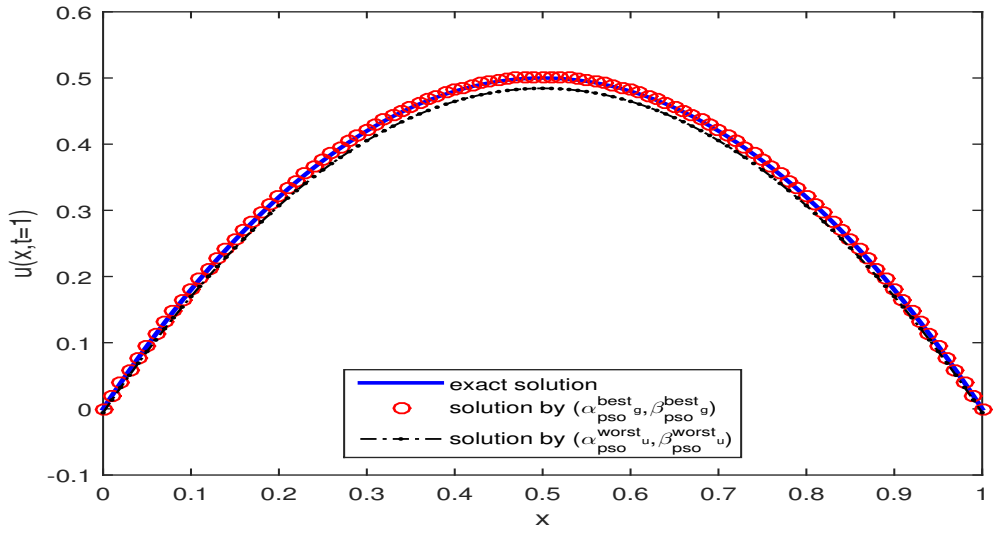

Figure 5: Comparison of the exact solution with the numerical solutions for the best parameter $(\alpha_{pso}^{best_g}, \beta_{pso}^{best_g})$ and the worst parameter $(\alpha_{s.b.s}^{worst_u}, \beta_{s.b.s}^{worst_u})$ at $t = 1$ when $M = 5$ and $N = 1000$ for Example 4.1.

Figure 7 shows the steps of this search.

Figure8 shows the errors of the proposed algorithm at $t = 1$ for different parameters. As can be seen, improper selection of these parameters can increase the error of the algorithm by thousands of times. From S.b.S method for $u(x, t = 1)$ approximation by using the proposed method, the worst parameters obtained as

$$(\alpha_{S.b.S}^{worst_u} = 10, \beta_{S.b.S}^{worst_u} = 3.5) \quad e_{S.b.S}^{worst_u} = 0.1824.$$

In Figure 9, the exact solution is compared with the solutions obtained from the proposed numerical method with the best parameters for g, $\alpha_{pso}^{best_g} = 10$, $\beta_{pso}^{best_g} = 3.5$, and the worst parameters for u, $\alpha_{S.b.S}^{worst_u} = 10$, $\beta_{S.b.S}^{worst_u} = 3.5$.

This figure shows that the proposed parameters obtained by using the PSO algorithm on the known function $g(x)$ can greatly increase the accuracy of the proposed algorithm. It should not be forgotten that we never claim that the suggested parameters $\alpha_{pso}^{best_g}$, $\beta_{pso}^{best_g}$ are the best parameters, but from the known information of the problem, these parameters may be the best possible parameters. In other words, finding the better parameters than the proposed parameters is possible only when the exact solution is available, but in practice this exact solution is not available. From the observation of Figure 10 and Figure 11, it seems that the effect of the changes of parameters α and β on the approximation accuracy of these two functions $g(x)$ and $u(x,t=1)$ are almost similar to each other, so choosing the optimal parameters from the approximation of function $g(x)$ is not so inappropriate.

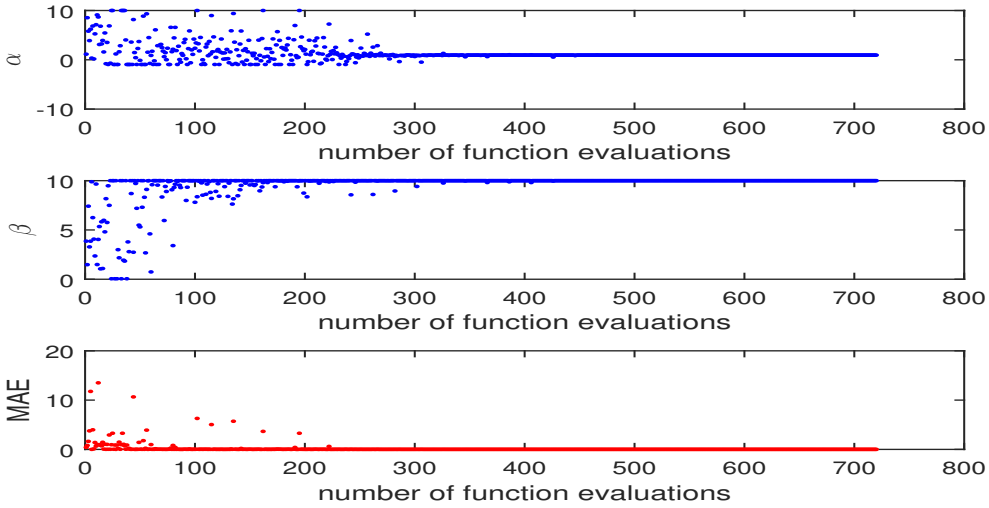

Figure 6: Minimizing the maximum absolute error for g when $M = 5$ by using *PSO* algorithm for Example 4.2.

