Sādhanā (2022) 47:253
https://doi.org/10.1007/s12046-022-02005-y

A difference scheme based on cubic B-spline quasi-interpolation for the solution of a fourth-order time-fractional partial integro-differential equation with a weakly singular kernel

M TAGHIPOUR[1] and H AMINIKHAH[1,2,*]

[1]Department of Applied Mathematics and Computer Science, Faculty of Mathematical Sciences, University of Guilan, P.O. Box 1914, Rasht 41938, Iran
[2]Center of Excellence for Mathematical Modelling, Optimization and Combinational Computing (MMOCC), University of Guilan, P.O. Box 1914, Rasht 41938, Iran
e-mail: mtp20222@yahoo.com; hossein.aminikhah@gmail.com

MS received 25 February 2022; revised 26 August 2022; accepted 1 September 2022

Abstract. This paper presents a difference scheme by considering cubic B-spline quasi-interpolation for the numerical solution of a fourth-order time-fractional integro-differential equation with a weakly singular kernel. The fractional derivative of the mentioned equation has been described in the Caputo sense. Time fractional derivative is approximated by a scheme of order $O(\tau^{2-\alpha})$ and the Riemann–Liouville fractional integral term is discretized by the fractional trapezoidal formula. The spatial second derivative has been approximated using the second derivative of the cubic B-spline quasi-interpolation. The discrete scheme leads to the solution of a system of linear equations. We show that the proposed scheme is stable and convergent. In addition, we have shown that the order of convergence is $O(\tau^{2-\alpha}+h^2)$. Finally, various numerical examples are presented to support the fruitfulness and validity of the numerical scheme.

Keywords. Weakly singular kernel; fourth-order time fractional partial integro-differential equation; cubic B-spline quasi-interpolation; finite difference; Riemann–Liouville integral; stability; convergence.

1. Introduction

Partial integro-differential equations with a weakly singular kernel have many applications in various fields of science and engineering, such as heat conduction in materials with memory, viscoelasticity, reactor dynamics, biomechanics, and pressure in porous media [1–4]. Several numerical methods have been used for solving integro-differential equations with a weakly singular kernel. For example, Wang *et al* [5] proposed a high order compact alternating direction implicit scheme for solving two-dimensional time-fractional integro-differential equations with a weakly singularity near the initial time. Qiu *et al* [6] introduced and analyzed a Sinc–Galerkin method for solving the fourth-order partial integro-differential equation with a weakly singular kernel. In [7], the Sinc-collocation approach combined with the double exponential transformation has been employed for solving a class of variable-order fractional integro-partial differential equations. Fakhar–Izadi [8] derived a space-time Spectral–Galerkin method for the solution of one and two-dimensional fourth order time-fractional partial integro-differential equations with a weakly singular kernel. Zhang *et al* [9] proposed the quintic B-spline collocation method for solving fourth order partial integro-differential equations with a weakly singular kernel. Dehestani *et al* [10] applied Legender–Laguerre functions and the collocation method for solving variable-order time-fractional partial integro-differential equations. Hashemizadeh *et al* [11] presented a spectral method for solving nonlinear Volterra integral equations with a weakly singular kernel based on Genocchi polynomials. Biazar and Sadri [12] presented an operational approach based on shifted Jacobi polynomials for solving a class of weakly singular fractional integro-differential equations.

Fractional calculus has proved to be a valuable tool in modeling of different materials and processes in many applied sciences like biology, bio-mechanic, electrochemistry and etc, in accordance with their memory and hereditary properties [13–16]. Various numerical schemes are presented for solving fractional partial differential equations, such as finite difference [17–22], spectral [23–25], meshless [26, 27], and finite element [28, 29] methods.

In this paper, we consider the fourth-order time-fractional integro-differential equation with a weakly singular kernel as follows [30]:

*For correspondence
Published online: 24 November 2022

$$\begin{cases} {}_C\mathcal{D}^{\alpha}_{0,t}u(x,t) - u_{xx}(x,t) - \mathcal{I}^{(\beta)}u_{xx}(x,t) + u_{xxxx}(x,t) = f(x,t), \\ (x,t) \in \Omega, \\ u(x,0) = u^0(x), 0 \le x \le L, \\ u(0,t) = u(L,t) = u_{xx}(0,t) = u_{xx}(L,t) = 0, 0 < t \le T, \end{cases} \tag{1}$$

where $\Omega = (0, L) \times (0, T], 0 < \alpha, \beta < 1, f(x, t)$ is source term and $u^0(x)$ is given smooth function. In fact, problem (1) is equivalent to

$$\begin{cases} {}_C\mathcal{D}^{\alpha}_{0,t}u(x,t) - v(x,t) - \mathcal{I}^{(\beta)}v(x,t) + v_{xx}(x,t) = f(x,t), \\ (x,t) \in \Omega, \\ v(x,t) = u_{xx}(x,t), 0 < x < L, 0 < t \le T, \\ u(x,0) = u^0(x), 0 \le x \le L, \\ u(0,t) = u(L,t) = v(0,t) = v(L,t) = 0, 0 < t \le T. \end{cases} \tag{2}$$

In (2), ${}_C\mathcal{D}^{\alpha}_{0,t}$ is fractional derivative operator in caputo sense and $\mathcal{I}^{(\beta)}$ is defined as follows

$$\mathcal{I}^{(\beta)}u_{xx}(x,t) = \frac{1}{\Gamma(\beta)}\int_0^t (t-s)^{\beta-1}u_{xx}(x,s)ds, t > 0, \tag{3}$$

where $\Gamma(.)$ is the Gamma function.

Equation (1), can be found in the modeling of floor systems, window glasses, airplane wings, and bridge slabs [31, 32]. In fact, fourth-order spatial derivative operators are needed in the modeling of heat flow in materials with memory, strain gradient elasticity, and phase separation in binary mixtures [33–35].

The fourth-order fractional equations have recently attracted the attention of researchers. For example, in [36], the authors proposed a new study for weakly singular kernel fractional fourth-order partial integro-differential equations by means of optimum q-HAM. Tariq and Akram developed a quintic spline technique for time fractional fourth-order partial differential equations [32]. Heydari and Avazzadeh used the orthonormal Bernstein polynomials to solve nonlinear variable-order time fractional fourth-order diffusion-wave equations with nonsingular fractional derivative [37]. Abdelkawy *et al* [38] derived a highly accurate technique for solving distributed-order time-fractional-sub-diffusion equations of the fourth order. Yang *et al* [39] introduced a quasi-wavelet based numerical method for fourth-order partial integro-differential equations with a weakly singular kernel. Roul and Goura considered a high order numerical method for time-fractional fourth order partial differential equations [40].

Cubic B-spline quasi-interpolation has been applied in some papers, see [41–47]. The fundamental benefit of B-spline quasi-interpolation is that they may be built directly without solving any systems of linear equations. It also results in a better approximation of smooth functions. Furthermore, they are local in the sense that the value of B-spline quasi-interpolant at a given point is determined solely by the values of the given function in the neighborhood of that point. Sablonniere [48] found that the cubic B-spline quasi-interpolation first derivative is more accurate than the finite difference approximation. Among the numerical methods so far proposed to solve time-fractional integro-differential equations, B-spline quasi-interpolations have rarely been used. This motivates us to construct a numerical scheme by using cubic B-spline quasi-interpolation to solve equation (1).

In this paper, we construct a difference method using cubic B-spline quasi-interpolation for problem (1). We approximate the temporal Caputo derivative with a L_1-discrete formula. Meanwhile, we apply a second-order formula to approximate $\mathcal{I}^{(\beta)}$ operator. Then we proved the stability and convergence of the difference method. Numerical examples verify the accuracy of the proposed method. Also, the convergence order of the scheme is $(2 - \alpha)$ for time and 2 for space. The advantages of the method are flexibility and simplicity. The method is computationally optimal and fast.

The remainder of the paper is organized as follows. In section 2, we introduce some definitions and preliminaries to fractional calculus and cubic B-spline quasi-interpolation. The difference scheme for the fourth-order time-fractional integro-differential equation with a weakly singular kernel is derived in section 3. The stability and convergence of the method are investigated in Sections 4 and 5. In section 6, some numerical examples are provided to demonstrate the theoretical results. A conclusion ends the article.

2. Some definitions and preliminary

The domain is divided into a uniform grid of mesh points (x_j, t_k) with $x_j = jh, h = \frac{L}{M}, 0 \le j \le M$ and $t_k = k\tau, \ \tau = \frac{T}{N}$, $0 \le k \le N$. The values of the function u at the grid points are denoted $u(x_i, t_k)$ and U_i^k is the approximate solution at the point (x_i, t_k).

Definition 1 The left- and right-sided Riemann–Liouville integrals of a suitably smooth function $f(x)$ on (a, b) are defined by [31, 49, 50]

$${}_{RL}\mathcal{I}^{\alpha}_{a,x}f(x) = \frac{1}{\Gamma(\alpha)}\int_a^x \frac{f(t)}{(x-t)^{n-\alpha}}dt, a < x, n-1 < \alpha \le n, \tag{4}$$

$${}_{RL}\mathcal{I}^{\alpha}_{x,b}f(x) = \frac{1}{\Gamma(\alpha)}\int_x^b \frac{f(t)}{(t-x)^{n-\alpha}}dt, x < b, n-1 < \alpha \le n, \tag{5}$$

respectively.

Definition 2 The left- and right-sided Riemann–Liouville derivatives of order α are defined by [31, 49, 51]

$$\begin{cases} {}_{RL}\mathcal{D}^{\alpha}_{a,x} f(x) = \dfrac{d^n}{dx^n}\left({}_{RL}\mathcal{D}^{-(n-\alpha)}_{a,x} f(x)\right) \\ \qquad\qquad = \dfrac{1}{\Gamma(n-\alpha)}\dfrac{d^n}{dx^n}\displaystyle\int_a^x \frac{f(t)}{(x-t)^{\alpha-n+1}}dt, x > a, \end{cases} \tag{6}$$

and

$$\begin{cases} {}_{RL}\mathcal{D}^{\alpha}_{x,b} f(x) = (-1)^n \dfrac{d^n}{dx^n}\left({}_{RL}\mathcal{D}^{-(n-\alpha)}_{x,b} f(x)\right) \\ \qquad\qquad = \dfrac{(-1)^n}{\Gamma(n-\alpha)}\dfrac{d^n}{dx^n}\displaystyle\int_a^x \frac{f(t)}{(x-t)^{\alpha-n+1}}dt, x < b, \end{cases} \tag{7}$$

respectively, where n is a positive integer satisfying $n-1<\alpha\le n$.

Definition 3 The left- and right-sided Caputo derivatives of order α are defined by [16, 31, 49]

$${}_{C}\mathcal{D}^{\alpha}_{a,x} f(x) = \frac{1}{\Gamma(n-\alpha)}\int_a^x \frac{f^n(t)}{(x-t)^{\alpha-n+1}}dt, a < x, \tag{8}$$

and

$${}_{C}\mathcal{D}^{\alpha}_{x,b} f(x) = \frac{(-1)^n}{\Gamma(n-\alpha)}\int_a^x \frac{f^n(t)}{(t-x)^{\alpha-n+1}}dt, x < b, \tag{9}$$

respectively, where n is a positive integer satisfying $n-1<\alpha\le n$.

Lemma 1 *(L_1 approximation) Let $\alpha\in(0,1)$ and $u(.,t)\in C_t^2([0,T])$ then the following approximation formula holds [52, 53]*

$$\begin{aligned} &{}_{C}\mathcal{D}^{\alpha}_{0,t} u(x,t_k) \\ &= \frac{\tau^{-\alpha}}{\Gamma(2-\alpha)}\left[b_0 u(x,t_k) - \sum_{j=1}^{k-1}(b_{k-j-1}-b_{k-j})u(x,t_j)\right. \\ &\left. - b_{k-1}u(x,t_0)\right] + R, \end{aligned} \tag{10}$$

in which

$$b_j = \left[(l+1)^{1-\alpha} - l^{1-\alpha}\right], 0\le l\le k-1, \tag{11}$$

$$|R| \le C\tau^{2-\alpha} \tag{12}$$

where C is a positive constant given by

$$C = \frac{1}{\Gamma(2-\alpha)}\left[\frac{1-\alpha}{12} + \frac{2^{2-\alpha}}{2-\alpha} - (2^{-\alpha}+1)\right]\max_{t_0\le t\le t_j}|u''(x,t)|.$$

Lemma 2 *Let $\beta\in(0,1)$ and u(., t) is suitably smooth on (0, T) then for the $\mathcal{I}^{(\beta)}$ there holds that [49]*

$$\mathcal{I}^{(\beta)}u(x,t_k) = \sum_{j=0}^{k} a_{j,k}u(x,t_j) + O(\tau^2), \tag{13}$$

where

Now, we introduce B-spline and univariate B-spline

$$a_{j,k} = \frac{\tau^{\beta}}{\Gamma(\beta+2)}\begin{cases} (k-1)^{\beta+1} - (k-1-\beta)k^{\beta}, & j=0, \\ (k-j+1)^{\beta+1} + (k-1-j)^{\beta+1} - 2(k-j)^{\beta+1}, & \\ & 1\le j\le k-1, \\ 1, & j=k. \end{cases}$$

quasi-interpolants that we will use in the next section. In order to define B-splines, we need the concept of knot sequences.

Definition 4 A knot sequence $\boldsymbol{\xi}$ is a nondecreasing sequence of real numbers,

$$\boldsymbol{\xi} := \{\xi_i\}_{i=1}^{m} = \{\xi_1\le\xi_2\le\cdots\le\xi_m\}, m\in\mathbb{N}.$$

The elements ξ_i are called knots.

Provided that $m\ge p+2$ we can define B-splines of degree p over the knot sequence $\boldsymbol{\xi}$.

