Ain Shams Engineering Journal 17 (2026) 103868

Contents lists available at ScienceDirect

Ain Shams Engineering Journal

journal homepage: www.sciencedirect.com

Full length article

Solving time fractional diffusion-wave equation using hyperbolic polynomial B-splines: A uniform grid approach

Aleeza Kiran[a], Muhammad Yaseen[a], Aziz Khan[b], Thabet Abdeljawad[b,c,d,e,*], Manar A. Alqudah[f], Rajermani Thinakaran[g]

[a] Department of Mathematics, University of Sargodha, Sargodha, 40100, Pakistan
[b] Department of Mathematics and Sciences, Prince Sultan University, P.O. Box 66833, 11586 Riyadh, Saudi Arabia
[c] Department of Fundamental Sciences, Faculty of Engineering and Architecture, Istanbul Gelisim University, Avcılar, Istanbul, 34310, Turkiye
[d] Department of Mathematics and Applied Mathematics, Sefako Makgatho Health Sciences University, Garankuwa, Medusa 0204, South Africa
[e] Department of Medical Research, China Medical University, Taichung 40402, Taiwan
[f] Department of Mathematical Sciences, Faculty of Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[g] Faculty of Data Science and Information Technology, INTI International University, Nilai, Negeri Sembilan, Malaysia

ARTICLE INFO

Keywords:
Uniform hyperbolic polynomial B-splines
Time fractional diffusion wave equation
Caputo derivative
Stability
Convergence

ABSTRACT

In this study, we present an efficient numerical scheme based on uniform hyperbolic polynomial B-splines for solving the time-fractional diffusion-wave equation involving the Caputo derivative. This equation models various physical phenomena including anomalous diffusion and complex dynamical behavior. The proposed method ensures a smooth and continuous approximation that effectively captures both local and global features of the solution such as sharp gradients and long-range memory effects. The key advantage of uniform hyperbolic polynomial B-splines lies in their flexibility and high accuracy across the computational domain. Stability and convergence analyses are carried out to confirm the method's robustness and error control. Finally, numerical results are compared with those reported in existing literature to demonstrate the accuracy and reliability of the scheme as process innovation.

1. Introduction

For $T > 0$, we consider the following time-fractional diffusion-wave equation with a reaction term [22]:

$$\frac{\partial^\beta \upsilon(z,t)}{\partial t^\beta} + \gamma \upsilon(z,t) = \frac{\partial^2}{\partial z^2}\upsilon(z,t) + f(z,t), \qquad \beta \in (1,2], \tag{1.1}$$

where:

- $z \in [a,b], t \in [0,T]$
- $\gamma > 0$ is the reaction coefficient.
- $f(z,t)$ is a source term.

The initial conditions (ICs) are:

$$\upsilon(z,0) = \phi_1(z) \qquad \upsilon_t(z,0) = \phi_2(z), \qquad z \in [a,b] \tag{1.2}$$

and the boundary conditions (BCs) are:

$$\upsilon(a,t) = \chi_1(t) \qquad \upsilon(b,t) = \chi_2(t), t \in [0,T], \tag{1.3}$$

where $\phi_1(z)$, $\phi_2(z)$, $\chi_1(t)$ and $\chi_2(t)$ are given functions. The fractional derivative $\frac{\partial^\beta \upsilon(z,t)}{\partial t^\beta}$ is defined in the Caputo sense as:

$$\frac{\partial^\beta}{\partial t^\beta}\upsilon(z,t) = \begin{cases} \dfrac{1}{\Gamma(2-\beta)}\displaystyle\int_0^t \frac{\partial^2 \upsilon(z,s)}{\partial s^2}(t-s)^{1-\beta}\,ds, & 1 < \beta < 2, \\ \dfrac{\partial^2 \upsilon(z,s)}{\partial s^2}, & \beta = 2. \end{cases}$$

The Caputo fractional derivative is widely used in modeling anomalous diffusion and wave propagation because it incorporates memory effects while allowing for physically meaningful initial conditions. It reduces to the classical second derivative when $\beta = 2$. It naturally incorporates initial conditions in terms of integer-order derivatives which is convenient for physical problems. It exhibits nonlocal behavior, capturing the long-memory effect characteristic of many diffusion and wave processes.

* Corresponding author at: Department of Mathematics and Sciences, Prince Sultan University, P.O. Box 66833, 11586 Riyadh, Saudi Arabia.
Email addresses: aleezarana45@gmail.com (A. Kiran), yaseen.yaqoob@uos.edu.pk (M. Yaseen), akhan@psu.edu.sa (A. Khan), tabdeljawad@psu.edu.sa (T. Abdeljawad), maalqudah@pnu.edu.sa (M.A. Alqudah), rajermani.thina@newinti.edu.my (R. Thinakaran).

https://doi.org/10.1016/j.asej.2025.103868
Received 3 June 2025; Received in revised form 7 October 2025; Accepted 4 November 2025
Available online 21 November 2025

This work introduces a novel numerical approach using uniform hyperbolic polynomial B-splines to solve the time-fractional diffusion-wave equation. Such equations model anomalous diffusion, wave propagation and complex physical phenomena, making efficient numerical methods crucial for real-world applications.

In recent years, numerous techniques have been proposed to address fractional PDEs. For instance, Arqub et al. [1] developed a reproducing kernel method for time-fractional PDEs in fluid dynamics. Sadri et al. [2] developed a numerical scheme for solving two-dimensional time-fractional diffusion-wave equation using a hybrid approach that combines classical and shifted Jacobi polynomials. Sadri and Aminikhah [3] proposed an efficient numerical algorithm for multi-term variable order time-fractional diffusion-wave equations using fifth-kind Chebyshev polynomials. Zhou and Xu [4] employed a Chebyshev wavelets collocation technique for time-fractional diffusion-wave equation. Zeng [5] devised a second-order stable finite difference scheme for such problems. Khader and Adel [6] developed an innovative finite difference method using Hermite interpolation for solving fractional wave equation. Jena and [7] proposed a fourth-order improvised cubic B-spline collocation scheme integrated with the Crank-Nicolson method for nonlinear Klein-Gordon and Sine-Gordon equations which govern soliton propagation. Ali et al. [8] presented a compact implicit difference scheme for time-fractional diffusion-wave equation. Salehi [9] created a meshless point collocation method for two-dimensional multi-term time-fractional diffusion-wave equations. Ali et al. [10] proposed a fourth-order difference approximation for the time-fractional modified sub diffusion equations.