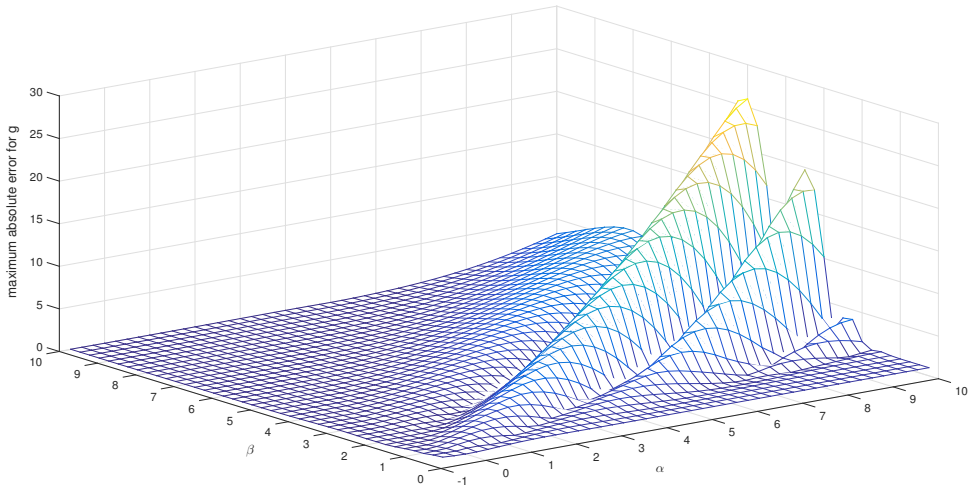

Figure 7: The maximum absolute error for g when $M = 5$ by using *S.b.S* method for Example 4.2.

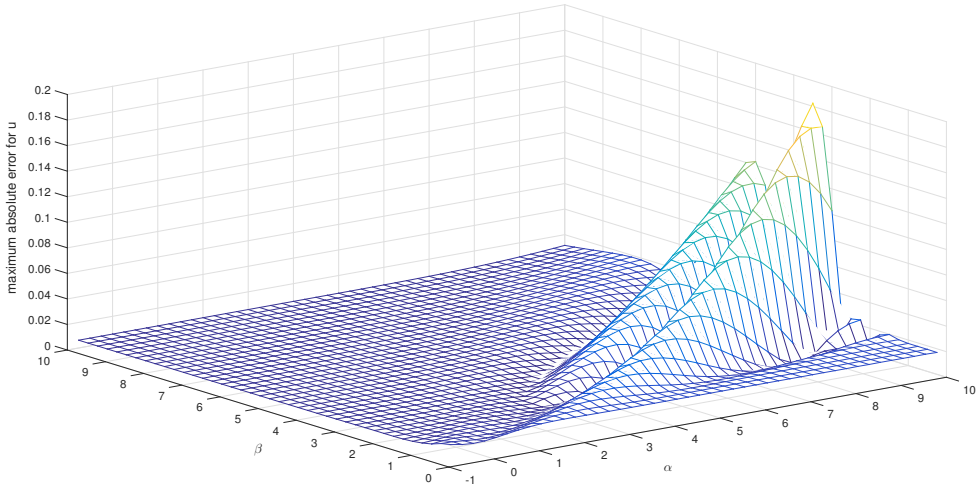

Figure 8: The maximum absolute error for u when $M = 5$ and $N = 1000$ for Example 4.2.

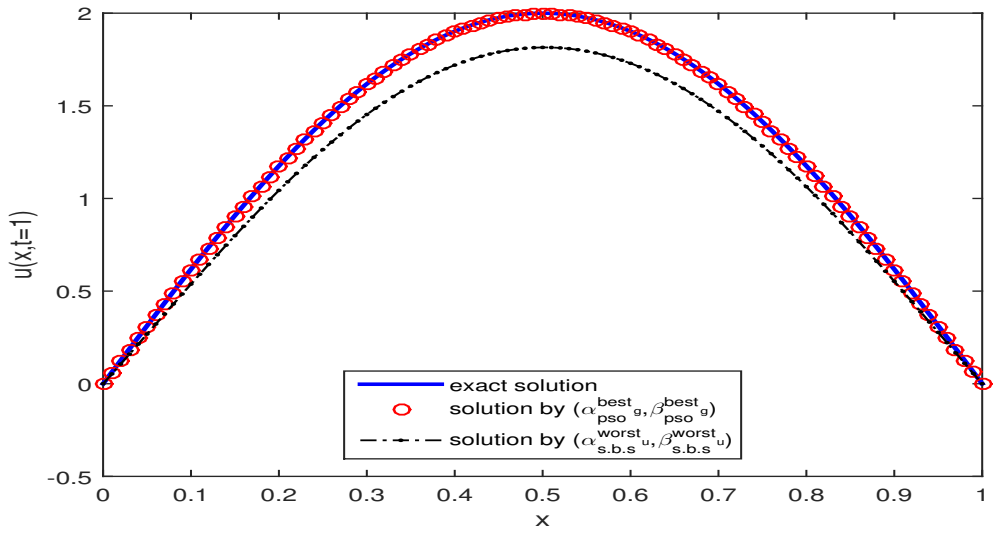

Figure 9: Comparison of the exact solution with the numerical solutions for the best parameter $(\alpha^{best_g}_{pso}, \beta^{best_g}_{pso})$ and the worst parameter $(\alpha^{worst_u}_{s.b.s}, \beta^{worst_u}_{s.b.s})$ at $t = 1$ when $M = 5$ and $N = 1000$ for Example 4.2.