Definition 5 Suppose for a nonnegative integer p and some integer j that $\xi_{j-p-1}\le\xi_{j-p}\le\cdots\le\xi_j$ are $p+2$ real numbers taken from a knot sequence $\boldsymbol{\xi}$. The j-th B-spline $B_{j,p,\xi}:\mathbb{R}\to\mathbb{R}$ of degree p is identically zero if $\xi_{j-p-1}=\xi_j$ and otherwise defined recursively by [54]

$$B_{j,p,\xi}(x) = \frac{x-\xi_{j-p-1}}{\xi_{j-1}-\xi_{j-p-1}}B_{j-1,p-1,\xi}(x) + \frac{\xi_j - x}{\xi_j - \xi_{j-p}}B_{j,p-1,\xi}(x), \tag{14}$$

starting with

$$B_{i,0,\xi}(x) = \begin{cases} 1, & \text{if} \quad x\in[\xi_{i-1},\xi_i), \\ 0, & \text{otherwise}. \end{cases}$$

A B-spline of degree 3 is also called a cubic B-spline. Using the relation (14), the cubic B-spline $B_{j,3,\xi}$ are given by

$$B_{j,3,\xi}(x) = \begin{cases} \dfrac{(x-\xi_{j-4})^3}{(\xi_{j-3}-\xi_{j-4})(\xi_{j-2}-\xi_{j-4})(\xi_{j-1}-\xi_{j-4})}, \text{if } \xi_{j-4} \le x < \xi_{j-3} \\ \dfrac{(x-\xi_{j-4})^2(\xi_{j-2}-x)}{(\xi_{j-2}-\xi_{j-4})(\xi_{j-2}-\xi_{j-3})(\xi_{j-1}-\xi_{j-4})} \\ \quad + \dfrac{(x-\xi_{j-4})(\xi_{j-1}-x)(x-\xi_{j-3})}{(\xi_{j-1}-\xi_{j-4})(\xi_{j-1}-\xi_{j-3})(\xi_{j-2}-\xi_{j-3})} \\ \quad + \dfrac{(\xi_j-x)(x-\xi_{j-3})^2}{(\xi_j-\xi_{j-3})(\xi_{j-1}-\xi_{j-3})(\xi_{j-2}-\xi_{j-3})}, \text{if } \xi_{j-3} \le x < \xi_{j-3} \\ \dfrac{(x-\xi_{j-4})(\xi_{j-1}-x)^2}{(\xi_{j-1}-\xi_{j-4})(\xi_{j-1}-\xi_{j-3})(\xi_{j-1}-\xi_{j-2})} \\ \quad + \dfrac{(x-\xi_{j-3})(\xi_{j-1}-x)(x-\xi_j)}{(\xi_{j-1}-\xi_{j-3})(\xi_{j-1}-\xi_{j-2})(\xi_j-\xi_{j-3})} \\ \quad + \dfrac{(\xi_j-x)^2(x-\xi_{j-2})}{(\xi_j-\xi_{j-3})(\xi_j-\xi_{j-2})(\xi_{j-1}-\xi_{j-2})}, \text{if } \xi_{j-2} \le x < \xi_{j-1} \\ \dfrac{(\xi_j-x)^3}{(\xi_j-\xi_{j-3})(\xi_j-\xi_{j-2})(\xi_j-\xi_{j-1})}, \text{if } \xi_{j-1} \le x < \xi_j \\ 0, \quad \text{otherwise}. \end{cases} \tag{15}$$

In accordance [54], suppose for integers $n > p \ge 0$ that a knot sequence

$$\boldsymbol{\xi} := \{\xi_i\}_{i=n-p-1}^{n+p} = \{\xi_{n-p-1} \le \xi_{n-p} \le \cdots \le \xi_{n+p}\}, \quad n \in \mathbb{N}, p \in \mathbb{N}_0,$$

is given. This knot sequence allows us to define a set of $n+p$ B-splines of degree p, namely

$$\{B_{1,p,\xi}, \cdots, B_{n+p,p,\xi}\}. \tag{16}$$

We consider the space of splines spanned by the B-splines in (16) over the interval $[\xi_0, \xi_n]$,

$$\mathcal{S}_{p,\xi} := \{s : [\xi_0, \xi_n] \to \mathbb{R} : s = \sum_{j=1}^{n+p} c_j B_{j,p,\xi}, c_j \in \mathbb{R}\}. \tag{17}$$

We now introduce two definitions about knots which are crucial for splines.

Definition 6 A knot sequence $\boldsymbol{\xi}$ is called $(P+1)$-regular if $\xi_{i-p-1} < \xi_i$ for $i = 1, \cdots, n+p$. Such a knot sequence ensures that all the B-splines in (16) are not identically zero [54].

Definition 7 A knot sequence $\boldsymbol{\xi}$ is called $(P+1)$-open on an interval $[a, b]$ if it is $(P+1)$-regular and it has end knots of multiplicity $p+1$,i.e., [54]

$$\begin{aligned} a := \xi_{-p} = \xi_{-p+1} = \cdots = \xi_{-1} = \xi_0 < \xi_1 \le \xi_2 \le \cdots \\ \le \xi_{n-1} < \xi_n \\ = \xi_{n+1} = \cdots = \xi_{n+p} =: b. \end{aligned} \tag{18}$$

Suppose $\{B_{j,p,\xi}\}_{j=1}^{n+p}$ form a basis for $\mathcal{S}_{p,\xi}$. For each $j = 1, \cdots, n+p$, let λ_j be a linear functional defined on $C[a, b]$ that can be computed from values of f at some set of points in $[a, b]$. We have the following definition.

Definition 8 A formula of the form

$$Q_p f(x) := \sum_{j=1}^{n+p} (\lambda_j f) B_{j,p,\xi}(x), \tag{19}$$

is called a B-spline quasi-interpolation formula of degree p [55].

According to [54, 56] the error of a quasi-interpolation satisfies

$$|f(x) - (Q_p f)(x)| \leq \frac{\|Q_p\|}{(p+1)!} \|f^{(p+1)}\|_{\infty,S_x} \Delta(x)^{p+1}, x \in S_\xi^p, \tag{20}$$

where $S_\xi^p = [\xi_0, \xi_n]$, S_x is the union of the supports of all B-splines $B_{i,p,\xi}$, $i \sim x$ and $\|f^{(p+1)}\|_{\infty,S_x}$ denotes the maximum norm of $f^{(p+1)}$ on S_x and $\Delta(x) = \max_{y \in S_x} |y - x|$ that $\sim$ is used to indicate proportionality. If the local mesh ratio is bounded, i.e., if the quotients of the lengths of adjacent knot intervals are $\leq r_y$, then the error of the derivatives on the knot intervals (ξ_0, ξ_n) can be estimated by

$$|f^{(j)}(x) - (Q_p f)^{(j)}(x)| \leq c(p, r_y) \|Q_p\| \|f^{(p+1)}\|_{\infty,S_x} \Delta(x)^{p+1-j}, \tag{21}$$

for $j \leq p$.

Suppose $a = x_0 < \ldots < x_n = b$ are equally spaced points in the interval $[a, b]$. We have the following theorem.

Theorem 1 *Given a function f defined on [a, b], let*

$$\lambda_j f := \begin{cases} f(x_0), & j = 1, \\ \frac{1}{18}(7f(x_0) + 18f(x_1) - 9f(x_2) + 2f(x_3)), & j = 2, \\ \frac{1}{6}(-f(x_{j-3}) + 8f(x_{j-2}) - f(x_{j-1})), & 3 \leq j \leq n+1, \\ \frac{1}{18}(2f(x_{n-3}) - 9f(x_{n-2}) + 18f(x_{n-1}) + 7f(x_n)), & j = n+2, \\ f(x_n), & j = n+3. \end{cases} \tag{22}$$

Then (19) defines a linear operator mapping C[a, b] into $\mathcal{S}_{p,\xi}$ with $Q_p s = s$ for all cubic polynomials s [55].

For approximate derivatives of f by derivatives of $Q_3 f$ up to the order h^3, we can evaluate the value of f' and f'' at x_j by $(Q_3 f)'(x) = \sum_{j=1}^{n+3} (\lambda_j f) B'_{j,p,\xi}(x)$ and $(Q_3 f)''(x) = \sum_{j=1}^{n+3} (\lambda_j f) B''_{j,p,\xi}(x)$. We set $Y = (f_0, f_1, \ldots, f_n)^T$, $Y' = (f'_0, f'_1, \ldots, f'_n)^T$ and $Y'' = (f''_0, f''_1, \ldots, f''_n)^T$ where $f'_j = (Q_3 f)'(x_j)$, $j = 1, \ldots, n$ and $f''_j = (Q_3 f)''(x_j)$, $j = 1, \ldots, n$. The first and the second derivatives of $Q_3(f)$ are calculated as

$$f'_j = \sum_{j=1}^{n+3} (\lambda_j f) B'_{j,p,\xi}(x), j = 0, 1, \ldots, n, \tag{23}$$

$$f''_j = \sum_{j=1}^{n+3} (\lambda_j f) B''_{j,p,\xi}(x), j = 0, 1, \ldots, n, \tag{24}$$

where $B'_{i,p,\xi}(x)$ and $B''_{i,p,\xi}(x)$ are obtained from (15) such that $\xi_j = x_j$ for $j = 0, 1, \ldots, n$. Now according to (23) and (24) we have

$$(Q_3 f)'(x_0) = \frac{1}{h}\left(-\frac{11}{6} f(x_0) + 3f(x_1) - \frac{3}{2} f(x_2) + \frac{1}{3} f(x_3) \right)$$

$$(Q_3 f)'(x_1) = \frac{1}{h}\left(-\frac{1}{3} f(x_0) - \frac{1}{2} f(x_1) + f(x_2) - \frac{1}{6} f(x_3) \right)$$

$$(Q_3 f)'(x_{n-1}) = \frac{1}{h}\left(\frac{1}{6} f(x_{n-3}) - f(x_{n-2}) + \frac{1}{2} f(x_{n-1}) + \frac{1}{3} f(x_n) \right)$$

$$(Q_3 f)'(x_n) = \frac{1}{h}\left(-\frac{1}{3} f(x_{n-3}) + \frac{3}{2} f(x_{n-2}) - 3f(x_{n-1}) + \frac{11}{6} f(x_n) \right)$$

$$(Q_3 f)'(x_j) = \frac{1}{h}\left(\frac{1}{12} f(x_{j-2}) - \frac{2}{3} f(x_{j-1}) + \frac{2}{3} f(x_{j+1}) - \frac{1}{12} f(x_{j+2}) \right), \quad 2 \leq j \leq (n-2),$$

and

$$(Q_3 f)''(x_0) = \frac{1}{h^2}\left(2f(x_0) - 5f(x_1) + 4f(x_2) - f(x_3) \right)$$

$$(Q_3 f)''(x_1) = \frac{1}{h^2}\left(f(x_0) - 2f(x_1) + f(x_2) \right)$$

$$(Q_3 f)''(x_{n-1}) = \frac{1}{h^2}\left(f(x_{n-2}) - 2f(x_{n-1}) + f(x_n) \right)$$

$$(Q_3 f)''(x_n) = \frac{1}{h^2}\left(-f(x_{n-3}) + 4f(x_{n-2}) - 5f(x_{n-1}) + 2f(x_n) \right)$$

$$(Q_3 f)''(x_j) = \frac{1}{h^2}\left(-\frac{1}{6} f(x_{j-2}) + \frac{5}{3} f(x_{j-1}) - 3f(xj) + \frac{5}{3} f(x_{j+1}) - \frac{1}{6} f(x_{j+2}) \right), 2 \leq j \leq (n-2).$$

Therefore, we can display the approximation of f' and f'' in the following matrix form

$$Y' = \frac{1}{h} D_1 Y, Y'' = \frac{1}{h^2} D_2 Y, \tag{25}$$

where $D_1, D_2 \in \mathbb{R}^{(n+1)\times(n+1)}$ are obtained as follows:

$$\begin{aligned} &\mathcal{D}^{\alpha}_{0,t} u(x_i, t_k) - v(x_i, t_k) - \mathcal{I}^{(\beta)} v(x_i, t_k) + v_{xx}(x_i, t_k) = f(x_i, t_k), \\ &v(x_i, t_k) = u_{xx}(x_i, t_k), 1 \le i \le M-1, 1 \le k \le N. \end{aligned} \tag{26}$$

$$D_1 = \begin{pmatrix} -\frac{11}{6} & 3 & -\frac{3}{2} & \frac{1}{3} & 0 & 0 & \dots & 0 & 0 \\ -\frac{1}{3} & -\frac{1}{2} & 1 & -\frac{1}{6} & 0 & 0 & \dots & 0 & 0 \\ \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} & 0 & \dots & 0 & 0 \\ 0 & \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} & 0 \\ 0 & 0 & \dots & 0 & \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} \\ 0 & 0 & \dots & 0 & 0 & \frac{1}{6} & -1 & \frac{1}{2} & \frac{1}{3} \\ 0 & 0 & \dots & 0 & 0 & -\frac{1}{3} & \frac{3}{2} & -3 & \frac{11}{6} \end{pmatrix},$$

$$D_2 = \begin{pmatrix} 2 & -5 & 4 & -1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ -\frac{1}{6} & \frac{5}{3} & -3 & \frac{5}{3} & -\frac{1}{6} & 0 & \dots & 0 & 0 \\ 0 & -\frac{1}{6} & \frac{5}{3} & -3 & \frac{5}{3} & -\frac{1}{6} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & -\frac{1}{6} & \frac{5}{3} & -3 & \frac{5}{3} & -\frac{1}{6} & 0 \\ 0 & 0 & \dots & 0 & -\frac{1}{6} & \frac{5}{3} & -3 & \frac{5}{3} & -\frac{1}{6} \\ 0 & 0 & \dots & 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & \dots & 0 & 0 & -1 & 4 & -5 & 2 \end{pmatrix}.$$

3. Description of the difference scheme

In the present section we construct a difference scheme for solving (1).

Considering (2) at the point (x_i, t_k), one has

Using (10), (13), (23) and (24) equation (26) can be approximated by

$$\begin{aligned} &\frac{\tau^{-\alpha}}{\Gamma(2-\alpha)} \left[b_0 u_i^k - \sum_{j=1}^{k-1} (b_{k-j-1} - b_{k-j}) u_i^j - b_{k-1} u_i^0 \right] \\ &- v_i^k - \sum_{j=0}^{k} a_{j,k} v_i^j + \sum_{j=0}^{M} \frac{d_{ij}^2}{h^2} v_j^k = f_i^k + (R_1)_i^k, \end{aligned} \tag{27}$$

$$v_i^k = \sum_{j=0}^{M} \frac{d_{ij}^2}{h^2} u_j^k + (R_2)_i^k, 1 \le i \le M-1, 1 \le k \le N, \quad (28)$$

where $|(R_1)_i^k| \le C(\tau^{2-\alpha} + h^2)$ and $|(R_2)_i^k| \le Ch^2$.

After simplification we obtain

$$\begin{aligned} u_i^k - (\mu + \mu a_{k,k}) v_i^k + \mu \sum_{j=0}^{M} \frac{d_{ij}^2}{h^2} v_j^k &= \mu f_i^k + \mu \sum_{j=0}^{k-1} a_{j,k} v_i^j + \\ \sum_{j=1}^{k-1} (b_{k-j-1} - b_{k-j}) u_i^j + b_{k-1} u_i^0 + \mu (R_1)_i^k, \\ v_i^k = \sum_{j=0}^{M} \frac{d_{ij}^2}{h^2} u_j^k + (R_2)_i^k, 1 \le i \le M-1, 1 \le k \le N, \end{aligned} \quad (29)$$

where $\mu = \tau^\alpha \Gamma(2-\alpha)$.