Fu et al. [11] established a semi-analytical collocation Trefftz scheme for multi-term time-fractional diffusion-wave equations. Jena and Senapati [12] applied explicit and implicit spline collocation methods with the Thomas algorithm for efficient solutions of one-dimensional heat equation. Jena and Senapati [13] developed a hybrid numerical approach combining improvised cubic B-spline collocation, finite element method and the Crank-Nicolson technique to solve one-dimensional heat and advection-diffusion equations. Arqub et al. [14] combined cubic B-spline interpolation with finite differences to solve non-linear time-fractional diffusion-wave equation with reaction and damping terms. Avazzadeh et al.[15] developed a numerical method based on radial basis functions and finite difference method to obtain numerical solutions of the diffusion-wave equation. Heydari et al. [16] explored the use of wavelet-based methods for time-fractional diffusion-wave equations. Ebadian et al.[17] developed a novel approach using triangular basis functions for nonlinear fractional diffusion-wave equations. Nikan et al. [18] analyzed time-fractional fourth-order reaction-diffusion model for composite environments. Rashidinia and Mohmedi [19] investigated approximate solutions for multi-term time-fractional diffusion and diffusion-wave equations. Dehghan et al. [20] proposed two high-order numerical algorithms for multi-term time-fractional diffusion-wave equations. Wei et al. [21] established space-accurate finite element schemes for 2D multi-term time-fractional diffusion-wave equations. Yaseen et al. [22] proposed a numerical approach that approximates the time-fractional diffusion-wave equation by employing cubic trigonometric B-splines. specialized finite element methods for stochastic fractional differential equations. Wang [23] introduced a compact finite difference method for time fractional convection-diffusion-wave equations with variable coefficients. Shah et al. [24] investigated the time-fractional diffusion equation with a heat absorption term in spherical domains. Krishnaveni et al. [25] proposed an efficient fractional polynomial method for space-fractional diffusion equations. Singh et al. [26] developed a computational approach to fractional convection–diffusion equations using integral transforms. Farman et al. [27] constructed a fractional multi-scale soft-pattern modeling framework to explore ABPV infection dynamics within honeybee colonies. They illustrated that fractional-order formulations yield greater accuracy in modeling biological memory effects and nonlinear infection patterns, providing richer insight into colony production dynamics.

This work presents a novel numerical approach that combines uniform hyperbolic polynomial B-splines for spatial discretization with finite difference approximation of Caputo fractional derivative. Our proposed scheme undergoes rigorous stability and convergence analyses, establishing its theoretical foundations. To the best of our knowledge, this work presents the first application of hyperbolic B-splines to solve this class of fractional differential equation, representing a significant methodological advancement. The proposed method aims to provide:

- high accuracy on a uniform grid without requiring mesh adaptation,
- smooth and continuous approximations that effectively capture memory effects
- unconditional stability and proven convergence,
- computational efficiency compared with existing spline based and finite difference methods.

Although the present work is aimed at a one-dimensional formulation to develop the central methodology and fundamental theories, the presented method can be generalized to multi-dimensional situations. The spatial discretization by uniform hyperbolic polynomial B-splines can be extended through tensor products for two- or three-dimensional space domains. The time-discretization scheme for the Caputo derivative would remain notionally the same, although the resulting linear systems would be larger and more involved. We intend to investigate these multi-dimensional generalizations in our subsequent work. While the proposed UHPBS collocation on uniform grids offers high accuracy with banded linear systems and straightforward assembly, we note that meshless approaches such as the element-free Galerkin (EFG) methods [28,29] are attractive in complex two- and three-dimensional geometries where mesh generation is burdensome. EFG alleviates meshing by employing moving least squares (MLS) shape functions and typically uses background cells for numerical integration. We will explore the application of EFG for multi-dimensional fractional PDEs in a future study.The paper is structured as follows: Section 2 details the proposed numerical scheme. Section 3 provides a comprehensive stability analysis. Section 4 establishes convergence properties. Section 5 presents and discusses numerical results. Section 6 concludes with key findings and future research directions.

2. Derivation of the scheme

We present an efficient numerical scheme for solving (1.1) using hyperbolic B spline for spatial discretization and finite differences for temporal fractional derivative.

Let $\tau = \frac{T}{N}$ and $h = \frac{b-a}{M}$ be the temporal and spatial step sizes respectively, where $M, N \in \mathbb{Z}^+$. Define:

- temporal nodes: $t_n = n\tau$ for $n =, 0, 1, \ldots, N$.
- spatial nodes: $z_j = a + jh$ for $j = 0, 1, \ldots, M$.

B-spline functions are widely used for constructing smooth and flexible approximations due to their local support, partition of unity and differentiability properties [32]. These characteristics make them suitable for spatial discretization of fractional differential equations. The approximate solution $V(z,t)$ is constructed using uniform hyperbolic polynomial B-splines (UHPBS):

$$V(z,t) = \sum_{j=-1}^{M-1} C_j(t) HB_j(z), \tag{2.1}$$

where, $C_j(t)$ are time-dependent coefficients and $HB_j(z)$ are UHPBS basis functions with local support $[z_j, z_{j+1}]$. The explicit form of $HB_j(z)$ is:

$$HB_j(z) = \frac{1}{p1}\begin{cases} m(z_j) - (z - z_j), & z \in [z_j, z_{j+1}] \\ -2m(z_{j+1}) - m(z_{j+2}) + n(z_{j+1}) - h, & z \in [z_{j+1}, z_{j+2}] \\ m(z_{j+2}) + 2m(z_{j+3}) - n(z_{j+2}) + 2h\cos(h), & z \in [z_{j+2}, z_{j+3}] \\ -m(z_{j+4}) + (z - z_{j+4}), & z \in [z_{j+3}, z_{j+4}], \end{cases} \tag{2.2}$$

where $m(z_j) = \sinh(z - z_j), n(z_{j+1}) = (2\cos(h) + 1)(z - z_{j+1})$ and $p1 = 2h(\cos(h) - 1)$.

At the grid point (z_j, t_n), the approximation involves three neighbouring functions $HB_{j-1}(z)$, $HB_j(z)$ and $HB_{j+1}(z)$ so that we have:

$$v(z_j, t_n) = v_j^n = \sum_{w=j-1}^{j+1} C_w^n(t) HB_w(z). \tag{2.3}$$

Consequently, v_j^n and its derivatives are:

$$\begin{cases} v_j^n = \zeta_1 C_{j-1}^n + \zeta_2 C_j^n + \zeta_1 C_{j+1}^n, \\ (v_j^n)_z = -\zeta_3 C_{j-1}^n + \zeta_4 C_{j+1}^n + \zeta_3 C_{j+1}^n, \\ (v_j^n)_{zz} = \zeta_5 C_{j-1}^n + \zeta_6 C_j^n + \zeta_5 C_{j+1}^n, \end{cases} \tag{2.4}$$

where the coefficients ζ_k are defined in (2.5).

$$\begin{cases} \zeta_1 = \dfrac{\sinh(h) - h}{p_1}, \\ \zeta_2 = \dfrac{-2(\sinh(h) - h\cosh(h))}{p_1}, \\ \zeta_3 = \dfrac{1}{2h}, \\ \zeta_4 = 0, \\ \zeta_5 = \dfrac{\sinh(h)}{p_1}, \\ \zeta_6 = \dfrac{-2\sinh(h)}{p_1}. \end{cases} \tag{2.5}$$