Example 4.3. *Consider the time-fractional advection-diffusion equation as*

$$\begin{cases} \partial_t^{0.3} u = 5u_{xx} - u_x + \frac{-60}{\Gamma(5.7)} x(x-1)t^{4.7} - 5(2-t^5) + (2x-1)(1-0.5t^5), \\ B.C.: \quad u(0,t) = u(1,t) = 0, \\ I.C: \quad u(x,0) = x(x-1), \end{cases} \tag{4.42}$$

note that, $\lambda = 5$, $\mu = 1$, $\alpha = 0.3$, $g(x) = x(x-1)$ *and* $f(x,t) = \frac{-60}{\Gamma(5.7)} x(x-1)t^{4.7} - 5(2-t^5) + (2x-1)(1-0.5t^5)$,. *The exact solution for this problem is* $u(x,t) = x(x-1)(1-0.5t^5)$. *With the help of PSO algorithm, the best parameters*

for approximating of $g(x)$ are obtained as

$$(\alpha_{pso}^{best_g} = 1.1087, \beta_{pso}^{best_g} = 6.18108) \quad e_{pso}^{best_g} = 1.1102 \times 10^{-16},$$

Figure 10 shows the steps of this search. By using the step-by-step search, the best parameters for approximating of $g(x)$ are obtained as

$$(\alpha_{S.b.S}^{best_g} = -0.5, \beta_{S.b.S}^{best_g} = 5.5) \quad e_{S.b.S}^{best_g} = 8.3267 \times 10^{-17},$$

Figure 11 shows the steps of this search.

Figure 12 shows the errors of the proposed algorithm at $t = 1$ for different parameters. As can be seen, improper selection of these parameters can increase the error of the algorithm by thousands of times. From S.b.S method for $u(x, t = 1)$ approximation by using the proposed method, the worst parameters obtained as

$$(\alpha_{S.b.S}^{worst_u} = 9.25, \beta_{S.b.S}^{worst_u} = 0.25) \quad e_{S.b.S}^{worst_u} = 6.8804 \times 10^{-4}.$$

In Figure 13, the exact solution is compared with the solutions obtained from the proposed numerical method with the best parameters for g, $\alpha_{pso}^{best_g}$, $\beta_{pso}^{best_g}$, and the worst parameters for u, $\alpha_{S.b.S}^{worst_u}$, $\beta_{S.b.S}^{worst_u}$. This figure shows that the proposed parameters obtained by using the PSO algorithm on the known function $g(x)$ can increase the accuracy of the proposed algorithm.

In this example, similar to example **??**, *the function describing the initial condition is a polynomial of degree 2. Also, the exact solution in $t = 1$ is again a polynomial of degree 2. Therefore, they are approximated carefully, for this reason, the error in this example is not significantly different for the worst parameters and the best parameters.*

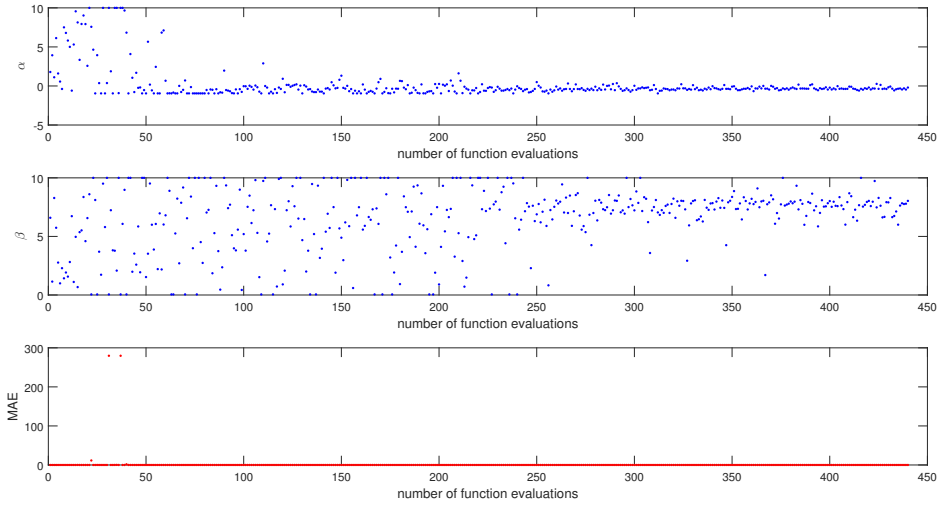

Figure 10: Minimizing the maximum absolute error for g when $M = 5$ by using *PSO* algorithm for Example 4.3.

Example 4.4. *Consider the time-fractional advection-diffusion equation as*

$$\begin{cases} \partial_t^{0.5} u = 4u_{xx} - 3u_x + (1 - e^{-x})cos(\frac{\pi}{2}x)\frac{\Gamma(4)}{\Gamma(3.5)}t^{2.5} \\ \qquad + (((7 - \pi^2)e^{-x} + \pi^2)cos(\frac{\pi}{2}x) + (\frac{11\pi}{2}e^{-x} - \frac{3\pi}{2})sin(\frac{\pi}{2}x))(1 + t^3) \\ B.C.: \quad u(0, t) = u(1, t) = 0, \\ I.C: \quad u(x, 0) = (1 - e^{-x})cos(\frac{\pi}{2}x), \end{cases} \tag{4.43}$$