Ignoring $(R_1)_i^k, (R_2)_i^k$ and replacing the functions u_i^k and v_i^k with its numerical approximations U_i^k and V_i^k in (29), we obtain the following difference scheme

$$U_i^k - (\mu + \mu a_{k,k}) V_i^k + \mu \sum_{j=0}^{M} \frac{d_{ij}^2}{h^2} V_j^k = \mu f_i^k + \mu \sum_{j=0}^{k-1} a_{j,k} V_i^j \quad (30)$$

$$+ \sum_{j=1}^{k-1} (b_{k-j-1} - b_{k-j}) U_i^j + b_{k-1} U_i^0 \quad (31)$$

$$1 \le i \le M-1, 1 \le k \le N, \quad (32)$$

$$V_i^k = \sum_{j=0}^{M} \frac{d_{ij}^2}{h^2} U_j^k, 1 \le i \le M-1, 1 \le k \le N, \quad (33)$$

$$U_0^k = U_M^k = 0, V_0^k = V_M^k = 0, 1 \le k \le N, \quad (34)$$

$$U_i^0 = U^0(x_i), 1 \le i \le M. \quad (35)$$

We set $l_1 = \mu + \mu a_{k,k}$ and $l_2 = \frac{\mu}{h^2}$. So that in each time step we encounter the following system of linear equations

$$AU^k = F^k, \quad (36)$$

where

$$A = \begin{pmatrix} I & B \\ C & I \end{pmatrix}, U^k = \begin{pmatrix} U_1^k \\ U_2^k \\ \vdots \\ U_{M-2}^k \\ U_{M-1}^k \\ V_1^k \\ V_2^k \\ \vdots \\ V_{M-2}^k \\ V_{M-1}^k \end{pmatrix}, F^k = \begin{pmatrix} F^1 \\ F^2 \end{pmatrix},$$

such that B and C are pentadiagonal matrices and I is identity matrix

$$B = \begin{pmatrix} -l_1 - 2l_2 & l_2 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ \frac{5}{3}l_2 & -l_1 - 3l_2 & \frac{5}{3}l_2 & -\frac{1}{6}l_2 & 0 & 0 & \dots & 0 & 0 \\ -\frac{1}{6}l_2 & \frac{5}{3}l_2 & -l_1 - 3l_2 & \frac{5}{3}l_2 & -\frac{1}{6}l_2 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -\frac{1}{6}l_2 & \frac{5}{3}l_2 & -l_1 - 3l_2 & \frac{5}{3}l_2 & -\frac{1}{6}l_2 \\ 0 & 0 & \dots & 0 & 0 & -\frac{1}{6}l_2 & \frac{5}{3}l_2 & -l_1 - 3l_2 & \frac{5}{3}l_2 \\ 0 & 0 & \dots & 0 & 0 & 0 & 0 & l_2 & -l_1 - 2l_2 \end{pmatrix},$$

$$C=\begin{pmatrix} \frac{2}{h^2} & -\frac{1}{h^2} & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ -\frac{5}{3h^2} & \frac{3}{h^2} & -\frac{5}{3h^2} & \frac{1}{6h^2} & 0 & 0 & \dots & 0 & 0 \\ \frac{1}{6h^2} & -\frac{5}{3h^2} & \frac{3}{h^2} & -\frac{5}{3h^2} & \frac{1}{6h^2} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & \frac{1}{6h^2} & -\frac{5}{3h^2} & \frac{3}{h^2} & -\frac{5}{3h^2} & \frac{1}{6h^2} \\ 0 & 0 & \dots & 0 & 0 & \frac{1}{6h^2} & -\frac{5}{3h^2} & \frac{3}{h^2} & -\frac{5}{3h^2} \\ 0 & 0 & \dots & 0 & 0 & 0 & 0 & \frac{2}{h^2} & \frac{1}{h^2} \end{pmatrix}, F^1$$

$$=\begin{pmatrix} \mu f_1^k+\mu\sum_{j=0}^{k-1}a_{j,k}V_1^k+\sum_{j=1}^{k-1}(b_{k-j-1}-b_{k-j})U_1^k+b_{k-1}U_1^0-l_2V_0^k \\ \mu f_2^k+\mu\sum_{j=0}^{k-1}a_{j,k}V_2^k+\sum_{j=1}^{k-1}(b_{k-j-1}-b_{k-j})U_2^k+b_{k-1}U_2^0+\frac{l_2}{6}V_0^k \\ \mu f_3^k+\mu\sum_{j=0}^{k-1}a_{j,k}V_3^k+\sum_{j=1}^{k-1}(b_{k-j-1}-b_{k-j})U_3^k+b_{k-1}U_3^0 \\ \vdots \\ \mu f_{M-2}^k+\mu\sum_{j=0}^{k-1}a_{j,k}V_{M-2}^k+\sum_{j=1}^{k-1}(b_{k-j-1}-b_{k-j})U_{M-2}^k+b_{k-1}U_{M-2}^0+\frac{l_2}{6}V_M^k \\ \mu f_{M-1}^k+\mu\sum_{j=0}^{k-1}a_{j,k}V_{M-1}^k+\sum_{j=1}^{k-1}(b_{k-j-1}-b_{k-j})U_{M-1}^k+b_{k-1}U_{M-1}^0-l_2V_M^k \end{pmatrix}, F^2=\begin{pmatrix} \frac{U_0^k}{h^2} \\ -\frac{U_0^k}{6h^2} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \frac{U_M^k}{6h^2} \\ \frac{U_M^k}{h^2} \end{pmatrix}.$$

4. Stability analysis

In the current section, the stability of the scheme (31)–(33) can be analyzed by using the Fourier method [57]. We assume that the exact solution u is continuous and the derivative of u is square integrable. Let $\tilde{U}_j^k$ be the approximate solution of the scheme, and define

$\zeta_j^k = U_j^k - \tilde{U}_j^k, \qquad 1\le j\le M-1, \quad 1\le k\le N,$

with corresponding vector

$$\zeta^k=(\zeta_1^k,\zeta_2^k,\dots,\zeta_{M-1}^k)^T.$$

Thanks to (31)–(33) we have

$$\begin{aligned} &U_i^k-s\left(-\frac{1}{6}U_{i-2}^k+\frac{5}{3}U_{i-1}^k-3U_i^k+\frac{5}{3}U_{i+1}^k-\frac{1}{6}U_{i+2}^k\right)\\ &\quad+r\left(-\frac{1}{6}V_{i-2}^k+\frac{5}{3}V_{i-1}^k-3V_i^k+\frac{5}{3}V_{i+1}^k-\frac{1}{6}V_{i+2}^k\right)\\ &\quad=\mu f_i^k+\mu\sum_{j=0}^{k-1}a_{j,k}V_i^j+\sum_{j=1}^{k-1}(b_{k-j-1}-b_{k-j})U_i^j+b_{k-1}U_i^0, \end{aligned}$$

$$s=\frac{\mu+\mu a_{k,k}}{h^2}, r=\frac{\mu}{h^2}.$$

So that

$$\begin{aligned} &U_i^k-s\left(-\frac{1}{6}U_{i-2}^k+\frac{5}{3}U_{i-1}^k-3U_i^k+\frac{5}{3}U_{i+1}^k-\frac{1}{6}U_{i+2}^k\right)\\ &\quad-\frac{r}{6h^2}\left(-\frac{1}{6}U_{i-4}^k+\frac{5}{3}U_{i-3}^k-3U_{i-2}^k+\frac{5}{3}U_{i-1}^k-\frac{1}{6}U_i^k\right)\\ &\quad\frac{5r}{3h^2}\left(-\frac{1}{6}U_{i-3}^k+\frac{5}{3}U_{i-2}^k-3U_{i-1}^k+\frac{5}{3}U_i^k-\frac{1}{6}U_{i+1}^k\right)\\ &\quad-\frac{3r}{h^2}\left(-\frac{1}{6}U_{i-2}^k+\frac{5}{3}U_{i-1}^k-3U_i^k+\frac{5}{3}U_{i+1}^k-\frac{1}{6}U_{i+2}^k\right)\\ &\quad+\frac{5r}{3h^2}\left(-\frac{1}{6}U_{i-1}^k+\frac{5}{3}U_i^k-3U_{i+1}^k+\frac{5}{3}U_{i+2}^k-\frac{1}{6}U_{i+3}^k\right)\\ &\quad-\frac{r}{6h^2}\left(-\frac{1}{6}U_i^k+\frac{5}{3}U_{i+1}^k-3U_{i+2}^k+\frac{5}{3}U_{i+3}^k-\frac{1}{6}U_{i+4}^k\right)\\ &\quad=\mu f_i^k+\mu\sum_{j=0}^{k-1}\frac{a_{j,k}}{h^2}\left(-\frac{1}{6}U_{i-2}^k+\frac{5}{3}U_{i-1}^k-3U_i^k\right.\\ &\quad\left.+\frac{5}{3}U_{i+1}^k-\frac{1}{6}U_{i+2}^k\right)\\ &\quad+\sum_{j=1}^{k-1}(b_{k-j-1}-b_{k-j})U_i^j+b_{k-1}U_i^0. \end{aligned}$$

Set

$$\lambda_1 = \frac{r}{36h^2}, \lambda_2 = -\frac{10r}{18h^2}, \lambda_3 = \frac{s}{6} + \frac{r}{h^2} + \frac{25r}{9h^2},$$
$$\lambda_4 = -\frac{5s}{3} - \frac{10r}{18h^2} - \frac{10r}{h^2}, \lambda_5 = 1 + 3s + \frac{r}{18h^2} + \frac{50r}{9h^2} + \frac{9r}{h^2}$$

then we have

$$\begin{aligned}
&\lambda_1 U_{i-4}^k + \lambda_2 U_{i-3}^k + \lambda_3 U_{i-2}^k + \lambda_4 U_{i-1}^k \\
&\quad + \lambda_5 U_i^k + \lambda_4 U_{i+1}^k + \lambda_3 U_{i+2}^k + \lambda_2 U_{i+3}^k + \lambda_1 U_{i+4}^k \\
&\quad = \mu f_i^k + \mu \sum_{j=0}^{k-1} \frac{a_{j,k}}{h^2} \left(-\frac{1}{6} U_{i-2}^k + \frac{5}{3} U_{i-1}^k - 3U_i^k \right. \\
&\quad \left. + \frac{5}{3} U_{i+1}^k - \frac{1}{6} U_{i+2}^k \right) \\
&\quad + \sum_{j=1}^{k-1} (b_{k-j-1} - b_{k-j}) U_i^j + b_{k-1} U_i^0.
\end{aligned} \tag{37}$$

Next, we define the grid functions as follows:

$$\zeta^k(x) = \begin{cases} \zeta_j^k, & x_j - \frac{h}{2} < x \le x_j + \frac{h}{2}, \\ 0, & 0 \le x \le \frac{h}{2} or L - \frac{h}{2} < x \le L. \end{cases} \tag{38}$$

We can expand $\zeta^k(x)$ into a Fourier series

$$\zeta^k(x) = \sum_{l=-\infty}^{\infty} d_k(l) e^{i2\pi lx/L}, \tag{39}$$

where

$$d_k(l) = \frac{1}{L} \int_0^L \zeta^k(x) e^{-i2\pi lx/L} \, \mathrm{d}x. \tag{40}$$

Denoting

$$\|\zeta^k\|_2 = \left(\int_0^L \|\zeta^k(x)\|^2 \, \mathrm{d}x \right)^{\frac{1}{2}}, \tag{41}$$

and using the Parseval equality

$$\int_0^L \|\zeta^k(x)\|^2 \, \mathrm{d}x = \sum_{l=-\infty}^{\infty} \|d_k(l)\|^2, \tag{42}$$

one has

$$\|\zeta^k\|^2 = \sum_{l=-\infty}^{\infty} \|d_k(l)\|^2. \tag{43}$$

We can expand ζ_j^k into Fourier series, and Because the difference equations are linear, we can analyze the behavior of the total error by tracking the behavior of an arbitrary nth component [58]. So we can assume that the solution of (37) has the following form

$$\zeta_j^k = d_k e^{i\sigma_x jh},$$

where $\sigma_x = 2\pi l/L$. Substituting the above expression into (37) we obtain

$$d_k = \frac{b_{k-1}}{z} d_0 + \frac{\sum_{j=1}^{k-1} (b_{k-j-1} - b_{k-j}) d_j}{z} + \frac{rs' \sum_{j=0}^{k-1} a_{j,k} d_j}{z}, \tag{44}$$

where

$$\begin{aligned}
s' &= -\frac{1}{3}\cos(2\sigma_x h) + \frac{10}{3}\cos(\sigma_x h) - 3 \le 0, \\
z &= 2\lambda_1 \cos(4\beta h) + 2\lambda_2 \cos(3\beta h) \\
&\quad + 2\lambda_3 \cos(2\beta h) + 2\lambda_4 \cos(\beta h) + \lambda_5.
\end{aligned}$$

Theorem 2 *Suppose that* d_k, $(1 \le k \le N-1)$ *are defined by (44), then we obtain*

$$|d_k| \le C_k |d_0|, \quad k = 1, 2, \cdots, N-1.$$

Proof We will prove this claim by mathematical induction. For $k = 1$ we prove that there exist a constant C_1 such that

$$|d_1| = |d_0| \frac{|1 + rs' a_{0,1}|}{|z|} \le C_1 |d_0|.$$

For this purpose, we have

$$\begin{aligned}
1 + rs' a_{0,1} &= 1 + \frac{\mu}{h^2} a_{0,1} \left(-\frac{1}{3} \left(1 - 2\sigma_x^2 h^2 + O(h^4) \right) \right. \\
&\quad \left. + \frac{10}{3} \left(1 - \frac{\sigma_x^2 h^2}{2} + O(h^4) \right) - 3 \right) \\
&= 1 + \frac{\mu}{h^2} a_{0,1} \left(\frac{2\sigma_x^2 h^2}{3} - \frac{5\sigma_x^2 h^2}{3} + O(h^4) \right) \\
&= 1 - \mu a_{0,1} \sigma_x^2 + \mu a_{0,1} O(h^2),
\end{aligned}$$

and

$$\begin{aligned}
z &= \frac{\mu}{18h^4} \left(1 - 8\beta^2 h^2 + O(h^4) \right) \\
&\quad - \frac{10\mu}{9h^4} \left(1 - \frac{9\beta^2 h^2}{2} + O(h^4) \right) \\
&\quad + \left(\frac{\mu + \mu a_{1,1}}{3h^2} + \frac{2\mu}{h^4} + \frac{50\mu}{9h^4} \right) \left(1 - 2\beta^2 h^2 + O(h^4) \right) \\
&\quad + \left(\frac{-10\mu - 10\mu a_{1,1}}{3h^2} - \frac{10\mu}{9h^4} - \frac{20\mu}{h^4} \right) \\
&\quad \times \left(1 - \frac{\beta^2 h^2}{2} + O(h^4) \right) + 1 \\
&\quad + \frac{3\mu + 3\mu a_{1,1}}{h^2} + \frac{\mu}{18h^4} + \frac{50\mu}{9h^4} + \frac{9\mu}{h^4}
\end{aligned}$$

$$
\begin{aligned}
&= \frac{\mu}{18h^4} - \frac{10\mu}{18h^4} + \frac{2\mu}{h^4} + \frac{50\mu}{9h^4} - \frac{10\mu}{9h^4} - \frac{20\mu}{h^4} + \frac{\mu}{18h^4} \\
&\quad + \frac{50\mu}{9h^4} + \frac{9\mu}{h^4} - \frac{8\mu\beta^2}{18h^2} + \frac{5\mu\beta^2}{h^2} - \left(\frac{2\mu + 2\mu a_{1,1}}{3}\right)\beta^2 \\
&\quad + \left(\frac{5\mu + 5\mu a_{1,1}}{3}\right)\beta^2 + \frac{3\mu + 3\mu a_{1,1}}{h^2} + \frac{\mu + \mu a_{1,1}}{3h^2} \\
&\quad - \left(\frac{10\mu + 10\mu a_{1,1}}{3h^2}\right) \\
&\quad + \frac{\mu}{18}O(1) - \frac{10\mu}{9}O(1) + \frac{68\mu}{9}O(1) - \frac{190\mu}{9}O(1) \\
&\quad + \left(\frac{\mu + \mu a_{1,1}}{3}\right)O(h^2) \\
&\quad - \left(\frac{10\mu + 10\mu a_{1,1}}{3}\right)O(h^2) + 1 \\
&= -\frac{8\mu\beta^2}{18h^2} + \frac{5\mu\beta^2}{h^2} + (\mu + \mu a_{1,1})\beta^2 - \frac{263\mu}{18}O(1) \\
&\quad - (3\mu + 3\mu a_{1,1})O(h^2) + 1.
\end{aligned}
$$