An efficient approximate Caputo fractional derivative is obtained as:

$$\begin{aligned} \frac{\partial^\beta v(z, t_{n+1})}{\partial t^\beta} &= \frac{1}{\Gamma(2-\beta)} \int_0^{t_{n+1}} \frac{\partial^2 v(z,s)}{\partial s^2} \frac{ds}{(t_{n+1} - s)^{1-\beta}} \\ &= \frac{1}{\Gamma(2-\beta)} \sum_{j=0}^{n} \int_{t_j}^{t_{j+1}} \frac{\partial^2 v(z,s)}{\partial s^2} \frac{ds}{(t_{n+1} - s)^{1-\beta}} \\ &= \frac{1}{\Gamma(2-\beta)} \sum_{j=0}^{n} \frac{v(z,t_{j+1}) - 2v(z,t_j) + v(z,t_{j-1})}{(\tau^2)} \int_{t_j}^{t_{j+1}} \frac{ds}{(t_{n+1} - s)^{1-\beta}} + \iota_\tau^{n+1} \\ &= \frac{1}{\Gamma(2-\beta)} \sum_{j=0}^{n} \frac{v(z,t_{j+1}) - 2v(z,t_j) + v(z,t_{j-1})}{(\tau^2)} \int_{t_{n+1-j}}^{t_{n-j}} \frac{dr}{r^{1-\beta}} + \iota_\tau^{n+1} \\ &= \frac{1}{\Gamma(2-\beta)} \sum_{j=0}^{n} \frac{v(z,t_{j+1}) - 2v(z,t_n) + v(z,t_{n-1})}{(\tau^2)} \int_{t_j}^{t_{j+1}} \frac{dr}{r^{1-\beta}} + \iota_\tau^{n+1} \\ &= \frac{1}{\Gamma(3-\beta)} \sum_{j=0}^{n} \frac{v(z,t_{n+1-j}) - 2v(z,t_{n-j}) + v(z,t_{n-1-j})}{(\tau^\beta)} \\ &\quad ((j+1)^{2-\beta} - j^{2-\beta}) + \iota_\tau^{n+1} \\ &= \frac{1}{\Gamma(3-\beta)} \sum_{j=0}^{n} g_j \frac{v(z,t_{n+1-j}) - 2v(z,t_{n-j}) + v(z,t_{n-1-j})}{(\tau^\beta)} + \iota_\tau^{n+1} \end{aligned} \tag{2.6}$$

Where ι_τ^{n+1} is the approximation error and

$$g_j = (j+1)^{2-\beta} - j^{2-\beta}.$$

The coefficients g_j satisfy the following properties:

- $g_j > 0$, $j = 0, 1, 2, \ldots, n$,
- $1 = g_0 > g_1 > g_2 > \cdots > g_n$ and $g_n \to 0$ as $n \to \infty$,
- $\sum_{j=0}^{n}(g_j - g_{j+1}) = (1 - g_1) + \sum_{j=1}^{n-1}(g_j - g_{j+1}) + g_n = 1$.

Substituting (2.6) into (1.1), we obtain:

$$\begin{aligned} &\gamma_0\left(v^{n+1} - 2v^n + v^{n-1}\right) + \gamma_0 \sum_{j=1}^{n} g_j \left(v^{n+1-j} - 2v^{n-j} + v^{n-1-j}\right) + \gamma v^{n+1} \\ &\quad = \frac{\partial^2 v^{n+1}}{\partial z^2} + f(z, t_{n+1}), \end{aligned} \tag{2.7}$$

where $\gamma_0 = \frac{1}{\tau^\beta \Gamma(3-\beta)}$ and $v^{n+1} = v(z, t^{n+1})$.

In the case $n = 0$ or $j = n$, the term v^{-1} appears. We eliminate this using the initial condition:

$$v_t^0 = \frac{v^1 - v^{-1}}{2\tau} \to v^{-1} = v^1 - 2\tau\phi_2(z).$$

Finally substituting the HPBS approximation (2.4) into (2.7) yields the fully discrete scheme:

$$\begin{aligned} &((\gamma_0 + \gamma)\zeta_1 - \zeta_4)C_{j-1}^{n+1} + ((\gamma_0 + \gamma)\zeta_2 - \zeta_5)C_j^{n+1} + ((\gamma_0 + \gamma)\zeta_1 - \zeta_4)C_{j+1}^{n+1} \\ &\quad = \Bigg\{ 2\gamma_0\left(\zeta_1 C_{j-1}^n + \zeta_2 C_j^n + \zeta_1 C_{j+1}^n\right) - \gamma_0\left(\zeta_1 C_{j-1}^{n-1} + \zeta_2 C_j^{n-1} + \zeta_1 C_{j+1}^{n-1}\right) \\ &\qquad - \sum_{\theta=1}^{n} g_\theta \Big[\zeta_1\left(C_{j-1}^{n+1-\theta} - 2C_{j-1}^{n-\theta} + C_{j-1}^{n-1-\theta}\right) + \zeta_2\left(C_j^{n+1-\theta} - 2C_j^{n-\theta} + C_j^{n-1-\theta}\right) \\ &\qquad + \zeta_1\left(C_{j+1}^{n+1-\theta} - 2C_{j+1}^{n-\theta} + C_{j+1}^{n-1-\theta}\right)\Big] + \gamma_0 f(z_j, t^{n+1}) \Bigg\} \end{aligned} \tag{2.8}$$

This consists of $(M+1)$ equations with $(M+3)$ unknowns. To ensure a unique solution, we impose two additional linear equations from the boundary conditions as:

$$\begin{cases} \zeta_1 C_{-1}^{n+1} + \zeta_2 C_0^{n+1} + \zeta_1 C_1^{n+1} = 0, \\ \zeta_1 C_{M-1}^{n+1} + \zeta_2 C_M^{n+1} + \zeta_1 C_{M+1}^{n+1} = 0. \end{cases} \tag{2.9}$$

This results in a $(M+3) \times (M+3)$ system with a unique solution.

Remark 1. The explicit computation of the convolution sum in Eq. (2.8) results in a cost that increases as $O(N^2)$ with the number of steps in time N. For large-time simulations, this may be too expensive. We observe that fast algorithms, like one discussed in [30], can be used to bring this complexity close to $O(N)$ by taking advantage of the decay of the convolution coefficients and efficient summation methods. The use of such a quick algorithm will be taken under consideration for an upcoming, performance-focussed iteration of this technique.