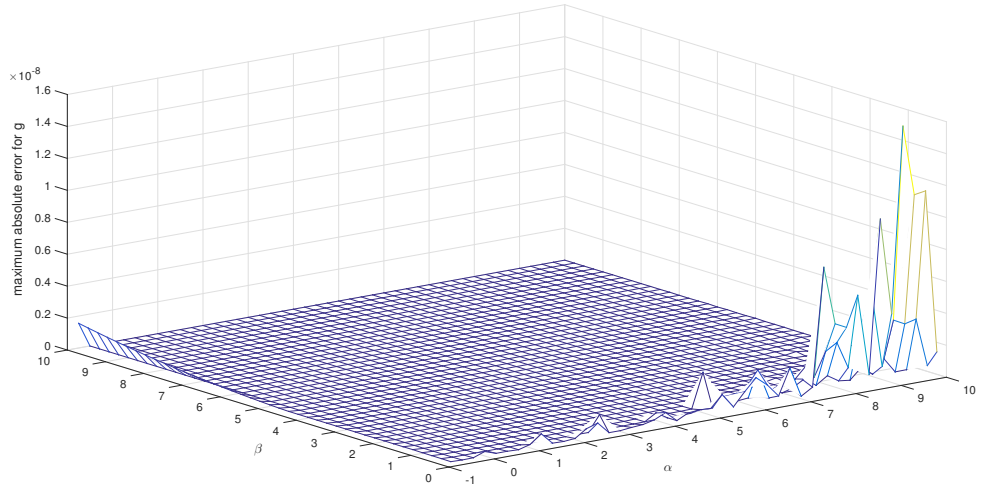

Figure 11: The maximum absolute error for g when $M = 5$ by using *S.b.S* method for Example 4.3.

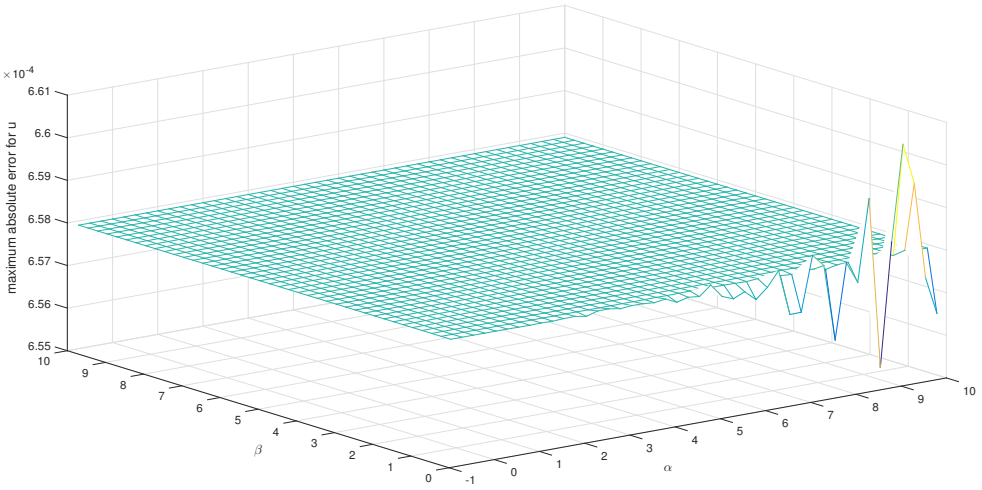

Figure 12: The maximum absolute error for u when $M = 5$ and $N = 1000$ for Example 4.3.

note that, $\lambda = 4$, $\mu = 3$, $\gamma = 0.5$, $g(x) = (1 - e^{-x})cos(\frac{\pi}{2}x)$ *and* $f(x,t) = (1 - e^{-x})cos(\frac{\pi}{2}x)\frac{\Gamma(4)}{\Gamma(3.5)}t^2.5 + (((7 - \pi^2)e^{-x} + \pi^2)cos(\frac{\pi}{2}x) + (\frac{11\pi}{2}e^{-x} - \frac{3\pi}{2})sin(\frac{\pi}{2}x))(1 + t^3)$. *The exact solution for this problem is* $u(x,t) = (1 - e^{-x})cos(\frac{\pi}{2}x)(1 + t^3)$. *With the help of PSO algorithm, the best parameters for approximating of* $g(x)$ *are obtained as*

$$(\alpha_{pso}^{best_g} = 1.1233, \beta_{pso}^{best_g} = 9.9992) \quad e_{pso}^{best_g} = 1.0891 \times 10^{-4},$$

Figure 14 shows the steps of this search. By using the step-by-step search, the best parameters for approximating of $g(x)$ *are obtained as*

$$(\alpha_{S.b.S}^{best_g} = 1, \beta_{S.b.S}^{best_g} = 9.75) \quad e_{S.b.S}^{best_g} = 1.1349 \times 10^{-4},$$