We take the limit from $\frac{1+rs'a_{0,1}}{z}$ as $\tau \to 0$ and $h \to 0$ in such a way that we maintain the ratio $\frac{\mu}{h^2} = \frac{\tau^\alpha \Gamma(2-\alpha)}{h^2}$ equal to a fixed constant H. So that

$$
\frac{1 + rs'a_{0,1}}{z} \to \frac{1}{\frac{41}{9}\beta^2 H + 1} = H_1.
$$

As a result, there is a positive constant C_1 independent of N, M that says

$$
\left|\frac{1 + rs'a_{0,1}}{z}\right| \le C_1.
$$

Assume that

$$
|d_n| \le C_n|d_0|, 1 \le n \le k-1.
$$

We have

$$
|d_k| \le \frac{b_{k-1}|d_0| + \sum_{j=1}^{k-1}(b_{k-j-1} - b_{k-j})|d_j| + |rs'||\sum_{j=0}^{k-1} a_{j,k}||d_j|}{|z|}.
$$

Now assume that

$$
C' = \max\{C_1, C_2, \ldots, C_{k-1}\}, C'' > C', C'' \ge 1, \qquad (45)
$$

so similar to initial case $k = 1$, we obtain

$$
\begin{aligned}
|d_k| &\le \frac{b_{k-1}C''|d_0| + \sum_{j=1}^{k-1}(b_{k-j-1} - b_{k-j})C''|d_0| + |rs'|C''|\sum_{j=0}^{k} a_{j,k}||d_0|}{|z|} \\
&= \frac{\left(b_{k-1} + \sum_{j=1}^{k-1}(b_{k-j-1} - b_{k-j})\right)C''|d_0| + |rs'|C''|\sum_{j=0}^{k} a_{j,k}||d_0|}{|z|} \\
&= \frac{\left(C'' + |rs'|C''|\sum_{j=0}^{k} a_{j,k}|\right)|d_0|}{|z|} \le C_k|d_0|,
\end{aligned}
$$

This completes the proof. □

Theorem 3 *The finite difference scheme (31)–(35) is unconditionally stable for* $\alpha \in (0, 1)$.

Proof Thanks to theorem (2) and Parseval's equality, we obtain

$$
\|U^k - \tilde{U}^k\|_{l^2}^2 = \|\zeta^k\|_{l^2}^2 \le C_k^2\|\zeta^0\|_{l^2}^2,
$$

so that

$$
\|U^k - \tilde{U}^k\|_{l^2} \le C\|U^0 - \tilde{U}^0\|_{l^2},
$$

which indicates that the numerical scheme is stable. □

5. Convergence

In this section, we prove that convergence of the difference scheme (31)–(35). Similar to the previous section let $e_j^k = u_j^k - U_j^k, 1 \le j \le M-1, 0 \le k \le N-1$ and and denote, $e^k = (e_1^k, e_2^k, \ldots, e_{M-1}^k)^T, \mathbf{R}^k = (R_1^k, R_2^k, \ldots, R_{M-1}^k)^T$, $0 \le k \le N-1$.

From Equations (31)–(35) and $R_j^{k+1} = O(\tau^{2-\alpha} + h^2)$ and noticing that $e_j^0 = 0$, similar to (37) one has

$$
\begin{aligned}
&\lambda_1 e_{i-4}^k + \lambda_2 e_{i-3}^k + \lambda_3 e_{i-2}^k + \lambda_4 e_{i-1}^k + \lambda_5 e_i^k + \lambda_4 e_{i+1}^k + \lambda_3 e_{i+2}^k \\
&\quad + \lambda_2 e_{i+3}^k + \lambda_1 e_{i+4}^k \\
&= \mu f_i^k + \mu \sum_{j=0}^{k-1} \frac{a_{j,k}}{h^2}\left(-\frac{1}{6}e_{i-2}^k + \frac{5}{3}e_{i-1}^k - 3e_i^k + \frac{5}{3}e_{i+1}^k - \frac{1}{6}e_{i+2}^k\right) \\
&\quad + \sum_{j=1}^{k-1}(b_{k-j-1} - b_{k-j})e_i^j + \mu R_i^k.
\end{aligned} \qquad (46)
$$

Using the similar idea of stability analysis, we define the following functions

$$
e^k(x) = \begin{cases} e_j^k, & x_j - \frac{h}{2} < x \le x_j + \frac{h}{2}, 1 \le j \le M-1, \\ 0, & 0 \le x \le \frac{h}{2} or L - \frac{h}{2} < x \le L. \end{cases} \qquad (47)
$$

and

$$
R^k(x) = \begin{cases} R_j^k, & x_j - \frac{h}{2} < x \le x_j + \frac{h}{2}, 1 \le j \le M-1, \\ 0, & 0 \le x \le \frac{h}{2} or L - \frac{h}{2} < x \le L. \end{cases} \qquad (48)
$$

We expand the $e^k(x)$ and $R^k(x)$ into the following Fourier series expansions

$$e^k(x) = \sum_{l=-\infty}^{\infty} \eta_k(l) e^{i2\pi lx/L}, R^k(x) = \sum_{l=-\infty}^{\infty} \xi_k(l) e^{i2\pi lx/L}, \tag{49}$$

where

$$\eta_k(l) = \frac{1}{L}\int_0^L e^k(x) e^{-i2\pi lx/L}\,\mathrm{d}x, \xi_k(l) = \frac{1}{L}\int_0^L R^k(x) e^{-i2\pi lx/L}\,\mathrm{d}x. \tag{50}$$

Applying the Parseval equality

$$\int_0^L \|e^k(x)\|^2\,\mathrm{d}x = \sum_{l=-\infty}^{\infty} \|\eta_k(l)\|^2, \int_0^L \|R^k(x)\|^2\,\mathrm{d}x = \sum_{l=-\infty}^{\infty} \|\xi_k(l)\|^2, \tag{51}$$

and

$$\int_0^L \|e^k(x)\|^2\,\mathrm{d}x = \sum_{j=1}^{M-1} h\|e_j^k\|^2, \int_0^L \|R^k(x)\|^2\,\mathrm{d}x = \sum_{j=1}^{M-1} h\|R_j^k\|^2, \tag{52}$$

we have

$$\|e^k\|_2^2 = \sum_{l=-\infty}^{\infty} \|\eta_k(l)\|^2, \|R^k\|_2^2 = \sum_{l=-\infty}^{\infty} \|\xi_k(l)\|^2. \tag{53}$$

Now, we suppose that

$$e_j^k = \eta_k e^{i\sigma_x jh},$$
$$R_j^k = \xi_k e^{i\sigma_x jh},$$

where $\sigma_x = \frac{2l\pi}{L}$. By replacing the above relations into (46) leads to

$$\eta_k = \frac{\sum_{j=1}^{k-1}(b_{k-j-1} - b_{k-j})\eta_j}{z} + \frac{rs'\sum_{j=0}^{k-1} a_{j,k}\eta_j}{z} + \frac{\mu\xi_k}{z}. \tag{54}$$

Lemma 3 *(Discrete Gronwall inequality) Let y_n and g_n be nonnegative sequences and b be a nonnegative constant. If [59]*

$$y_n \le b + \sum_{0\le k<n} g_k y_k, n \ge 0,$$

then

$$y_n \le \prod_{0\le j<n}(1+g_j) \le b\exp\left(\sum_{0\le j<n} g_j\right).$$

Theorem 4 *If η_k be the solution of Equation (54), then there is positive constant C such that*

$$|\eta_k| \le C|\xi_1|. \tag{55}$$

Proof In view of the convergence of the series on the right-hand side of Equation (53), we know that there exists a positive constant C_2, such that

$$|\xi_k| \le C_2|\xi_1|, k = 1, 2, \ldots, N-1. \tag{56}$$

According to Equations (54), (56) and theorem (2), we have

$$\begin{aligned}|\eta_k| &\le \sum_{j=1}^{k-1}(b_{k-j-1} - b_{k-j}) \times \frac{|\eta_j|}{|z|} + \sum_{j=0}^{k-1} a_{j,k}\frac{|\eta_j||rs'|}{|z|} + \frac{\mu|\xi_j|}{|z|}\\ &\le \left(C_1\sum_{j=0}^{k-1}(b_{k-j-1} - b_{k-j}) + C_2\sum_{j=0}^{k-1} a_{j,k}\right)|\eta_j| + C_3|\xi_1|\\ &\le C_3|\xi_1|\exp(C_1(1-b_k) + C_2C_4)\\ &\le C_3|\xi_1|\exp(C_1 + C_2C_4) = C|\xi_1|.\end{aligned}$$

This completes the proof. □

Theorem 5 *The difference scheme (31)–(35) is convergent, and the order of convergence is $O(\tau^{2-\alpha} + h^2)$.*

Proof By theorem (4) and Equation (56), we can obtain

$$\begin{aligned}\|e^k\|_{l^2}^2 &= \sum_{l=-\infty}^{\infty} \|\eta_k(l)\|^2 \le \sum_{l=-\infty}^{\infty} C^2\|\xi_1(l)\|^2\\ &= C^2\sum_{l=-\infty}^{\infty} \|\xi_1(l)\|^2 = C^2\|R^1\|_{l^2}^2,\end{aligned}$$

furthermore, there exists a positive constant C_1, such that

$$\begin{aligned}R_i^k \le C_1(\tau^{2-\alpha} + h^2) \Rightarrow ||R^k|| &\le C_1\sqrt{(M-1)h}(\tau^{2-\alpha} + h^2)\\ &\le C_1\sqrt{L}(\tau^{2-\alpha} + h^2)\end{aligned}$$

So that

$$\|e^k\|_{l^2} \le C\|R^1\|_{l^2} \le C'(\tau^{2-\alpha} + h^2),$$

where $C' = C\sqrt{L}$. This completes the proof. □

6. Numerical experiments

In this section, five test problems are presented to check the effectiveness, validity, stability, and convergence orders of the present method. The domain in all examples is $\Omega = [0,1]\times[0,1]$. All computations are implemented with MATLAB R2020b. The error norms used in this section are as follows:

$$||e||_\infty = \max_{0\le i\le N, 0\le j\le M} |u(x_i,t_j) - U(x_i,t_j)|,$$
$$\|e(\tau,h)\| = \|e^N\| = \left(\Delta x\sum_{j=1}^{M}(e_j^N)^2\right)^{\frac{1}{2}},$$

where $e_j^k = u(x_j,t_k) - U_j^k$. In all examples we have used the following formulas to calculate the convergence rate:

Table 1. L2-norm errors and order of convergence for $\alpha = 0.1, 0.3, 0.5$ and $\beta = 0.1, 0.15, 0.45$ for example 1.

h	τ $\frac{1}{55}$	$\alpha = 0.1$ $\|e^N\|$	$\beta = 0.1$ $r_1(\tau, h)$	τ $\frac{1}{135}$	$\alpha = 0.3$ $\|e^N\|$	$\beta = 0.15$ $r_1(\tau, h)$	τ $\frac{1}{15}$	$\alpha = 0.5$ $\|e^N\|$	$\beta = 0.45$ $r_1(\tau, h)$
$\frac{1}{10}$		9.6233e−03			9.9334e−03			1.0321e−02	
$\frac{1}{20}$		2.6307e−03	1.8711		2.6851e−03	1.8873		2.7430e−03	1.9116
$\frac{1}{40}$		6.4975e−04	2.0175		6.6955e−04	2.0037		6.7759e−04	2.0173
$\frac{1}{80}$		1.5008e−04	2.1141		1.6518e−04	2.0191		1.6533e−04	2.0350
$\frac{1}{160}$		2.5577e−05	2.5528		4.0002e−05	2.0459		3.8777e−05	2.0921
$\frac{1}{320}$		5.4572e−06	2.2286		8.8637e−06	2.1741		7.3685e−06	2.3958

Table 2. L2-norm errors and order of convergence for $\alpha = 0.1, 0.3, 0.95$ and $\beta = 0.65, 0.45, 0.15$ for example 1.