2.1. *Initial state*

To initiate the iteration, we first determine the initial vector C^0, defined as:

$$C^0 = \left[C_{-1}^0, \quad C_0^0, \quad \ldots \quad, C_M^0, \quad C_{M+1}^0\right]^T.$$

The initial conditions for the problem are incorporated via the the following equations:

$$\begin{cases} v_0' = \phi_1'(z_0), \\ v_j^0 = \phi_1(z_j), \qquad j = 0, 1, 2, 3, \ldots, M, \\ v_M' = \phi_1'(z_M), \end{cases}$$

were ϕ represents the initial function. These equations specify the initial values and spatial derivatives of the approximate solution at the initial time level ($n = 0$). This results in system of $(M+3) \times (M+3)$ linear equations which can be expressed concisely in matrix form as:

$$A_3 C^0 = B_2,$$

where the matrices A_3, C^0 and B_2 are defined as:

$$A_3 = \begin{bmatrix} -\zeta_3 & \zeta_4 & \zeta_3 & 0 & \ldots & 0 \\ \zeta_1 & \zeta_2 & \zeta_1 & 0 & \ldots & 0 \\ 0 & \zeta_1 & \zeta_2 & \zeta_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & \zeta_1 & \zeta_2 & \zeta_1 \\ 0 & \ldots & 0 & -\zeta_3 & \zeta_4 & \zeta_3 \end{bmatrix},$$

$$C^0 = \left[C_{-1}^0, \quad C_0^0, \quad \ldots, \quad C_M^0, \quad C_{M+1}^0\right]^T,$$

$$B_2 = \left[\phi_1'(z_0), \quad \phi_1(z_0) \quad, \ldots, \quad \phi_1(z_M), \quad \phi_1'(z_M)\right]^T.$$

3. Stability analysis

This section analyzes the stability of the proposed numerical scheme for the homogeneous case (i.e., $f = 0$.) We introduce a Fourier mode perturbation δ_j^n and its approximation $\tilde{\delta}_j^n$, defining the error vector F_j^n as:

$$F_j^n = \delta_j^n - \tilde{\delta}_j^n. \tag{3.1}$$

Substituting into the discretized scheme (2.8) yields the error equation:

$$\begin{aligned}
&((\gamma_0+\gamma)\zeta_1-\zeta_4)F_{j-1}^{n+1}+((\gamma_0+\gamma)\zeta_2-\zeta_5)F_j^{n+1}+((\gamma_0+\gamma)\zeta_1-\zeta_4)F_{j+1}^{n+1}\\
&\quad=2\gamma_0\left(\zeta_1F_{j-1}^n+\zeta_2F_j^n+\zeta_1F_{j+1}^n\right)-\gamma_0\left(\zeta_1F_{j-1}^{n-1}+\zeta_2F_j^{n-1}+\zeta_1F_{j+1}^{n-1}\right)\\
&\qquad-\sum_{\theta=1}^{n}g_\theta\Big[\zeta_1\left(F_{j-1}^{n+1-\theta}-2F_{j-1}^{n-\theta}+F_{j-1}^{n-1-\theta}\right)\\
&\qquad\qquad+\zeta_2\left(F_j^{n+1-\theta}-2F_j^{n-\theta}+F_j^{n-1-\theta}\right)\\
&\qquad\qquad+\zeta_1\left(F_{j+1}^{n+1-\theta}-2F_{j+1}^{n-\theta}+F_{j+1}^{n-1-\theta}\right)\Big].
\end{aligned}\tag{3.2}$$

The error satisfies the boundary conditions:

$$F_0^\theta = \chi_1(t_\theta), \qquad F_M^\theta = \chi_2(t_\theta), \quad \theta = 0,1,\dots,N \tag{3.3}$$

and initial conditions:

$$F_j^0 = \phi_1(z_j), \qquad (F_t)_j^0 = \phi_2(z_j), \quad j = 1,2,\dots,M \tag{3.4}$$

Let the grid function be:

$$F^\theta(z)=\begin{cases}F_j^\theta, & \text{if } z_j-\dfrac{h}{2}<z\le z_j+\dfrac{h}{2}, \quad j=1,\dots,M-1,\\ 0, & \text{if } a<z\le\dfrac{h}{2} \quad\text{or}\quad b-\dfrac{h}{2}<z\le b.\end{cases}$$

Expressing $F^y(z)$ as a Fourier series, we have:

$$F^\theta(z)=\sum_{m=-\infty}^{\infty}a_\theta(m)\exp\left(\frac{I2\pi mz}{b-a}\right), \quad \theta=0,1,\dots,N$$

where

$$a_\theta(m)=\frac{1}{b-a}\int_a^b F^\theta(z)\exp\left(\frac{-I2\pi mz}{b-a}\right)dz.$$

Let

$$F^\theta=\left[F_1^\theta,F_2^\theta,\dots,F_{M-1}^\theta\right]^T.$$

Denoting the norm as:

$$\|F^\theta\|_2=\left(\sum_{j=1}^{M-1}h|F_j|^2\right)^{\frac12}=\left[\int_a^b|F^\theta(z)|^2dz\right]^{\frac12},$$

and using the Parseval's equality, we get:

$$\int_a^b|F^\theta(z)|^2dz=\sum_{m=-\infty}^{\infty}|a_\theta(m)|^2,$$

leading to:

$$||F^\theta(z)||_2^2=\sum_{m=-\infty}^{\infty}|a_\theta(m)|^2.$$

Based on this analysis using Fourier series, we assume that the solution of Eqs (3.2)–(3.4) follows:

$$F_j^n=\sigma_n e^{I\alpha jh},$$

where $I=\sqrt{-1}$ and α is real. Substituting this into (3.2), we obtain

$$\begin{aligned}
&\left[(\gamma_0+\gamma)\zeta_1-\zeta_4\right]\sigma_{n+1}e^{i\alpha(j-1)h}+\left[(\gamma_0+\gamma)\zeta_2-\zeta_5\right]\sigma_{n+1}e^{i\alpha jh}\\
&\quad+\left[(\gamma_0+\gamma)\zeta_1-\zeta_4\right]\sigma_{n+1}e^{i\alpha(j+1)h}\\
&=2\gamma_0\left[\zeta_1\sigma_n e^{i\alpha(j-1)h}+\zeta_2\sigma_n e^{i\alpha jh}+\zeta_1\sigma_n e^{i\alpha(j+1)h}\right]\\
&\quad-\gamma_0\left[\zeta_1\sigma_{n-1}e^{i\alpha(j-1)h}+\zeta_2\sigma_{n-1}e^{i\alpha jh}+\zeta_1\sigma_{n-1}e^{i\alpha(j+1)h}\right]\\
&\quad-\sum_{\theta=1}^{n}g_\theta\Big\{\zeta_1\left[\sigma_{n+1-\theta}e^{i\alpha(j-1)h}-2\sigma_{n-\theta}e^{i\alpha(j-1)h}+\sigma_{n-1-\theta}e^{i\alpha(j-1)h}\right]\\
&\qquad+\zeta_2\left[\sigma_{n+1-\theta}e^{i\alpha jh}-2\sigma_{n-\theta}e^{i\alpha jh}+\sigma_{n-1-\theta}e^{i\alpha jh}\right]\\
&\qquad+\zeta_1\left[\sigma_{n+1-\theta}e^{i\alpha(j+1)h}-2\sigma_{n-\theta}e^{i\alpha(j+1)h}+\sigma_{n-1-\theta}e^{i\alpha(j+1)h}\right]\Big\}.
\end{aligned}\tag{3.5}$$