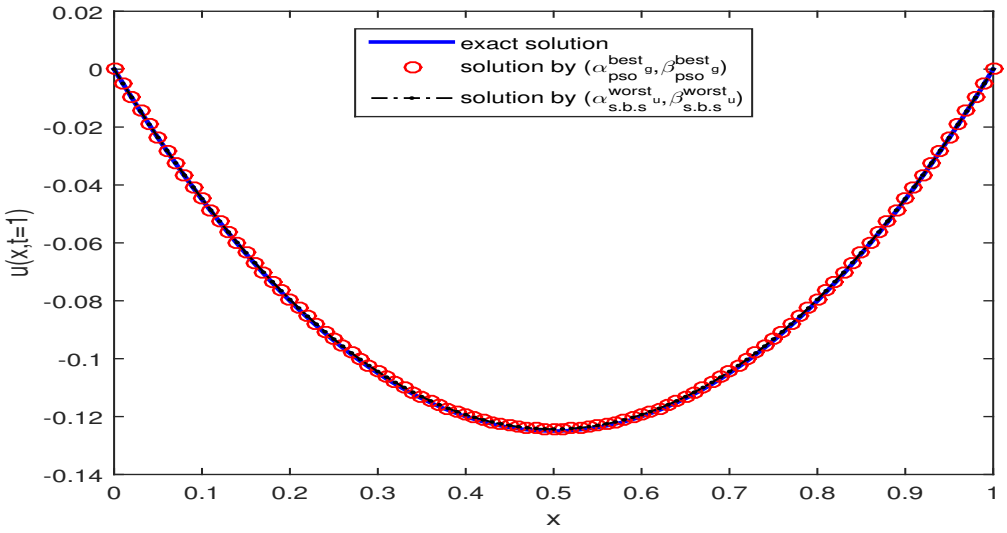

Figure 13: Comparison of the exact solution with the numerical solutions for the best parameter $(\alpha^{best_g}_{pso}, \beta^{best_g}_{pso})$ and the worst parameter $(\alpha^{worst_u}_{s.b.s}, \beta^{worst_u}_{s.b.s})$ at $t = 1$ when $M = 5$ and $N = 1000$ for Example 4.3.

Figure 15 shows the steps of this search.

Figure 16 shows the errors of the proposed algorithm at $t = 1$ for different parameters. As can be seen, improper selection of these parameters can increase the error of the algorithm by hundreds of times. From S.b.S method for $u(x, t = 1)$ approximation by using the proposed method, the worst parameters obtained as

$$(\alpha^{worst_u}_{S.b.S} = 10, \beta^{worst_u}_{S.b.S} = 3.25) \quad e^{worst_u}_{S.b.S} = 0.2767,$$

and the best parameters for u are

$$(\alpha^{best_u}_{S.b.S} = 9.5, \beta^{best_u}_{S.b.S} = 1.75) \quad e^{best_u}_{S.b.S} = 0.0365.$$

To compare the error of the proposed algorithm with the best and worst parameters, see Figure 17. As can be seen, the solution obtained with the best parameters is closer to the exact solution with acceptable accuracy, while the solution obtained from the worst parameter has a non-negligible difference with the exact solution.

It should be noted that unlike the Examples 4.1 and 4.3, the function describing the initial condition in Examples 4.2 and 4.4 is not a polynomial function. Therefore, it is not possible to approximate these functions with the help of $L^{\alpha,\beta}_n(x)_{n=0}^{N=5}$ carefully and it is because the error in Examples 4.2 and 4.4 is considerably unacceptable.

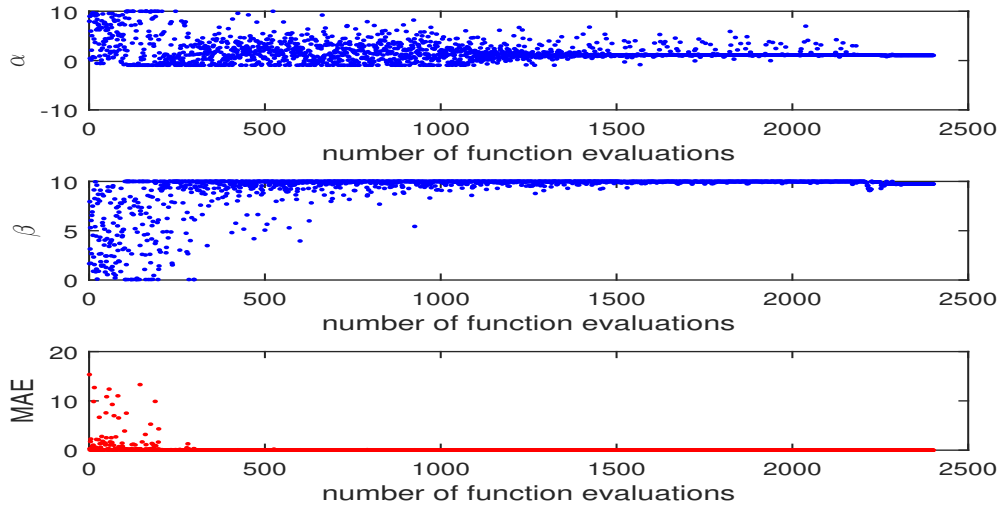

Figure 14: Minimizing the maximum absolute error for g when $M = 5$ by using *PSO* algorithm for Example 4.4

.

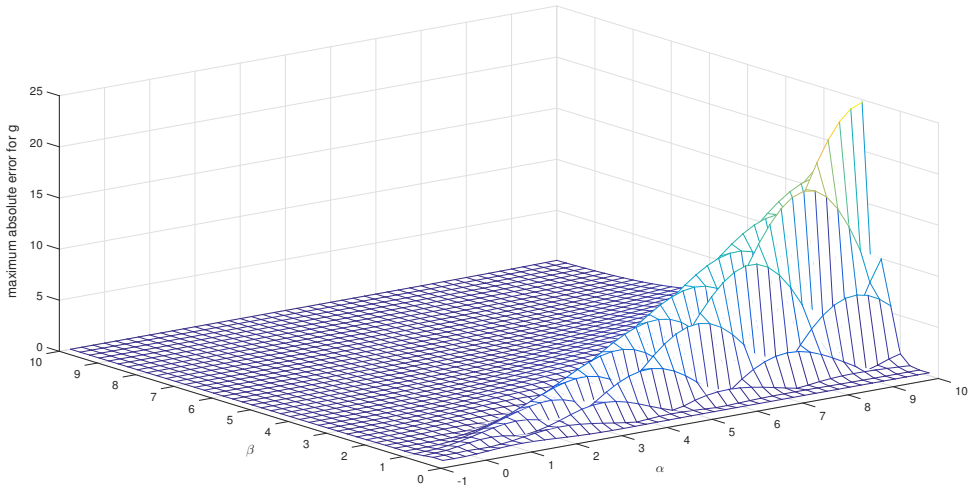

Figure 15: The maximum absolute error for g when $M = 5$ by using *S.b.S* method for Example 4.4.