τ	h $\frac{1}{1000}$	$\alpha = 0.1$ $\|e^N\|$	$\beta = 0.65$ $r_2(\tau, h)$	h $\frac{1}{1000}$	$\alpha = 0.3$ $\|e^N\|$	$\beta = 0.45$ $r_2(\tau, h)$	h $\frac{1}{1000}$	$\alpha = 0.95$ $\|e^N\|$	$\beta = 0.15$ $r_2(\tau, h)$
$\frac{1}{4}$		3.6258e−04			2.5677e−04			4.0930e−05	
$\frac{1}{8}$		1.1178e−04	1.6976		7.7851e−05	1.7217		2.4556e−05	0.7370
$\frac{1}{16}$		3.3758e−05	1.7273		2.3047e−05	1.7561		1.2870e−05	0.9321
$\frac{1}{32}$		9.6178e−06	1.8115		6.2854e−06	1.8745		6.0982e−06	1.0775

Table 3. L2-norm errors and orders for $\tau = 1/109$ and $\alpha = 0.9, \beta = 0.75$ for example 1.

h	$\alpha = 0.9$ $\|e^N\|$	$\beta = 0.75$ $r_1(\tau, h)$
$\frac{1}{10}$	1.0717e−02	
$\frac{1}{20}$	2.7944e−03	1.9393
$\frac{1}{40}$	6.7617e−04	2.0471
$\frac{1}{80}$	1.5544e−04	2.1210
$\frac{1}{160}$	2.7379e−05	2.5052
$\frac{1}{320}$	4.3300e−06	2.6606

$$r_1(\tau, h) = \log_2\left(\frac{\|e(\tau, 2h)\|}{\|e(\tau, h)\|}\right), r_2(\tau, h) = \log_2\left(\frac{\|e(2\tau, h)\|}{\|e(\tau, h)\|}\right).$$

Example 1 For the first example, consider the following problem:

$$c\mathcal{D}^{\alpha}_{0,t}u(x,t) - u_{xx}(x,t) - \mathcal{I}^{(\beta)}u_{xx}(x,t) + u_{xxxx}(x,t) = f(x,t),$$

with the initial condition $u^0(x) = 0$. The source term is

$$f(x,t) = \left(\frac{\Gamma(\alpha+\beta+1)}{\Gamma(\beta+1)}t^{-\alpha} + \frac{\pi^2\Gamma(\alpha+\beta+1)}{\Gamma(2\beta+1)}t^{\beta} + \pi^2 + \pi^4\right)t^{\alpha+\beta}\sin(\pi x).$$

The exact solution is

$$u(x,t) = t^{\alpha+\beta}\sin(\pi x).$$

In tables 1 and 3, we record the norm of errors and convergence orders in spatial direction for different values of α and β . In table 2, the orders of convergence with respect to time for different values of α and β are reported. For each value of α and β, we chose different spatial step sizes $h = 1/10, 1/20, \ldots, 1/320$ and a fixed temporal step length of τ to obtain the numerical convergence rates in

Table 4. L_∞-norm errors with $h = 1/512$ for example 1.

τ	$\alpha = 0.15$ $\|e\|_\infty$	$\beta = 0.95$ CPU	$\|e\|_\infty^{[30]}$	$\alpha = 0.95$ $\|e\|_\infty$	$\beta = 0.85$ CPU	$\|e\|_\infty^{[30]}$
$\frac{1}{16}$	1.8425e−05	0.5749s	9.444e−05	5.7083e−04	0.5517s	5.632e−04
$\frac{1}{32}$	9.2283e−06	1.2213s	2.410e−05	2.9111e−04	1.1902s	2.882e−04
$\frac{1}{64}$	6.7322e−06	3.1247s	6.012e−06	1.5331e−04	3.0141s	1.529e−04
$\frac{1}{128}$	6.0623e−06	9.1477s	1.732e−06	7.7794e−05	8.9472s	7.770e−05

Table 5. Error for different T, α and fixed h, τ, β for example 1.

T	h	τ	$\alpha = 0.4, \beta = 0.5$ $\|e^N\|$	$\alpha = 0.7, \beta = 0.5$ $\|e^N\|$
2	$\frac{1}{500}$	$\frac{1}{10}$	$5.6081e{-}06$	$2.2204e{-}05$
4	$\frac{1}{500}$	$\frac{1}{10}$	$4.7115e{-}06$	$3.5576e{-}05$
8	$\frac{1}{500}$	$\frac{1}{10}$	$1.2735e{-}05$	$4.8432e{-}05$
10	$\frac{1}{500}$	$\frac{1}{10}$	$2.6216e{-}05$	$7.5994e{-}05$

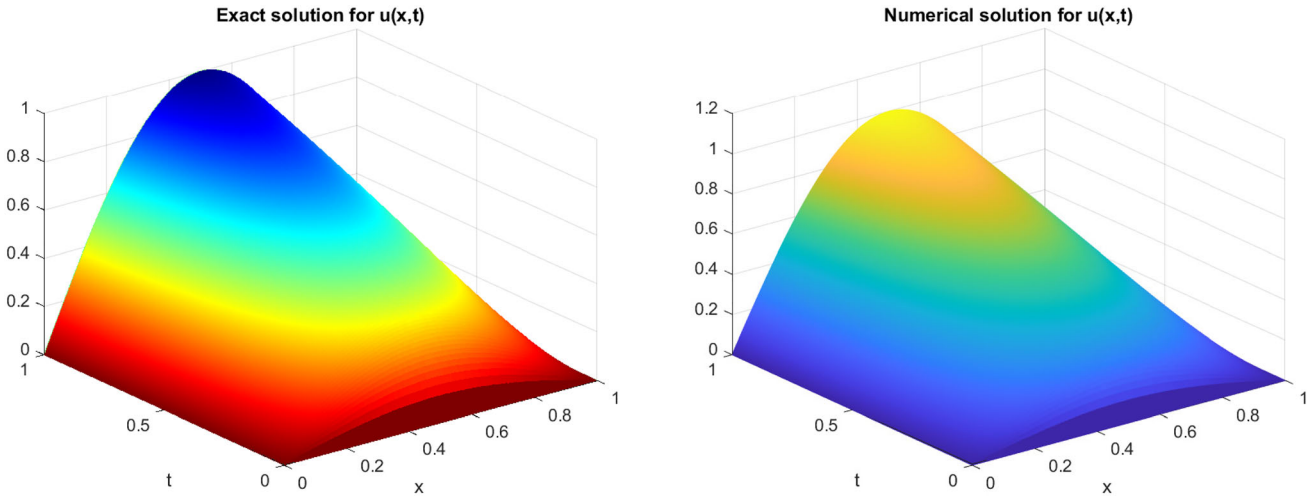

Figure 1. The graph of exact (left) and numerical (right) solutions at $\tau = 1/55$ and $h = 1/320$ with $\alpha = \beta = 0.1$ for Example 1.

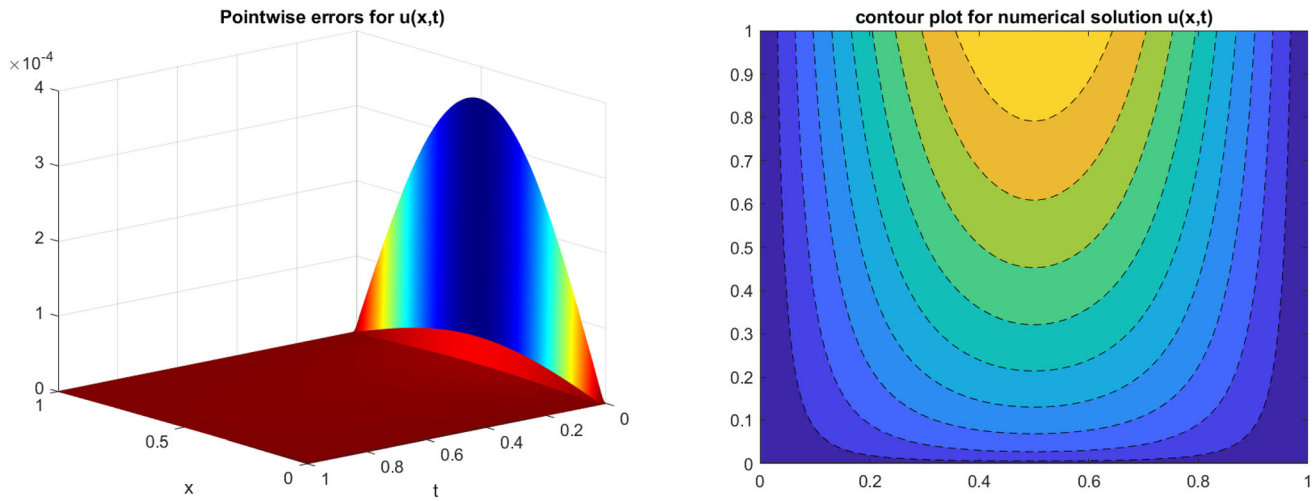

Figure 2. The surface of absolute pointwise errors when $\tau = 1/55$, $h = \frac{1}{320}$ and contour plot of numerical solution for Example 1.

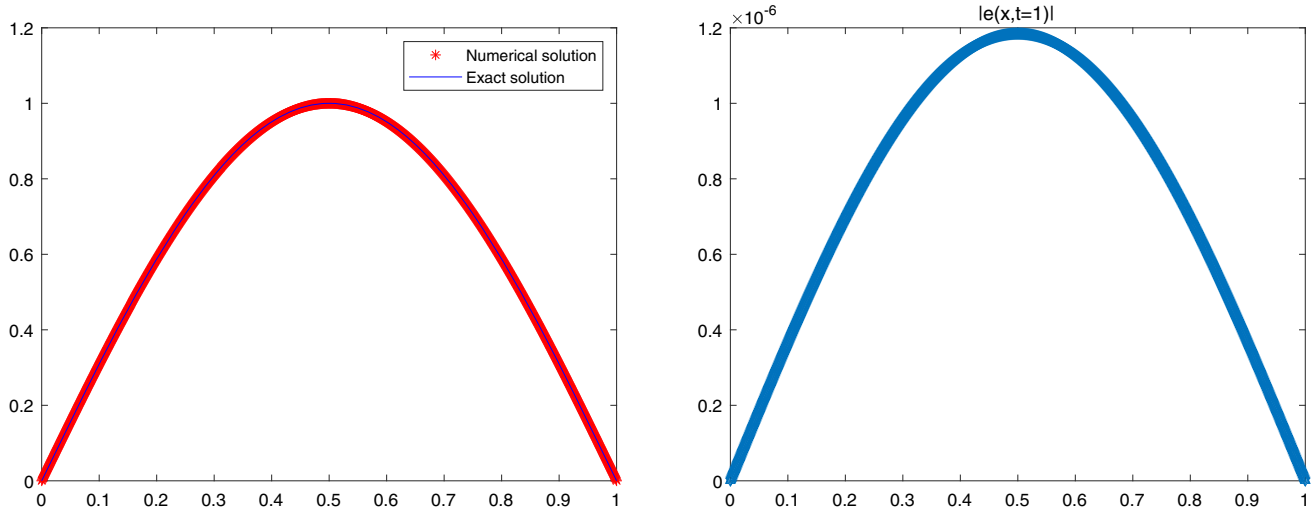

Figure 3. The comparison between numerical solution and exact solution (left) and pointwise absolute error (right) with $\tau = 1/55$ and $h = \frac{1}{320}$ at $t = 1$ for example 1.

Table 6. L2-norm errors and order of convergence for $\alpha = 0.1, 0.4, 0.8$ and $\beta = 0.3, 0.5, 0.6$ when $\tau = 1/277$ for example 2.

	$\alpha = 0.1$	$\beta = 0.3$		$\alpha = 0.4$	$\beta = 0.5$		$\alpha = 0.8$	$\beta = 0.6$	
h	$\|e^N\|$	$r_1(\tau, h)$	CPU	$\|e^N\|$	$r_1(\tau, h)$	CPU	$\|e^N\|$	$r_1(\tau, h)$	CPU
$\frac{1}{10}$	1.0339e−02		0.7798s	1.0563e−02		0.7663s	1.0646e−02		0.7657s
$\frac{1}{20}$	2.7348e−03	1.9186	1.4044s	2.7753e−03	1.9283	1.3889s	2.7890e−03	1.8867	1.3777s
$\frac{1}{40}$	6.7373e−04	2.0212	2.5325s	6.8187e−04	2.0251	2.3544s	6.8486e−04	2.0717	2.5309s
$\frac{1}{80}$	1.6383e−04	2.0400	4.8787s	1.6701e−04	2.0296	4.6083s	1.6718e-04	2.0344	4.8575s
$\frac{1}{160}$	3.8025e−05	2.1072	9.6636s	4.0284e−05	2.0516	9.0576s	3.9813e−05	2.0701	9.6751s
$\frac{1}{320}$	6.8213e−06	2.4788	20.1019s	8.8901e−06	2.1799	19.0211s	8.2699e−06	2.2673	20.320s

Table 7. L_∞-norm errors with $h = 1/512$ for example 2.

	$\alpha = 0.50$	$\beta = 0.35$		$\alpha = 0.55$	$\beta = 0.15$		$\alpha = 0.15$	$\beta = 0.60$	
τ	$\|e\|_\infty$	CPU	$\|e\|_\infty^{[30]}$	$\|e\|_\infty$	CPU	$\|e\|_\infty^{[30]}$	$\|e\|_\infty$	CPU	$\|e\|_\infty^{[30]}$
$\frac{1}{16}$	1.3292e−04	0.5127s	1.400e−04	1.2294e−04	0.5598s	1.268e−04	6.9553e−05	0.5420s	6.930e−05
$\frac{1}{32}$	5.0593e−05	1.0761s	5.562e−05	5.4439e−05	1.2022s	5.829e−05	2.0456e−05	1.1891s	2.340e−05
$\frac{1}{64}$	1.8086e−05	2.6267s	2.237e−05	2.3390e−05	3.0994s	2.763e−05	4.0925e−06	3.1243s	7.567e−06
$\frac{1}{128}$	5.0688e−06	7.4859s	9.078e−06	9.0937e−06	9.1954s	1.329e−05	1.3550e−06	9.1544s	72.492e−06

Table 8. Error for different T, α and fixed h, τ, β for example 2.

T	h	τ	$\alpha = 0.4, \beta = 0.5$ $\|e^N\|$	$\alpha = 0.7, \beta = 0.5$ $\|e^N\|$
2	$\frac{1}{500}$	$\frac{1}{10}$	1.3528e−06	5.0532e−07
4	$\frac{1}{500}$	$\frac{1}{10}$	5.2240e−06	3.9042e−06
8	$\frac{1}{500}$	$\frac{1}{10}$	9.4774e−06	8.6232e−06
10	$\frac{1}{500}$	$\frac{1}{10}$	1.0983e−05	1.0252e−05

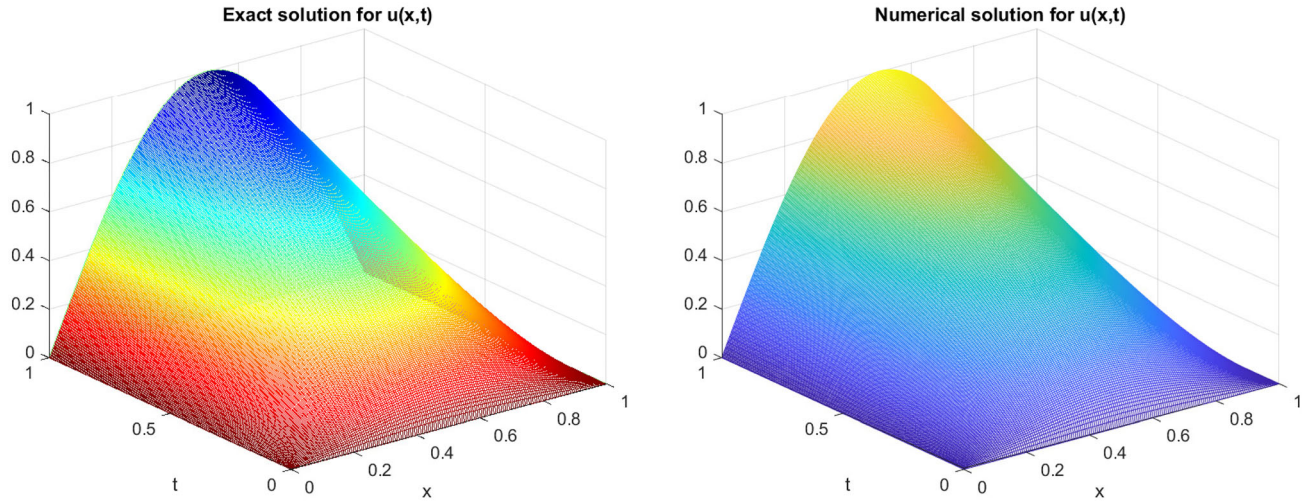

Figure 4. The exact (left) and numerical (right) solutions at $\tau = 1/277$ and $h = 1/640$ with $\alpha = 0.8, \beta = 0.6$ for example 2.