Dividing the above equation by $e^{I\alpha jh}$ and using the relation $e^{-I\alpha h}+e^{I\alpha h}=2\cos(\alpha h)$, we get:

$$\begin{aligned}
&((\gamma_0+\gamma)\zeta_1-\zeta_4)\sigma_{n+1}[2\cos(\alpha h)]+((\gamma_0+\gamma)\zeta_2-\zeta_5)\sigma_{n+1}\\
&\quad=2\gamma_0\left(\zeta_1\sigma_n[2\cos(\alpha h)]+\zeta_2\sigma_n\right)-\gamma_0\left(\zeta_1\sigma_{n-1}[2\cos(\alpha h)]+\zeta_2\sigma_{n-1}\right)\\
&\quad-\sum_{\theta=1}^{n}g_\theta\Big[\zeta_1\left(\sigma_{n+1-\theta}[2\cos(\alpha h)]-2\sigma_{n-\theta}[2\cos(\alpha h)]+\sigma_{n-1-\theta}[2\cos(\alpha h)]\right)\\
&\quad+\zeta_2\left(\sigma_{n+1-\theta}-2\sigma_{n-\theta}+\sigma_{n-1-\theta}\right)\Big].
\end{aligned}\tag{3.6}$$

This simplifies to

$$\begin{aligned}
&\left[1+\frac{2(\zeta_1\gamma-\zeta_4)\cos(\alpha h)+(\zeta_2\gamma-\zeta_5)}{\gamma_0\left[2\zeta_1\cos(\alpha h)+\zeta_5\right]}\right]\sigma_{n+1}\\
&\quad=2\sigma_n-\sigma_{n-1}-\sum_{\theta=1}^{n}g_\theta\left[\sigma_{n+1-\theta}-2\sigma_{n-\theta}+\sigma_{n-1-\theta}\right].
\end{aligned}$$

and finally:

$$\sigma_{n+1}=\frac{2}{\mu}\sigma_n-\frac{1}{\mu}\sigma_{n-1}-\frac{1}{\mu}\sum_{\theta=1}^{n}g_\theta\left(\sigma_{n+1-\theta}-2\sigma_{n-\theta}+\sigma_{n-1-\theta}\right), \tag{3.7}$$

where $\mu=\left[1+\frac{2(\zeta_1\gamma-\zeta_4)\cos(\alpha h)+(\zeta_2\gamma-\zeta_5)}{\gamma_0(2\zeta_1\cos(\alpha h)+\zeta_5)}\right]$ and $\mu\ge1$.

Proposition 1. *If σ_n is the solution of (3.7), then*

$$|\sigma_n|\le2|\sigma_0|, \quad n=0,1,\dots,T\times N.$$

Proof. We prove the result using mathematical induction.

Base Case ($n = 0$): From (3.7), we have that σ_0 satisfies the initial condition. Clearly,

$$|\sigma_0|\le2|\sigma_0|.$$

Induction Hypothesis: Assume that for some $n \ge 0$, the inequality $|\sigma_\theta|\le2|\sigma_0|$ holds for all $\theta=0,1,\dots,n$.

Inductive Step: We need to show that $|\sigma_{n+1}|\le2|\sigma_0|$. From (3.7):

$$|\sigma_{n+1}|\le\frac{2|\sigma_n|}{\mu}-\frac{|\sigma_{n-1}|}{\mu}-\frac{1}{\mu}\sum_{\theta=1}^{n}g_\theta\left(|\sigma_{n+1-\theta}|-2|\sigma_{n-\theta}|+|\sigma_{n-1-\theta}|\right).$$

Using the induction hypothesis $|\sigma_y|\le2|\sigma_0|$ for all y and $\mu\ge1$:

$$|\sigma_{n+1}|\le\frac{4|\sigma_0|}{\mu}-\frac{2|\sigma_0|}{\mu}-\frac{2}{\mu}\sum_{y=1}^{n}|g_\theta|(|\sigma_0|-2|\sigma_0|+|\sigma_0|).$$

Hence:

$$|\sigma_{n+1}|\le2|\sigma_0|.$$

Thus, by induction, the inequality holds for all $n=0,1,\dots,T\times N$. □

Theorem 1. *The method (2.8) is unconditionally stable.*

Proof. Using Proposition 1 and the condition $||F^{\theta}(z)||_2^2 = \sum_{m=-\infty}^{\infty} |a_\theta(m)|^2$, we have

$$||F^{\theta}||_2 \leq 2||F^0||_2, \quad \theta = 0, 1, \ldots, N,$$

which proves the unconditional stability of the scheme. □

Remark 2. Based on Duhamel's principle [31], the solution of an inhomogeneous linear problem can be written using the solution of the corresponding homogeneous problem. This means that if the numerical scheme is stable for the homogeneous case ($f = 0$ then the solution for the inhomogeneous case ($f \neq 0$ will also be stable — as long as the source term $f(z,t)$ is smooth and bounded.

Remark 3. The stability and convergence analyses employ different but related norms:

- **Stability Analysis (Section 3)** uses the *discrete L^2 norm*:

$$\|F^n\|_2 = \left(h \sum_{j=1}^{M-1} |F_j^n|^2 \right)^{1/2}$$

This norm is natural for the Fourier (von Neumann) stability analysis of the discrete scheme.

- **Convergence Analysis (Section 4)** uses the *continuous L^2 norm*:

$$\|e^n\| = \left(\int_a^b |e^n(x)|^2 dx \right)^{1/2}$$

This is the appropriate norm for measuring the error between the exact solution and our spline approximation in the function space $L^2([a,b])$.