5. Conclusion

Modified generalized Laguerre polynomials were used to solve problem (1.1) of spectral and collacation methods. It was shown that the error is very sensitive to parameters α and β. Using *PSO* algorithm, the best values of parameters α and β were found for each of the examples so that the error of the algorithm was minimized for that problem. Numerical results show that when the function describing the initial condition is a polynomial function, in this case, the sensitivity to the parameters is less, provided that the number of expansion terms are sufficient but when the number of expansion terms is not enough, the sensitivity to the parameters is more. Also, when the initial condition is a transcendental function, the sensitivity to the

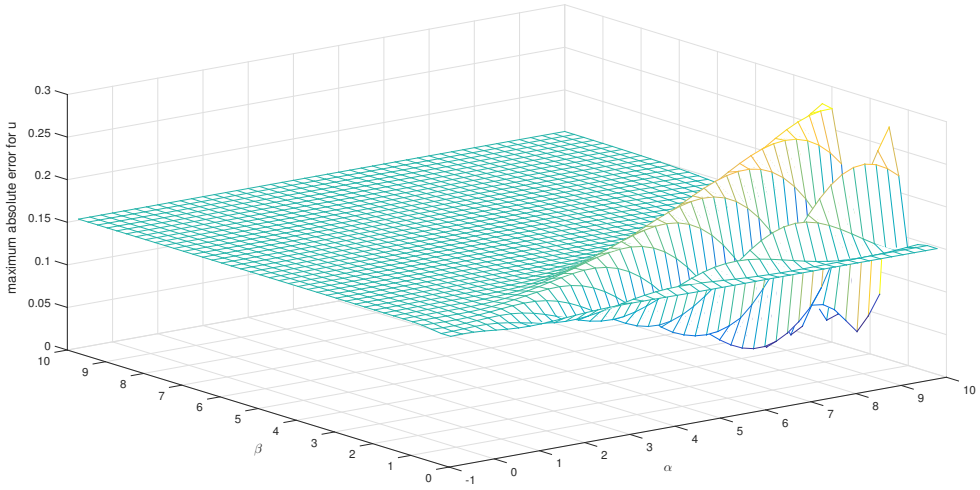

Figure 16: The maximum absolute error for u when $M = 5$ and $N = 1000$ for Example 4.4.

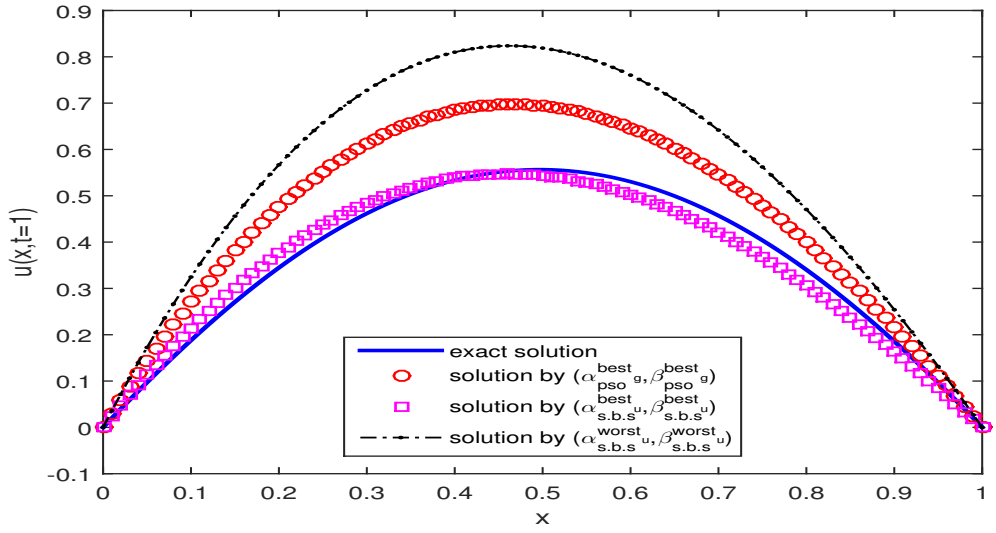

Figure 17: Comparison of the exact solution with the numerical solutions for the best parameters $(\alpha^{best_g}_{pso}, \beta^{best_g}_{pso})$, $(\alpha^{best_u}_{s.b.s}, \beta^{best_u}_{s.b.s})$ and the worst parameter $(\alpha^{worst_u}_{s.b.s}, \beta^{worst_u}_{s.b.s})$ at $t = 1$ when $M = 5$ and $N = 1000$ for Example 4.4.

parameters is very high.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and materials

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

Competing interests

The authors declare that they have no competing interests.

Funding

Not applicable.

Authors' contributions

The authors declare that the study was realized in collaboration with equal responsibility. All authors read and approved the final manuscript.