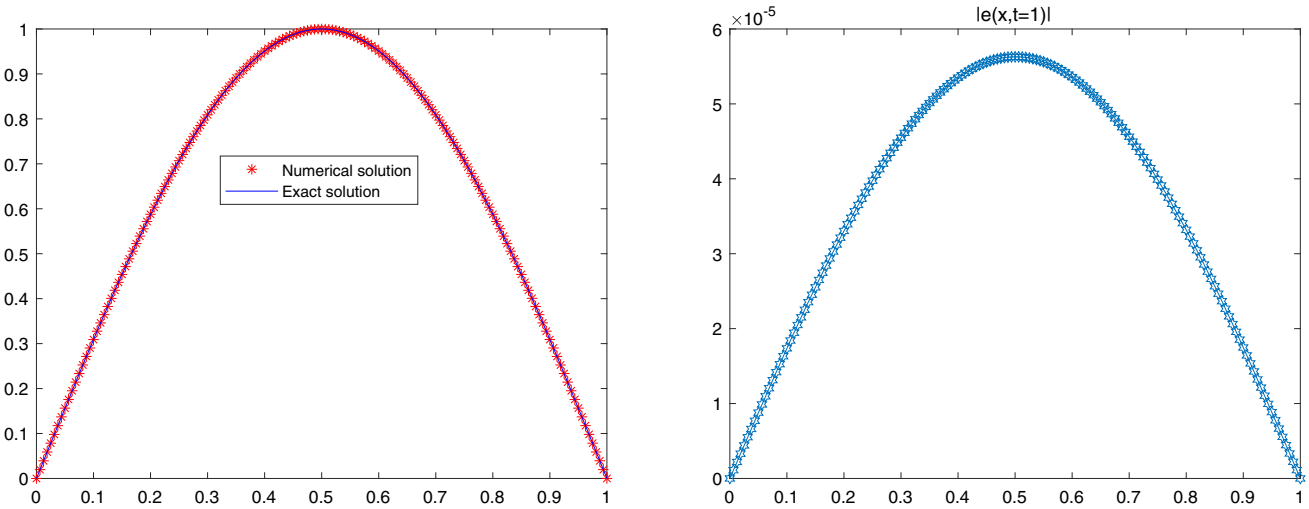

Figure 5. The comparison between numerical solution and exact solution (left) and pointwise absolute error (right) with $\tau = 1/277$ and $h = \frac{1}{640}$ at $t = 1$ for example 2.

spatial, which is in excellent agreement with our theoretical results. In table 4, we compared our results with the reference [30]. In table 5, we presented the results for large time instant t.

Figure 1 compares the plots of the exact and numerical sulotions computed by difference scheme using $\tau = \frac{1}{55}$ and $h = \frac{1}{320}$. The plot of pointwise errors and the contour plot of numerical solution at $t = 1$ with $\tau = \frac{1}{55}$ and $h = \frac{1}{320}$ is illustrated in figure 2. In figure 3, a comparison between the numerical and exact solutions at $t = 1$ with $\tau = \frac{1}{55}$ and $h = \frac{1}{320}$ is demonstrated. It can be seen from tables 1 and 3 that, when spatial step sizes decrease, we obtain better results. In tables 1, 2, and 3 the CPU time is almost 2 seconds. All the figures show that the numerical scheme is efficient and effective.

Example 2 Consider the problem (1) with exact solution $u(x,t) = t^{\beta}\sin(\pi x), (x,t) \in \Omega$. The source term is taken as

$$f(x,t) = \left(\frac{\Gamma(\beta+1)}{\Gamma(\beta+1-\alpha)}t^{-\alpha} + \frac{\pi^2\Gamma(\beta+1)}{\Gamma(2\beta+1)}t^{\beta} + \pi^2 + \pi^4\right)t^{\beta}\sin(\pi x).$$

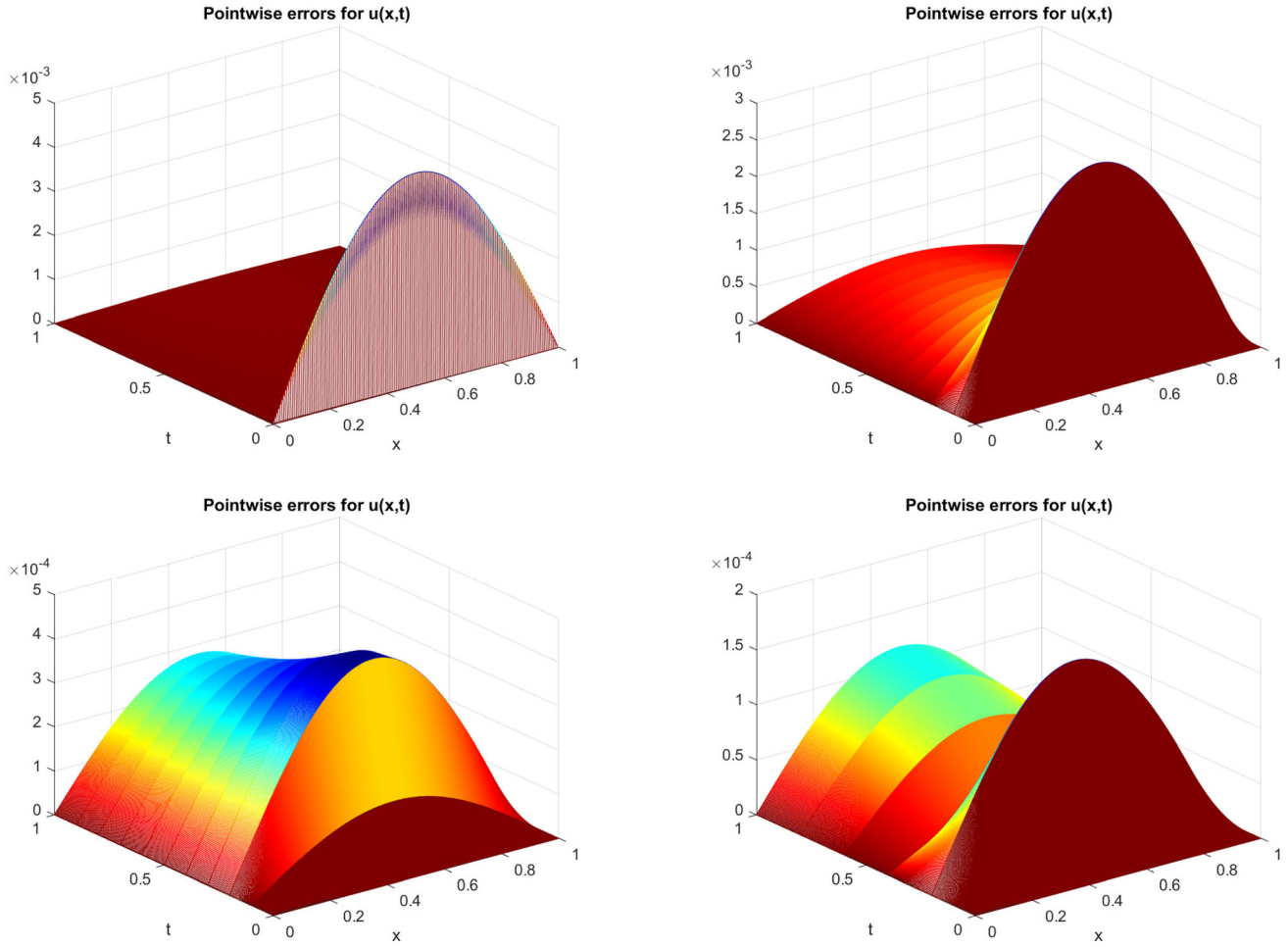

Figure 6. Poinwise errors for Example 2: top left $(\alpha = 0.8, \beta = 0.6, \tau = 1/10, h = 1/640)$, top right $(\alpha = 0.1, \beta = 0.3, \tau = 1/277, h = 1/640)$, bottom left $(\alpha = 0.4, \beta = 0.5, \tau = 1/10, h = 1/640)$, bottom right $(\alpha = 0.6, \beta = 0.9, \tau = 1/5, h = 1/640)$.

Table 9. L2-norm errors and order of convergence for $\alpha = 0.1, 0.5, 0.7$ and $\beta = 0.1, 0.5, 0.7$ for example 3.

h	τ	$\alpha = 0.1$ $\|e^N\|$	$\beta = 0.1$ $r_1(\tau, h)$	τ	$\alpha = 0.5$ $\|e^N\|$	$\beta = 0.5$ $r_1(\tau, h)$	τ	$\alpha = 0.7$ $\|e^N\|$	$\beta = 0.7$ $r_1(\tau, h)$
$\frac{1}{10}$	$\frac{1}{40}$	4.0189e−03		$\frac{1}{40}$	4.0180e−03		$\frac{1}{95}$	4.0259e−02	
$\frac{1}{20}$		1.1401e−03	1.8176		1.1517e−03	1.8276		1.1566e−03	1.7994
$\frac{1}{40}$		2.9085e−04	1.9708		2.9551e−04	1.9625		2.9706e−04	1.9611
$\frac{1}{80}$		7.3102e−05	1.9923		7.4336e−05	1.9913		7.4659e−05	1.9924
$\frac{1}{160}$		1.8346e−05	1.9932		1.8479e−05	2.0080		1.8460e−05	2.0159
$\frac{1}{320}$		4.6520e−06	1.9808		4.4587e−06	2.0512		4.3457e−06	2.0867
$\frac{1}{640}$		1.2216e−06	1.9291		9.4657e−07	2.2358		8.0916e−07	2.4251

Table 10. L2-norm errors and order of convergence for $\alpha = 0.1, 0.2, 0.95$ and $\beta = 0.71, 0.85, 0.15$ for Example 3.

τ	h	$\alpha = 0.1$ $\|e^N\|$	$\beta = 0.71$ $r_2(\tau, h)$	h	$\alpha = 0.2$ $\|e^N\|$	$\beta = 0.85$ $r_2(\tau, h)$	h	$\alpha = 0.95$ $\|e^N\|$	$\beta = 0.15$ $r_2(\tau, h)$
$\frac{1}{3}$	$\frac{1}{2000}$	2.3836e−05		$\frac{1}{2000}$	2.0196e−05		$\frac{1}{1000}$	4.4053e−05	
$\frac{1}{6}$		6.1586e−06	1.9525		4.9076e−06	2.0409		2.5523e−05	0.7874
$\frac{1}{12}$		1.6320e−06	1.9159		1.2385e−06	1.9864		1.3373e−05	0.9325
$\frac{1}{24}$		4.9616e−07	1.7178		3.7713e−07	1.7155		6.5759e−06	1.0241

Table 11. Error for different T, α and fixed h, τ, β for Example 3.

T	h	τ	$\alpha = 0.4, \beta = 0.5$ $\|e^N\|$	$\alpha = 0.7, \beta = 0.5$ $\|e^N\|$
2	$\frac{1}{200}$	$\frac{1}{10}$	9.8215e−05	8.7838e−05
4	$\frac{1}{200}$	$\frac{1}{10}$	7.5296e−04	7.3353e−04
8	$\frac{1}{200}$	$\frac{1}{10}$	5.7694e−03	5.7400e−03
10	$\frac{1}{200}$	$\frac{1}{10}$	1.1099e−02	1.1070e−02

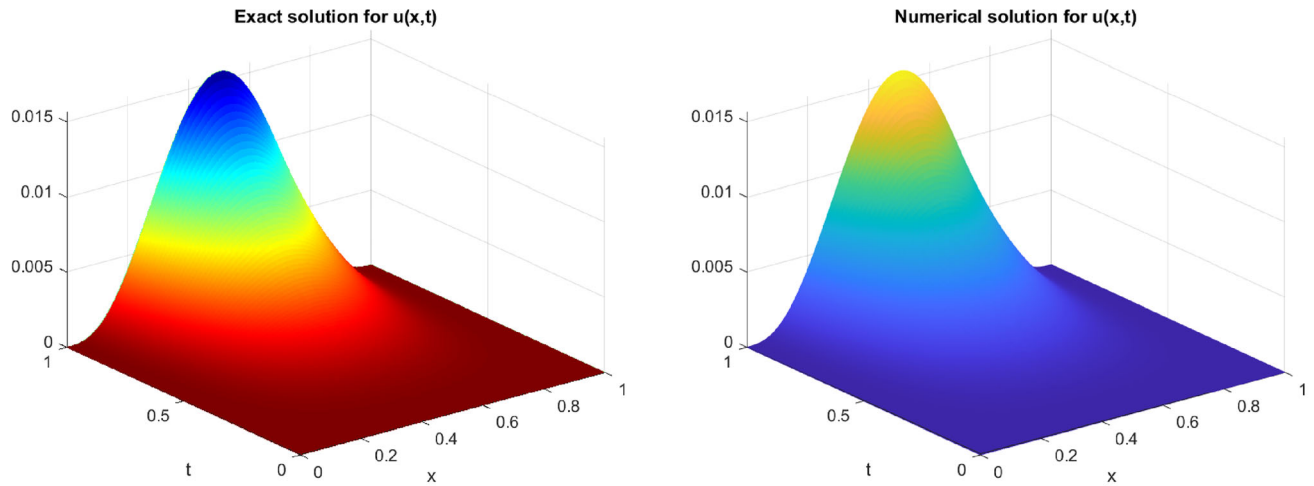

Figure 7. The exact (left) and numerical (right) solutions curves at $\tau = 1/95$ and $h = 1/1280$ with $\alpha = 0.7, \beta = 0.7$ for example 3.

In table 6, we list

L2-norm errors and experiment order of convergence for the difference scheme. Herein, we take $\tau = \frac{1}{277}$ and choose different spatial step sizes for different values of α and β . Also, the convergence rate in space is seen to be about 2. The CPU time is less than 30 seconds. In table 7, we compared our results with those of the reference [30]. In table 8, we presented the results for large time instant t. Figure 4 shows the exact and numerical solutions. Figure 5 presents the exact and numerical solutions and absolute error at $t = 1$. In figure 6, we depicted graph of pointwise errors for different values of α, β, τ, h. It is apparent from tables and figures that the numerical scheme works well.