4. Convergence analysis

This section focuses on the convergence analysis of our proposed scheme described in (2.8). Beginning with ($f = 0$) in (2.8), we convert it to homogeneous form and express the summation term as:

$$\sum_{j=0}^{n} \left(v^{n+1-j} - 2v^{n-j} + v^{n-1-j} \right) = v^{n+1} - v^n + g_n v^1 - g_n v^0$$
$$+ \sum_{j=0}^{n} (g_{j+1} - g_j) \left(v^{n-j} - v^{n-1-j} \right) - 2g_n \tau \phi_2(z). \tag{4.1}$$

Substituting (4.1) into the homogeneous form of (2.7)

$$\left[(\gamma_0 + \gamma) v^{n+1} - (v_{zz})^{n+1} \right] = \gamma_0 v^n - \gamma_0 g_n v^1 + \gamma_0 g_n v^0$$
$$- \gamma_0 \sum_{j=0}^{n-1} (g_{j+1} - g_j) \left(v^{n-j} - v^{n-1-j} \right) + 2g_n \tau \phi_2(z). \tag{4.2}$$

Theorem 2. *Let $v(z,t_n)_{n=0}^{N-1}$ be the analytical solution of (1.1) with initial and boundary conditions (1.2) and (1.3), and let ${v^n}_{n=0}^{N-1}$ be the numerical solution of (4.2). Then the error $e^{n+1} = v(z,t^{n+1}) - v^{n+1}$ satisfies:*

$$\|e^{n+1}\| \leq C\tau^{3-\lambda},$$

where C is a positive constant independent of τ.

Proof. The exact solution v satisfies the semi discrete scheme (4.2) upto the truncation error l_τ^{n+1}, where

$$l_j^{n+1} = C\,\tau^{3-\lambda}.$$

That is

$$(\gamma_0 + \gamma) v(z,t^{n+1}) - \left(v(z,t^{n+1}) \right)_{zz}$$
$$= \gamma_0 v(z,t^n) - \gamma_0 g_n v(z,t^1) + \gamma_0 g_n v(z,t^0)$$
$$- \gamma_0 \sum_{j=0}^{n-1} (g_{j+1} - g_j) \left(v(z,t^{n-j}) - v(z,t^{n-j-1}) \right) + l_\tau^{n+1} + 2g_n \tau \phi_2(z).$$

Subtracting (4.2) from (4.3) gives

$$\left[(\gamma_0 + \gamma) e^{n+1} - (e_{zz})^{n+1} \right] = \gamma_0 e^n - \gamma_0 g_n e^1 + \gamma_0 g_n e^0$$
$$- \gamma_0 \sum_{j=0}^{n-1} (g_{j+1} - g_j) \left(e^{n-j} - e^{n-j-1} \right) + l_\tau^{n+1}.$$

Noting that $e^0 = 0$ and taking the inner product with e^{n+1} yields:

$$(\gamma_0 + \gamma)\|e^{n+1}\|^2$$
$$= \left\langle (e^{n+1})_{zz}, e^{n+1} \right\rangle + \gamma_0 \left\langle e^n, e^{n+1} \right\rangle - \gamma_0 g_n \left\langle e^1, e^{n+1} \right\rangle + \gamma_0 g_n \left\langle e^0, e^{n+1} \right\rangle$$
$$- \gamma_0 \sum_{j=1}^{n} (g_{j+1} - g_j) \left[\left\langle e^{n-j}, e^{n+1} \right\rangle - \left\langle e^{n-1-j}, e^{n+1} \right\rangle \right] + \left\langle l_\tau^{n+1}, e^{n+1} \right\rangle$$
$$= -\|(e^{n+1})_z\|^2 + \gamma_0 \left\langle e^n, e^{n+1} \right\rangle - \gamma_0 g_n \left\langle e^1, e^{n+1} \right\rangle$$
$$- \gamma_0 \sum_{j=1}^{n} (g_{j+1} - g_j) \left[\left\langle e^{n-j}, e^{n+1} \right\rangle - \left\langle e^{n-1-j}, e^{n+1} \right\rangle \right] + \left\langle l_\tau^{n+1}, e^{n+1} \right\rangle$$
$$\leq \gamma_0 \left\langle e^n, e^{n+1} \right\rangle - \gamma_0 g_n \left\langle e^1, e^{n+1} \right\rangle$$
$$- \gamma_0 \sum_{j=0}^{n-1} (g_{j+1} - g_j) \left[\left\langle e^{n-j}, e^{n+1} \right\rangle - \left\langle e^{n-1-j}, e^{n+1} \right\rangle \right] + \left\langle l_\tau^{n+1}, e^{n+1} \right\rangle, \tag{4.3}$$

where we have used $\langle v_{zz}, v \rangle = -\langle v_z, v_z \rangle$ and the property - $\|(e^{n+1})_z\|^2 \leq 0$. Applying the Cauchy Shwarz inequality, $\langle x, y \rangle \leq \|x\|\|y\|$, we have

$$(\gamma_0 + \gamma)\|e^{n+1}\|^2 \leq \gamma_0 \|e^n\|\|e^{n+1}\| - \gamma_0 g_n \|e^1\|\|e^{n+1}\|$$
$$- \gamma_0 \sum_{j=0}^{n-1} (g_{j+1} - g_j) \left[\|e^{n-j}\|\|e^{n+1}\| - \|e^{n-j-1}\|\|e^{n+1}\| \right]$$
$$+ \|l_\tau^{n+1}\|\|e^{n+1}\|.$$

Dividing throughout by $\|e^{n+1}\|$ gives

$$(\gamma_0 + \gamma)\|e^{n+1}\| \leq \gamma_0 \|e^n\| - \gamma_0 g_n \|e^1\|$$
$$- \gamma_0 \sum_{j=0}^{n-1} (g_{j+1} - g_j) \left[\|e^{n-j}\| - \|e^{n-j-1}\| \right] + \|l_\tau^{n+1}\|. \tag{4.4}$$

We now prove (4.4) by mathematical induction. For the base case ($n = 0$):

$$(\gamma_0 + \gamma)\|e^1\| \leq \|l_j^1\|$$
$$\Rightarrow \quad \|e^1\| \leq \frac{1}{(\gamma_0 + \gamma)} \|l_\tau^1\|$$
$$\Rightarrow \quad \|e^1\| \leq C_0 \tau^{3-\lambda},$$

where C_0 is a constant independent of τ.

Induction step:

Assume that for $n = q - 1$, we have $\|e^q\| \leq C(\tau^2 + \tau h)$, where C is a constant independent of τ. For $n = q$, from (4.4) we have

$$(\gamma_0 + \gamma)\|e^{q+1}\| \leq \gamma_0 \|e^q\| - \gamma_0 g_q \|e^1\|$$
$$- \gamma_0 \sum_{j=0}^{q-1} (g_{j+1} - g_j) \left[\|e^{q-j}\| - \|e^{q-j-1}\| \right] + \|l_\tau^{q+1}\|.$$

Let

$$\|e^1\| \le C_1\tau^{3-\lambda}, \quad F = \max_{0\le i\le q}\|e^i\| \le C_2\tau^{3-\lambda}, \quad G = \max_{0\le j\le q}\left(\|e^{q-j}\| - \|e^{q-j+1}\|\right) \le C_3\tau^{3-\lambda},$$

where C_1, C_2, and C_3 are independent of τ. Then

$$(\gamma_0+\gamma)\|e^{q+1}\| \le \gamma_0 C_2\tau^{3-\lambda} - \gamma_0 g_q C_1\tau^{3-\lambda} + \gamma_0(1-g_n)C_3\tau^{3-\lambda} + C_v\tau^{3-\lambda}$$
$$\le \left[\gamma_0 C_2 - \gamma_0 g_q C_1 + \gamma_0(1-g_n)C_3 + C_v\right]\tau^{3-\lambda}.$$

Since the term in brackets is independent of τ, we conclude

$$\|e^{q+1}\| \le C\,\tau^{3-\lambda}.$$

This completes the proof. □

Remark 4. The stability and convergence analyses are carried out under the assumptions that the exact solution $v(z,t)$ and source term $f(z,t)$ are sufficiently smooth and bounded on the domain. The analysis is based on uniform grids and Fourier mode representation of the error, while the coefficients g_j from the Caputo derivative approximation are assumed to satisfy their standard positivity and monotonicity properties. Although the method is unconditionally stable in theory, in practice reduced accuracy may occur for very coarse meshes or in cases involving non-smooth data or non-uniform grids.