References

[1] N. Chems Eddine, M.A. Ragusa, D.D. Repov, *On the concentration–compactness principle for anisotropic variable exponent Sobolev spaces and its applications*, Fract. Calc. Appl. Anal.,(2024) **27**, 725–756. https://doi.org/10.1007/s13540-024-00246-8

[2] S. Etemad, M. A. Ragusa, Sh. Rezapour A. Zada,*Existence of property of solutions for multi–order q–difference FBVPs based on condensing operators and end–point technique*, Fixed Point Theory, (2024) **25**(1) 115–142. https://doi.org/10.24193/fpt-ro.2024.1.08

[3] A. De Gaetano, M. Jleli, M. A. Ragusa, B. Samet,*Nonexistence results for nonlinear fractional differential inequalities involving weighted fractional derivatives*, Discrete and Continuous Dynamical Systems - Series S,(2023) **16**(6), 1300–1322. https://doi.org/10.3934/dcdss.2022185

[4] I. Podlubny, *Fractional Differential Equations*, *Academic Press*, San Diego, (1999).

[5] P. Roul, V.M.K. Prasad Goura, R. Agarwal, *A new high order numerical approach for a class of nonlinear derivative dependent singular boundary value problems*, Appl. Numer. Math., (2019) **145** 315–341. https://doi.org/10.1016/j.apnum.2019.06.016.

[6] M. Irodotou-Ellina, E.N. Houstis, *An $O(h^6)$ quintic spline collocation method for fourth order two-point boundary value problems*, BIT, (1988) **28** 288–301. https://doi.org/10.1007/BF01934092.

[7] P. Roul, *A high accuracy numerical method and its convergence for time-fractional Black-Scholes equation governing European options*,Applied Numerical Mathematics, (2020), **151**, 472–493. https://doi.org/10.1016/j.apnum.2019.11.004.

[8] P. Roul, K. Thula, V.M.K. Prasad Goura,*An optimal sixth-order quartic B-spline collocation method for solving Bratu-type and Lane-Emden type problems*, Math. Methods Appl. Sci., (2019) **42** 2613–2630. https://doi.org/10.1002/mma.5537.

[9] K. Thula, P. Roul,*A high-order B-spline collocation method for solving nonlinear singular boundary value problems arising in engineering and applied science*, Mediterr. J. Math.,(2018) **15** 176. https://doi.org/10.1007/s00009-018-1220-y.

[10] H. Azin, F.Mohammadi, M. H. Heydari,*A hybrid method for solving time fractional advection-diffusion equation on unbounded space domain*, Advances in Difference Equations, (2020) 2020596. https://doi.org/10.1186/s13662-020-03053-6.

[11] V. Saw, S. Kumar, *Fourth kind shifted chebyshev polynomials for solving space fractional order advection–dispersion equation based on collocation method and finite difference approximation*, Int. J. Appl. Comput. Math., (2018) **4**82. https://doi.org/10.1007/s40819-018-0517-7.

[12] M. M. Khader, N. H. Sweilam, *Approximate solutions for the fractional advection–dispersion equation using Legendre pseudo-spectral method*, Comp. Appl. Math., (2014), **33**,739–750. https://doi.org/10.1007/s40314-013-0091-x.

[13] I. Karatay, N. Kale, S. R. Bayramoglu, *A new difference scheme for time fractional heat equations based on the Crank-Nicholson method*, Fractional Calculus and Applied Analysis, (2013) **16** (4), 892–910. https://doi.org/10.2478/s13540-013-0055-2.

[14] M. Shakeel, I. Hussain, H. Ahmad, I. Ahmad, P. Thounthong, Y.F. Zhang, *Meshless technique for the solution of time-fractional partial differential equations having real-world applications*, Hindawi, (2020), Article ID 8898309, 17 pages. https://doi.org/10.1155/2020/8898309.

[15] M. Partohaghighi, M. Mortezaee, A. Akgül , S. M. Eldin, *Numerical estimation of the fractional advection–dispersion equation under the modified Atangana–Baleanu–Caputo derivative*, Results in Physics,(2023),**49** 106451. https://doi.org/10.1016/j.rinp.2023.106451.

[16] Z. Liu, X. Li, *A Crank–Nicolson difference scheme for the time variable fractional mobile–immobile advection–dispersion equation*, Computational and Applied Mathematics, (2017) **56**, 391–410. https://doi.org/10.1007/s12190-016-1079-7.

[17] A. H. Bhrawy,*A new Numerical algorithm for solving a class of fractional advection-dispersion eEquation with variable coefficients using Jacobi polynomials*, Abstract and Applied Analysis, (2013), Article ID 954983, 9 pages. https://doi.org/10.1155/2013/954983.

[18] C. E. Mejía, A. Piedrahita, *A numerical method for a time-fractional advection–dispersion equation with a nonlinear source term*, Applied Mathematics and Computing, (2019), **61**, 593–609. https://doi.org/10.1007/s12190-019-01266-x.

[19] A. Bhardwaj, A. Kumar, *A numerical solution of time–fractional mixed diffusion and diffusion–wave equation by an RBF based meshless method*, Engineering with Computers, (2020), **38**, 1883–1903. https://doi.org/10.1007/s00366-020-01134-4.