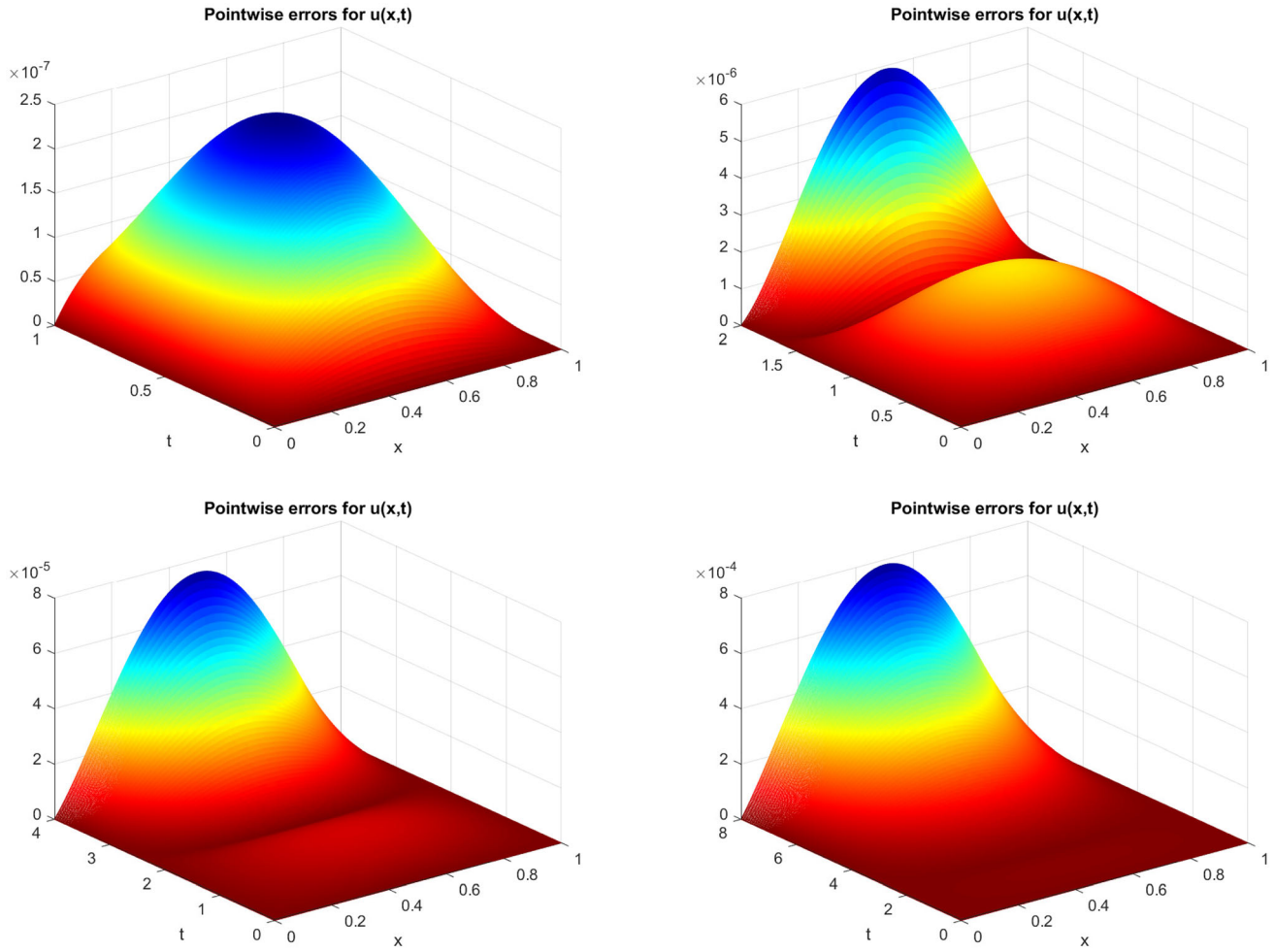

Figure 8. Poinwise errors for Example 3: top left $(\alpha = 0.7, \beta = 0.7, \tau = 1/95, h = 1/1280, T = 1)$, top right $(\alpha = 0.9, \beta = 0.1, \tau = 0.02, h = 1/640, T = 2)$, bottom left $(\alpha = 0.9, \beta = 0.1, \tau = 0.04, h = 1/640, T = 4)$, bottom right $(\alpha = 0.8, \beta = 0.4, \tau = 0.08, h = 1/640, T = 8)$.

Example 3 In this example the exact solution of the problem (1) is given by $u(x,t) = t^3x^3(1-x)^3$ and the inhomogeneous term is

$$f(x,t) = \frac{6x^3(1-x)^3t^{3-\alpha}}{\Gamma(4-\alpha)} - 6t^3x(-5x^3+10x^2-6x+1) - \frac{36x(-5x^3+10x^2-6x+1)t^{\beta+3}}{\Gamma(\beta+4)} - 72(5x^2-5x+1)t^3.$$

Tables 9 and 10 give the L2-norm errors and convergence orders using the present numerical method. It is observed that the numerical solutions of the numerical scheme are seen to be in good agreement with the exact ones. In table 11, we presented the results for large time instant t. The CPU time is less than 12 seconds. In figure 8, surfaces of pointwise error are portrayed at different $\alpha, \beta, \tau, h, T$. In figure 7, the numerical solution and exact solution curves have been demonstrated. In figure 9, the comparison between $u(x_j, t_k)$ and U_j^k at $t = 1$ is created to show the efficiency of the presented method. Also, the CPU time illustrates that the proposed scheme is fast.

Example 4 Consider fourth-order time-fractional integro-differential equation with a weakly singular kernel (1) with exact solution $u(x,t) = t^2e^xx^3(1-x)^3, (x,t) \in \Omega$. The source term is taken as

$$f(x,t) = \frac{2t^{2-\alpha}e^xx^3(1-x)^3}{\Gamma(3-\alpha)} + t^2e^xx(x^5+9x^4+3x^3-37x^2+30x-6) + \frac{2e^xx(x^5+9x^4+3x^3-37x^2+30x-6)}{\Gamma(\beta+3)}t^{\beta+2} - t^xe^x(x^6+21x^5+123x^4+167x^3 - 156x^2-108x+48).$$

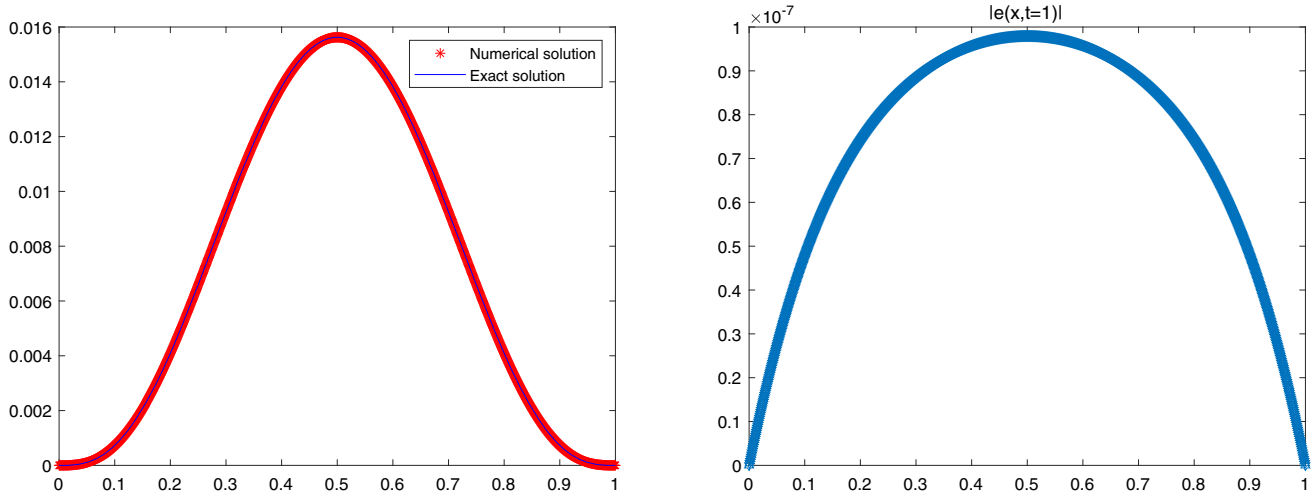

Figure 9. The comparison between numerical solution and exact solution (left) and absolute error (right) with $\tau = 1/277, h = \frac{1}{1280}$ and $\alpha = 0.7, \beta = 0.7$ at $t = 1$ for example 3.

Table 12. L2-norm errors and order of convergence for $\alpha = 0.1, 0.6, 0.8$ and $\beta = 0.1, 0.3, 0.5$ when $\tau = 1/40$ for Example 4.

h	$\alpha = 0.1$ $\|e^N\|$	$\beta = 0.1$ $r_1(\tau, h)$	$\alpha = 0.6$ $\|e^N\|$	$\beta = 0.3$ $r_1(\tau, h)$	$\alpha = 0.8$ $\|e^N\|$	$\beta = 0.5$ $r_1(\tau, h)$
$\frac{1}{10}$	5.8631e−03		5.7861e−03		5.7812e−02	
$\frac{1}{20}$	2.1882e−03	1.4219	2.1813e−03	1.4074	2.1907e−03	1.3999
$\frac{1}{40}$	6.0335e−04	1.8587	6.0458e−04	1.8512	6.0829e−04	1.8486
$\frac{1}{80}$	1.5495e−04	1.9612	1.5552e−04	1.9588	1.5617e−04	1.9616
$\frac{1}{160}$	3.9074e−05	1.9875	3.9049e−05	1.9937	3.8712e−05	2.0122
$\frac{1}{320}$	9.8318e−06	1.9907	9.5980e−06	2.0245	8.9835e−06	2.1074
$\frac{1}{640}$	2.5002e−06	1.9754	2.2062e−06	2.1212	1.5212e−06	2.5621
$\frac{1}{1280}$	6.6552e−07	1.9095	3.5654e−07	2.6694	3.6583e−07	2.0560

Table 13. Error for different τ, T, α and fixed h, β for Example 4.

T	h	τ	$\alpha = 0.3$ $\|e^N\|$	$\alpha = 0.8$ $\|e^N\|$
0.5	$\frac{1}{640}$	$\frac{1}{20}$	8.2658e−07	1.6247e−06
1	$\frac{1}{640}$	$\frac{1}{10}$	4.0473e−06	1.2768e−06
2	$\frac{1}{640}$	$\frac{1}{40}$	9.9581e−06	8.9517e−06
4	$\frac{1}{640}$	$\frac{1}{40}$	3.8448e−05	3.7578e−05
8	$\frac{1}{640}$	$\frac{1}{10}$	1.5223e−04	1.4828e−04

From table 12, we can see that by decreasing h, more accurate results can be achieved. In table 13, L-2 norm errors are demonstrated for $\alpha = 0.3, 0.8$. It is clear from table 12 that the presented method is accurate with a good order of convergence. The CPU time is less than 12 seconds. Figure 10 shows that the exact and numerical solutions are the same. The plot of pointwise error and comparison between numerical and exact solutions at $t = 1$ are illustrated in figure 11. All the tables and figures clearly show that the present difference scheme is impressive in term of accuracy.

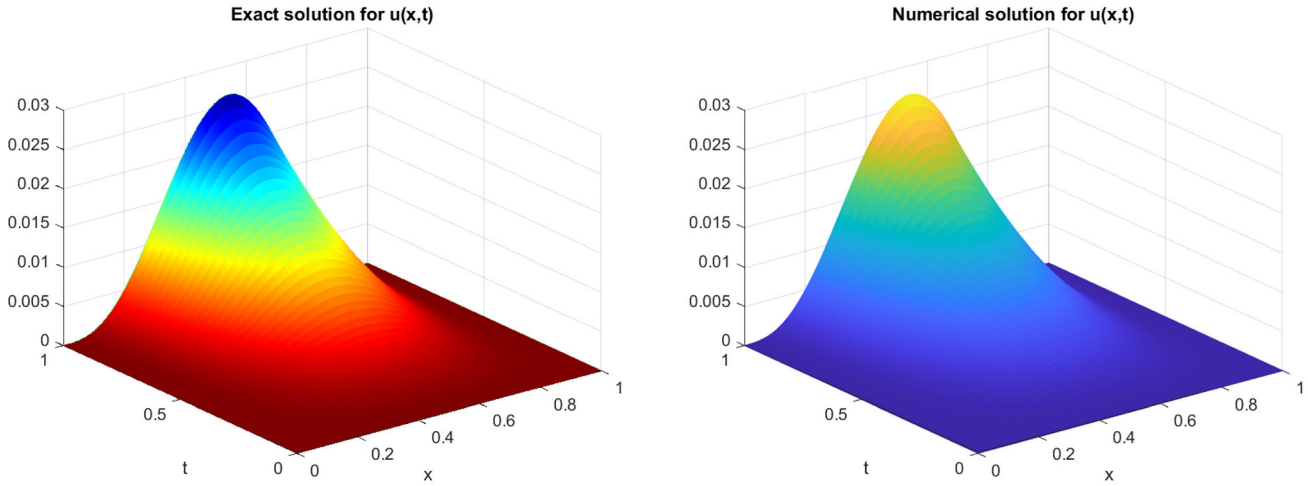

Figure 10. The exact (left) and numerical (right) solutions plots at $\tau = 1/40$ and $h = 1/1280$ with $\alpha = 0.8, \beta = 0.5$ for example 4.

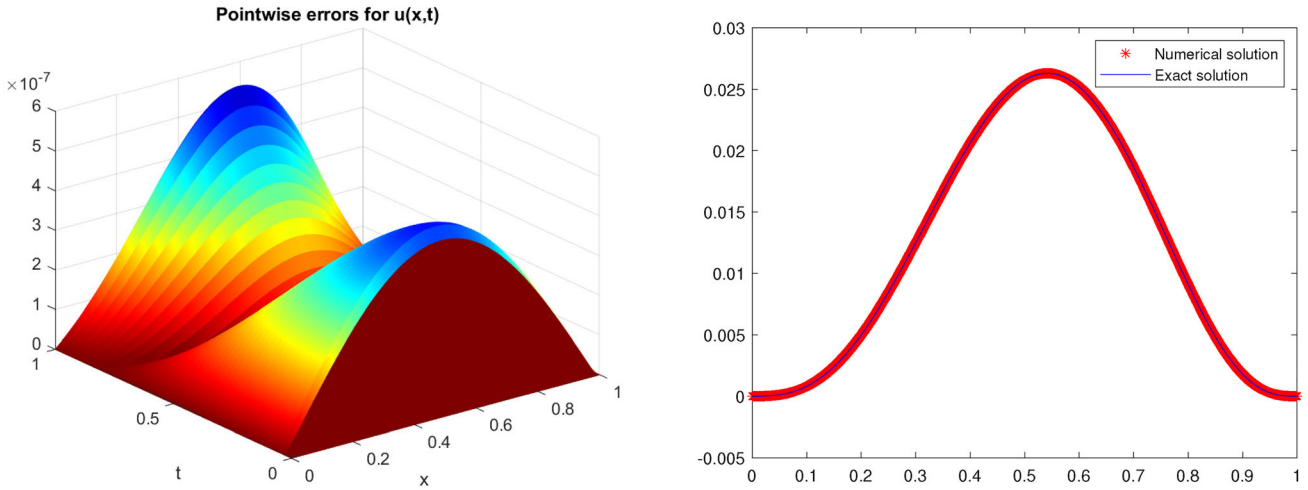

Figure 11. The poinwise error (left) and comparison between numerical and exact solutions (right) for example 4.