5. Numerical findings and discussions

This section validates and assesses the effectiveness of the proposed methodology by solving a range of test problems and evaluating the results using the L_2 and L_∞ error norms. The error measures are defined as follows:

$$\text{Absolute error:} \quad E_j = \left|v^{\text{ext}}(z_j) - V^{\text{app}}(z_j)\right|, \qquad z_j \in [a,b],$$

$$\text{Maximum error norm}(L^\infty): \quad L_\infty = \max_{0\le j\le M} E_j, \qquad z_j \in [a,b],$$

$$L_2\text{error norm:} \quad L_2 = \sqrt{h\sum_{j=0}^{M} E_j^2}, \qquad z_j \in [a,b],$$

where $v^{\text{ext}}(z_j)$ and $V^{\text{app}}(z_j)$ denote the exact and approximate solutions, respectively, at the spatial grid point z_j and fixed time t. The convergence rates for spatial and temporal refinements are given by:

$$p_h = \frac{\log\left(\frac{e(h_2)}{e(h_1)}\right)}{\log\left(\frac{h_2}{h_1}\right)}, \qquad h_2 > h_1,$$

$$p_\tau = \frac{\log\left(\frac{e(\tau_2)}{e(\tau_1)}\right)}{\log\left(\frac{\tau_2}{\tau_1}\right)}, \qquad \tau_2 > \tau_1.$$

All numerical calculations were performed on a personal computer equipped with an Intel Core i7 processor (2.80 GHz), 16 GB RAM and Windows 10 operating system. The codes were implemented using MATHEMATICA 13.2.

Example 1. Consider the fractional-order diffusion-wave equation for $\gamma = 0$:

$$\frac{\partial^\beta}{\partial t^\beta}v(z,t) = \frac{\partial^2}{\partial z^2}v(z,t) + f(z,t), \quad (z,t)\in[0,1]\times[0,T],$$

with initial conditions (ICs):

$$v(z,0) = 0, \quad v_t(z,0) = -\sin(\pi z),$$

and the boundary conditions (BCs):

$$v(0,t) = 0, \quad v(1,t) = 0.$$

The associated source term is given by:

$$f(z,t) = \frac{2t^{2-\beta}\sin(\pi z)}{\Gamma(3-\beta)} + \left(t^2 - t\right)\pi^2\sin(\pi z).$$

The exact solution for this problem is [6]:

$$v(z,t) = \sin(\pi z)(t^2 - t).$$

We apply the proposed numerical scheme to solve this problem. Tables 1 and 2 compare the maximum errors obtained using our method with those from the Hermite formula [14] for different values of β. Table 3 demonstrates that the proposed method achieves second-order spatial accuracy when the temporal discretization is sufficiently fine. Table 4 shows that the temporal convergence rate is consistent with the theoretical predications confirming the accuracy of the proposed time discretization scheme. Tables 5 and 6 present the error norms for various parameter values demonstrating the efficiency and accuracy of our method. Fig. 1 provides visual comparison between the approximate and exact solutions at different time steps. Fig. 2 gives comparison of the absolute errors for Example 1. Fig. 3 shows 3D error profile. Fig. 4 illustrates a 3D comparison between the approximate and exact solutions. For $M = 20$, $\tau = 0.01$ and $T = 0.5$ the approximate solution is:

Table 1
Maximum errors for Example 1 at various values of τ and h with $\beta = 1.5$ and $t = 0.2$.

h	$\frac{1}{5}$	$\frac{1}{10}$	$\frac{1}{20}$	$\frac{1}{30}$	$\frac{1}{30}$	$\frac{1}{40}$	$\frac{1}{40}$	$\frac{1}{45}$
τ	$\frac{1}{50}$	$\frac{1}{100}$	$\frac{1}{150}$	$\frac{1}{150}$	$\frac{1}{200}$	$\frac{1}{210}$	$\frac{1}{220}$	$\frac{1}{230}$
[6]	0.01149	0.00361	0.00120	0.00115	0.00021	0.00019	0.00006	0.00004
Present method	0.00151	0.00038	0.00009	0.00004	0.00004	0.00002	0.00002	0.00001

Table 2
Maximum errors for Example 1 at various values of τ and h with $\beta = 1.7$ and $t = 0.4$.

h	$\frac{1}{10}$	$\frac{1}{20}$	$\frac{1}{30}$	$\frac{1}{50}$	$\frac{1}{50}$	$\frac{1}{60}$	$\frac{1}{60}$	$\frac{1}{70}$
τ	$\frac{1}{50}$	$\frac{1}{100}$	$\frac{1}{200}$	$\frac{1}{250}$	$\frac{1}{300}$	$\frac{1}{400}$	$\frac{1}{450}$	$\frac{1}{480}$
[6]	0.01396	0.01064	0.00736	0.00653	0.00586	0.00494	0.00460	0.00443
Present method	0.00117	0.00029	0.00012	0.00004	0.00004	0.00003	0.00003	0.00002

$$v(s,0.5)=\begin{cases}1.89085\times10^{-11}-(7.77467)z-1.8912\times10^{-11}\cosh(z)+7.05793\sinh(z), & s\in\left[0,\frac{1}{20}\right)\\ -0.00957191-(7.58323)z+0.0095759\cosh(z)+6.86625\sinh(z), & s\in\left[\frac{1}{20},\frac{1}{10}\right)\\ -0.0473882-(7.20507)z+0.0474552\cosh(z)+6.4862\sinh(z), & s\in\left[\frac{1}{10},\frac{3}{20}\right)\\ \vdots & \\ \vdots & \\ -7.25246+(7.20507)z+7.69582\cosh(z)-10.0645\sinh(z), & s\in\left[\frac{17}{20},\frac{9}{10}\right)\\ -7.59281+(7.58323)z+8.08401\cosh(z)-10.6064\sinh(z), & s\in\left[\frac{9}{10},\frac{19}{20}\right)\\ -7.77467+(7.77467)z+8.29449\cosh(z)-10.891\sinh(z), & s\in\left[\frac{19}{20},1\right).\end{cases}$$