[20] A.S.V. Ravi Kanth S. Deepika, *Application and analysis of spline approximation for time fractional mobile–immobile advection-dispersion equation*, Numerical Methods for Partial Differential Equations, (2018), **34** (5), 1799–1819. https://doi.org/10.1002/num.22266.
[21] A. Mohebbi, M Abbaszadeh,*Compact finite difference scheme for the solution of time fractional advection-dispersion equation*, Numerical Algorithms,(2013)**63**(3), 431–452. https://doi.org/10.1007/s11075-012-9631-5.
[22] H. Singh, *Jacobi collocation method for the fractional advection-dispersion equation arising in porous media*, Numerical Methods for Partial Differential Equations, (2022), **38**(3), 636–653. https://doi.org/10.1002/num.22674.
[23] A. S. Moghadam, M. Arabameri, M. Barfeie, *Numerical solution of space-time variable fractional order advection-dispersion equation using radial basis functions*, Journal of Mathematical Modeling, (2022) **10**(3), 549–562. https://doi.org/10.1002/num.22674.
[24] S. Momani, Z. Odibat, *Numerical solutions of the space-time fractional advection–dispersion equation*, Numerical Methods for Partial Differential Equations, (2008) **24**(6),549–562. https://doi.org/10.1002/num.20324.
[25] S. Haq, A. Ghafoor, M. Hussain, *Numerical solutions of variable order time fractional (1+1)- and (1+2)-dimensional advection dispersion and diffusion models*,Applied Mathematics and Computation, (2019), **360**, 107–121. https://doi.org/10.1016/j.amc.2019.04.085.
[26] V. Saw, S. Kumar,*Second kind Chebyshev polynomials for solving space fractional advection–dispersion equation using collocation method*, Iranian Journal of Science and Technology, Transactions Science, (2018), **43**, 1027–1037. https://doi.org/10.1007/s40995-018-0480-5.
[27] M. K. Singh, A. Chatterjee, *Solution of one-dimensional space- and time–fractional advection–dispersion equation by homotopy perturbation method*, Acta Geophysica, (2017), **65**(2), 353–361. https://doi.org/10.1007/s11600-017-0035-8.
[28] M. K. Singh, A. Chatterjee, V. P. Singh, *Solution of one-dimensional time fractional advection dispersion equation by homotopy analysis method*, Journal of Engineering Mechanics, (2017) **143**(9). https://doi.org/10.1061/(ASCE)EM.1943-7889.0001318.
[29] S. Cao, J. Jiang, J. Wu, *Solving time–fractional advection–dispersion equation by variable weights particle tracking method*, Journal of Statistical Physics, (2017), **168**, 1248–1258. https://doi.org/10.1007/s10955-017-1849-0.
[30] F. Huang, F. Liu, *The time fractional diffusion equation and the advection dispersion equation*, The ANZIAM Journal, (2009) **46**(03), 317–330. https://doi.org/10.1017/S1446181100008282.
[31] G. R. Zaki, A. J. Akbarfam, S. Irandoust-Pakchin, *Parameters optimization for the fractional Advection-Dispersion equation using Crank-Nicholson and particle swarm optimization methods*, Fractional Calculus and Applied Analysis, (under review).
[32] T. N. Narasimhan, *Fourier's heat conduction equation, History, influence and connections*, Reviews of Geophysics, (1999), **37**(1), 151–172. https://doi.org/10.1029/1998RG900006.
[33] I. Karatay, N. Kale, S. R. Bayramoglu, *A new difference scheme for time fractional heat equations based on the Crank–Nicholson method*, Fractional Calculus and Applied Analysis, (2013), **16**, 892–910. https://doi.org/ 10.2478/s13540-013-0055-2.
[34] G.R.zaki, A.jodayree Akbarfam, S.Irandoust-Pakchin,*On the inverse problem of time fractional heat equatins by using Crank-Nicholson type methods and genetic algorithm*, Filomat, (2024) **38**(19).
[35] R. Khalil, *A new definition of fractional derivative*, Journal of Computational and Applied Mathematics,(2014), **264**, 65–70. https://doi.org/10.1016/j.cam.2014.01.002.
[36] Angelo B. Mingarelli, *On generalized and fractional derivatives and their applications to classical mechanics*, Journal of Physics A: Mathematical and Theoretical, (2018), **51**, 365204. https://doi.org/10.1088/1751-8121/aad345
[37] A. Zamiri, A. Borhanifar, A. Ghannadias, *Laguerre collocation method for solving Lane-Emden type equations*, Computational Methods for Diferential Equations, (2021), **9**(4), 1176–1197, https://doi.org/10.22034/cmde.2020.35895.1621.
[38] S. Yuzbasi, G. Yildirim, *A Laguerre approach for solving of the systems of linear diferential equations and residual improvement*, Computational Methods for Diferential Equations,(2021), **9**(2) 553–576. https://doi.org/10.22034/cmde.2020.34871.1591.
[39] D. Wang, D. Tan, L. Liu, *Particle swarm optimization algorithm: an overview*, Soft Comput, (2018), **22**, 387–408. https://doi.org/ 10.1007/s00500-016-2474-6
[40] M. R. Dastranj, M. Rouhani, *Design of Optimal Fractional Order PID Controller Using PSO Algorithm*, International Journal of Computer Theory and Engineering, (2015), 175–179. ;https://doi.org/10.7763/IJCTE.2012.V4.499.
[41] X. Zhang, R. Liu, *Adaptive Fractional Image Enhancement Algorithm Based on Rough Set and Particle Swarm Optimization*, Fractal and Fractional, (2022), **6**(2):100. ;https://doi.org/10.3390/fractalfract6020100
[42] B. Bollobas, W. Fulton, A. Katok, F. Kirwan, P. Sarnak, B. Simon, B. Totaro, *Special functions and orthogonal polynomials*, (2016) CAMBRIDGE STUDIES IN ADVANCED MATHEMATICS 153. https://doi.org/10.1017/CBO9781316227381.
[43] Guo Ben-yu, Zhang Xiao-yong, *A new generalized Laguerre spectral approximation and its applications*, Computational and Applied Mathematics, (2005) **184** 382–403, https://doi.org/10.1016/j.cam.2005.01.017.