Table 14. L2-norm errors for $\alpha = \beta = 0.1, 0.3, 0.7$ for Example 5.

h	τ	$\alpha = \beta = 0.1$ $\|e^N\|$	$\alpha = \beta = 0.3$ $\|e^N\|$	$\alpha = \beta = 0.7$ $\|e^N\|$
$\frac{1}{10}$	$\frac{1}{10}$	$2.0264e{-}03$	$2.0339e{-}03$	$2.0805e{-}03$
$\frac{1}{20}$	$\frac{1}{20}$	$4.2266e{-}04$	$4.2396e{-}04$	$4.3339e{-}04$
$\frac{1}{40}$	$\frac{1}{40}$	$3.7887e{-}05$	$3.8003e{-}05$	$3.8765e{-}05$
$\frac{1}{80}$	$\frac{1}{80}$	$3.1906e{-}06$	$3.1950e{-}06$	$3.2162e{-}06$

Table 15. Error for different T, α and fixed h, τ, β for Example 5.

T	h	τ	$\alpha = 0.4, \beta = 0.5$ $\|e^N\|$	$\alpha = 0.7, \beta = 0.5$ $\|e^N\|$
2	$\frac{1}{100}$	$\frac{1}{100}$	2.1160e−06	2.6063e−06
4	$\frac{1}{100}$	$\frac{1}{100}$	2.7389e−06	4.1694e−06
6	$\frac{1}{100}$	$\frac{1}{100}$	3.1801e−06	5.4769e−06
10	$\frac{1}{100}$	$\frac{1}{100}$	3.8303e−06	7.7039e−06

Example 5 In the last example, we consider $u(x,t) = (x^2 - x)^4 \sin(\pi x) t^\alpha$.

In table 14, L-2 norm errors are reported for $\alpha = \beta = 0.1, 0.3, 0.7$. In table 15, numerical results are presented with large time instant $t, (T = 2, 4, 6, 10)$. Tables 14 and 15 verify the efficiency of the proposed method.

7. Conclusions

In this paper, we presented a difference scheme using cubic B-spline quasi-interpolation for the numerical solution of a fourth-order time-fractional integro-differential equation with a weakly singular kernel. The time fractional derivative of the mentioned equation is approximated by a scheme of order $O(\tau^{2-\alpha})$ and the spatial derivative is replaced with a second order approximation. The fractional integral is approximated by polynomial interpolation. In terms of the implementation and speed of the method, it is easy to apply and almost fast. We have proved the stability and convergence of the numerical method with the order of convergence $O(\tau^{2-\alpha}+h^2)$. Five test problems have been performed to show the convergence orders, applicability, and capability of the method. All numerical computations are obtained by using MATLAB R2020b.

List of symbols

$_{RL}\mathcal{I}^{\alpha}_{a,x}$ Riemann–Liouville integral
$_{C}\mathcal{D}^{\alpha}_{a,x}$ Caputo fractional derivative

Acknowledgements

We sincerely appreciate the anonymous referees' insightful comments, which helped to strengthen the final text.

References

[1] Renardy M, Nohel J A and Lodge A S 1985 *Viscoelasticity and Rheology*. Academic Press
[2] Olmstead W E, Davis S H, Rosenblat S and Kath W L 1986 *Bifurcation with memory*. SIAM *J. Appl. Math.* 46(2): 171–188
[3] Sanz-Serna J M 1988 A numerical method for a partial integro-differential equation. SIAM *J. Numer. Anal.* 25(2): 319–327
[4] Chen C and Shih T 1998 *Finite element methods for integrodifferential equations*. World Scientific
[5] Wang Z, Cen D and Mo Y 2021 Sharp error estimate of a compact L1-ADI scheme for the two-dimensional time-fractional integro-differential equation with singular kernels. *Appl. Numer. Math*. 159: 190–203
[6] Qiu W, Xu D and Guo J 2021 The Crank-Nicolson-type Sinc-Galerkin method for the fourth-order partial integro-differential equation with a weakly singular kernel. *Appl. Numer. Math*. 159: 239–258
[7] Babaei A, Banihashemi S and Cattani C 2021 An efficient numerical approach to solve a class of variable-order fractional integro-partial differential equations. *Numer. Methods Partial Differ. Equ.* 37(1): 674–689
[8] Fakhar-Izadi F 2020 Fully spectral-Galerkin method for the one-and two-dimensional fourth-order time-fractional partial integro-differential equations with a weakly singular kernel. Numer. *Methods Partial Differ. Equ.* 38(2): 160–176
[9] Zhang H, Han X and Yang X 2013 Quintic B-spline collocation method for fourth order partial integro-differential equations with a weakly singular kernel . *Appl. Math. Comput.* 219(12): 6565–6575
[10] Dehestani H, Ordokhani Y and Razzaghi M 2020 *Numerical solution of variable-order time fractional weakly singular partial integro-differential equations with error estimation*. Math. Model. Anal. 25(4): 680–701
[11] Hashemizadeh E, Ebadi M A and Noeiaghdam S 2020 *Matrix Method by Genocchi Polynomials for Solving Nonlinear Volterra Integral Equations with Weakly Singular Kernels*. sym. 12(12): 2105
[12] Biazar J and Sadri K 2019 *Solution of weakly singular fractional integro-differential equations by using a new operational approach*. J. Comput. Appl. Math. 352: 453–477
[13] Allahviranloo T 2020 *Fuzzy fractional differential operators and equations*. Studies in fuzziness and soft computing series. Springer Nature
[14] Vaidyanathan S, Azar A T and Radwan AG 2018 *Mathematical Techniques of Fractional Order Systems*. Elsevier Science
[15] Matouk A Z ed 2020 *Advanced Applications of Fractional Differential Operators to Science and Technology*. IGI Global
[16] Yang X J, Ju X and Gao F 2020 *General Fractional Derivatives with Applications in Viscoelasticity*. Academic Press
[17] Cen D, Wang Z and Mo Y 2021 Second order difference schemes for time-fractional KdV-Burgers' equation with initial singularity. *Appl. Math. Lett.* 112: 106829
[18] Cen D and Wang Z 2022 Time two-grid technique combined with temporal second order difference method for two-dimensional semilinear fractional sub-diffusion equations. *Appl. Math. Lett.* 129: 107919
[19] Ou C, Cen D, Vong S and Wang Z 2022 Mathematical analysis and numerical methods for Caputo-Hadamard fractional diffusion-wave equations. *Appl. Numer. Math.* 177: 34–57
[20] Abdi N, Aminikhah H and Refahi Sheikhani A H 2021 High-order rotated grid point iterative method for solving 2D time fractional telegraph equation and its convergence analysis. *Comput. Appl. Math.* 40: 1–26
[21] Abdi N, Aminikhah H and Refahi Sheikhani A H 2021 On rotated grid point iterative method for solving 2D fractional reaction-subdiffusion equation with Caputo-Fabrizio operator. *J. Differ. Equ. Appl.* 27: 1134–60
[22] Abdi N, Aminikhah H and Refahi Sheikhani A H 2022 High-order compact finite difference schemes for the time-fractional Black-Scholes model governing European options. *Chaos Solit. Fractals.* 162: 112423
[23] Taghipour M and Aminikhah H 2022 Application of Pell collocation method for solving the general form of time-fractional Burgers equations. *Math. Sci.* 1–19
[24] Taghipour M and Aminikhah H 2022 A fast collocation method for solving the weakly singular fractional integro-differential equation. *Comput. Appl. Math.* 41: 142
[25] Taghipour M and Aminikhah H 2022 A spectral collocation method based on fractional Pell functions for solving time-

fractional Black-Scholes option pricing model. *Chaos Solit. Fractals.* 163: 112571

[26] Lin J, Bai J, Reutskiy S and Lu J 2022 A novel RBF-based meshless method for solving time-fractional transport equations in 2D and 3D arbitrary domains. *Eng. Comput.* 24: 1–8

[27] Molaee T and Shahrezaee A 2022 Numerical solution of an inverse source problem for a time-fractional PDE via direct meshless local Petrov-Galerkin method. *Eng. Anal. Bound. Elem.* 138: 211–218

[28] Wang H, Xu D, Zhou J and Guo J 2021 Weak Galerkin finite element method for a class of time fractional generalized Burgers' equation. *Numer. Methods Partial Differ. Equ.* 37(1): 732—749

[29] Zheng Y and Zhao Z 2020 The time discontinuous space-time finite element method for fractional diffusion-wave equation. *Appl. Numer. Math.* 150: 105—116

[30] Xu D, Qiu W and Guo J 2020 A compact finite difference scheme for the fourth-order time-fractional integro-differential equation with a weakly singular kernel. *Numer. Methods Partial Differ. Equ.* 36(2): 439-458

[31] Podlubny I 1998 *Fractional differential equations*. Elsevier

[32] Tariq H and Akram G 2017 *Quintic spline technique for time fractional fourth-order partial differential equation*. Numer. Methods Partial Differ. Equ. 33(2): 445–466

[33] Engel G, Garikipati K, Hughes T J R, Larson M J, Mazzei L and Taylor R L 2002 *Continuous/discontinuous finite element approximations of fourth-order elliptic problems in structural and continuum mechanics with applications to thin beams and plates, and strain gradient elasticity*. Comput. Methods Appl. Mech. Eng. 191(34): 3669–3750

[34] Miller R K 1978 *An integrodifferential equation for rigid heat conductors with memory*. J. Math. Anal. Appl. 66(2): 313–332

[35] Bazgir H and Ghazanfari B 2019 *Spectral solution of fractional fourth order partial integro-differential equations*. Comput. methods differ. equ. 7(2): 289–301

[36] Baleanu D, Darzi R and Agheli B 2017 *New study of weakly singular kernel fractional fourth-order partial integro-differential equations based on the optimum q-homotopic analysis method*. J. Comput. Appl. Math. 320: 193–201

[37] Heydari M H and Avazzadeh Z 2021 *Orthonormal Bernstein polynomials for solving nonlinear variable-order time fractional fourth-order diffusion-wave equation with nonsingular fractional derivative*. Math. Methods Appl. Sci. 44(4): 3098–3110

[38] Abdelkawy M A, Babatin M M and Lopes A M 2020 *Highly accurate technique for solving distributed-order time-fractional-sub-diffusion equations of fourth order*. Comput. Appl. Math. 39(2): 1–22, 2020

[39] Yang X, Xu D and Zhang H *Quasi-wavelet based numerical method for fourth-order partial integro-differential equations with a weakly singular kernel*. Int. J. Comput. Math. 88(15): 3236–3254

[40] Roul P and Goura V P 2020 *A high order numerical method and its convergence for time-fractional fourth order partial differential equations*. Appl. Math. Comput. 366: 124727

[41] Zhu C G and Kang W S 2010 *Numerical solution of Burgers-Fisher equation by cubic B-spline quasi-interpolation*. Appl. Math. Comput. 216(9): 2679–2686

[42] Zhu C G and Wang R H 2009 *Numerical solution of Burgers' equation by cubic B-spline quasi-interpolation*. Appl. Math. Comput. 208(1): 260–272

[43] Zhang J, Zheng J and Gao Q 2018 *Numerical solution of the Degasperis-Procesi equation by the cubic B-spline quasi-interpolation method*. Appl. Math. Comput. 324: 218–227

[44] Taghipour M and Aminikhah H 2021 A B-Spline Quasi Interpolation Crank-Nicolson Scheme for Solving the Coupled Burgers Equations with the Caputo-Fabrizio Derivative. *Math. Probl. Eng.*

[45] Sun Z 2009 *The Method of Order Reduction and Its Application to the Numerical Solutions of Partial Differential Equations* . Science Press, Beijing

[46] Mittal R C, Kumar S and Jiwari R 2020 A cubic B–spline quasi–interpolation method for solving two–dimensional unsteady advection diffusion equations. *Int. J. Numer. Methods Heat Fluid Flow.* 30(9): 4281—4306

[47] Mittal R C, Kumar S and Jiwari R 2021 A Comparative Study of Cubic B–spline–Based Quasi–interpolation and Differential Quadrature Methods for Solving Fourth-Order Parabolic PDEs. *Proc. Natl. Acad. Sci. India - Phys. Sci.* 91(3): 461-474

[48] Sablonniere P 2005 Univariate spline quasi-interpolants and applications to numerical analysis. *Rend. Semin. Mat. Univ*. 63(3): 211—222

[49] Li C and Cai M 2019 *Theory and numerical approximations of fractional integrals and derivatives*. Siam, Society for Industrial and Applied Mathematics

[50] Yang X J 2019 *General Fractional Derivatives: Theory, Methods and Applications*. Chapman and Hall/CRC

[51] Yang X J 2019 *General Fractional Derivatives:Theory,Methods and Applications*. Chapman and Hall/CRC

[52] Lin Y M and Xu C J 2007 Finite difference/spectral approximations for the time-fractional diffusion equation. *J. Comput. Phys.* 225: 1533—1552

[53] Sun L Y and Zhu C G 2020 Cubic B-spline quasi-interpolation and an application to numerical solution of generalized Burgers-Huxley equation. *Adv. Mech. Eng.* 12(11): 1687814020971061

[54] Kunoth A, Lyche T, Sangalli G and Serra-Capizzano S 2018 *Splines and PDEs: From approximation theory to numerical linear algebra*. Cetraro: Springer

[55] Schumaker L 1981 *Spline functions: basic theory* . Wiley Interscience

[56] Hollig K and Horner J 2013 *Approximation and modeling with B-splines*. SIAM

[57] Chen C, Liu F and Burrage K 2008 Finite difference methods and a Fourier analysis for the fractional reaction–subdiffusion equation. *Appl. Math. Comput.* 198: 754—769

[58] Bernatz R 2010 *Fourier series and numerical methods for partial differential equations*. New York: Wiley

[59] Holte J M 2009 *Discrete Gronwall lemma and applications*, MAA-NCS Meeting at the University of North Dakota