For $M=20$, $\tau=0.01$ and $T=1$ the approximate solution is:

$$v(s,1)=\begin{cases}7.9142\times10^{-13}+(0.56294)z-7.9183\times10^{-13}\cosh(z)-0.511043\sinh(z), & s\in\left[0,\frac{1}{20}\right)\\ 0.000693073+(0.549079)z-0.000693362\cosh(z)-0.497165\sinh(z), & s\in\left[\frac{1}{20},\frac{1}{10}\right)\\ 0.00343123+(0.521697)z-0.00343609\cosh(z)-0.469646\sinh(z), & s\in\left[\frac{1}{10},\frac{3}{20}\right)\\ \vdots & \\ \vdots & \\ 0.525129-(0.521697)z-0.557231\cosh(z)+0.72874\sinh(z), & s\in\left[\frac{17}{20},\frac{9}{10}\right)\\ 0.549772-(0.549079)z-0.585338\cosh(z)+0.76798\sinh(z), & s\in\left[\frac{9}{10},\frac{19}{20}\right)\\ 0.56294-(0.56294)z-0.600579\cosh(z)+0.788581\sinh(z), & s\in\left[\frac{19}{20},1\right).\end{cases}$$

Table 3
Spatial error norms and corresponding convergence rates p_h for Example 1, computed at various spatial resolutions with fixed time step $\tau=1/80$ and fractional order $\beta=1.5$.

M	L_2 Norm	L_∞ Norm	p_h
10	1.0317×10^{-3}	1.4590×10^{-3}	–
20	2.5798×10^{-4}	3.6484×10^{-4}	1.999675
40	6.4499×10^{-5}	9.1216×10^{-5}	1.999932
80	1.6125×10^{-5}	2.2804×10^{-5}	1.999983
160	4.0312×10^{-6}	5.7010×10^{-6}	1.999996

Table 4
Temporal error norms and corresponding convergence rates p_τ for Example 1, computed at various time step sizes with fixed spatial resolution $M=50$ and fractional order $\beta=1.5$.

τ	L_2 Norm	L_∞ Norm	p_τ
$\frac{1}{2}$	4.0676×10^{-1}	1.2863×10^{0}	–
$\frac{1}{4}$	1.4028×10^{-1}	4.4361×10^{-1}	1.5359
$\frac{1}{8}$	4.7822×10^{-2}	1.5123×10^{-1}	1.5526
$\frac{1}{16}$	1.1806×10^{-2}	3.7333×10^{-2}	2.0182
$\frac{1}{32}$	3.6906×10^{-3}	1.6708×10^{-2}	1.6775

Example 2. Consider the time-dependent fractional-order diffusion-wave equation:

$$\frac{\partial^\beta}{\partial t^\beta}v(z,t)+v(z,t)=\frac{\partial^2}{\partial z^2}v(z,t)+f(z,t),\quad (z,t)\in[0,1]\times[0,T]$$

subject to the initial conditions (ICs):

$$v(z,0)=0,\quad v_t(z,0)=0,$$

and boundary conditions (BCs):

$$v(0,t)=0,\quad v(1,t)=0.$$

and,

The source term is given by:

$$f(z,t)=\frac{2t^{2-\beta}z(1-z)}{\Gamma(3-\beta)}+t^2z(1-z)+2t^2.$$

The exact solution for this problem is [22] $v(z,t)=t^2z(1-z)$. We apply the proposed numerical scheme to solve Example 2. Table 7 presents a comparison of the errors generated by the scheme. Table 8 computes spatial convergence rates for $\beta=1.5$ when the temporal step size is sufficiently small. Table 9 demonstrates that the temporal convergence order

Table 5
Error norms for Example 1 at various spatial steps with $\tau = 1/120$ and $\beta = 1.5$.

N	L_2 *Norm*	L_∞ *Norm*
10	0.00026	0.00038
20	0.00006	0.00009
40	0.00001	0.00002
80	0.000004	0.000006
160	0.000001	0.000001

Table 6
Error norms for Example 1 at various time levels with $\tau = \frac{1}{120}$, $\beta = 1.5$ and $h = \frac{1}{60}$.

t	L_2 *Norm*	L_∞ *Norm*
0.2	0.000007	0.00001
0.4	0.00002	0.00004
1	0.00003	0.00004
2	0.00030	0.00043

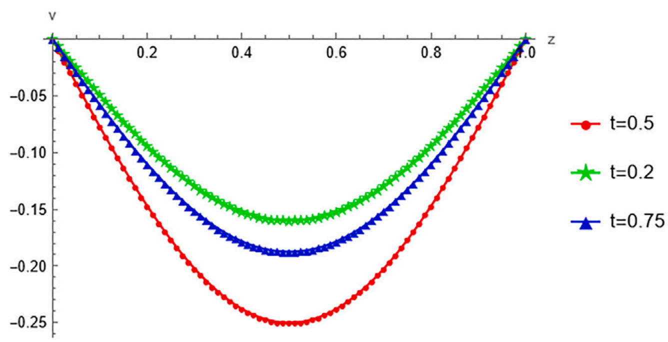


Fig. 1. Comparison of analytical (solid lines) and approximate solutions (circles, stars and triangles) for Example 1 at various time levels with $\beta = 1.5$, $\tau = 0.01$, $\gamma = 0$ and $h = 0.0125$.

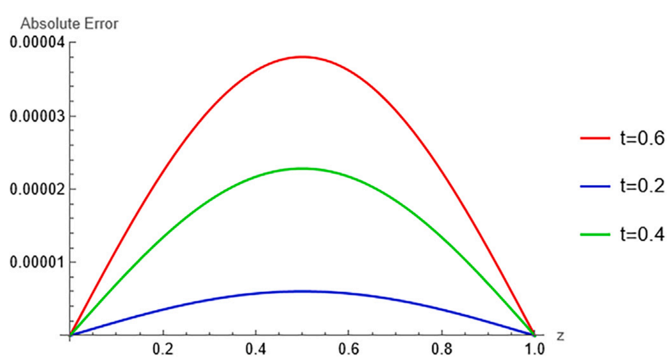


Fig. 2. Absolute error for Example 1 at various time levels with $\beta = 1.5$, $\tau = 0.01$, $\gamma = 0$ and $h = 0.0125$.

is approximately second order with a slight increase for finer time steps, indicating the efficiency of the time discretization scheme. Fig. 5 provides a visual comparison between the approximate and exact solutions at different time levels. Fig. 6 displays the absolute errors for Example 2. Fig. 7 depicts the 3D error profile and Fig. 8 illustrates a 3D comparison between approximate and exact solutions. The approximate solution obtained using the present scheme for Example 2 with $T = 0.5$, $\tau = 0.01$ and $M = 20$ is given by:

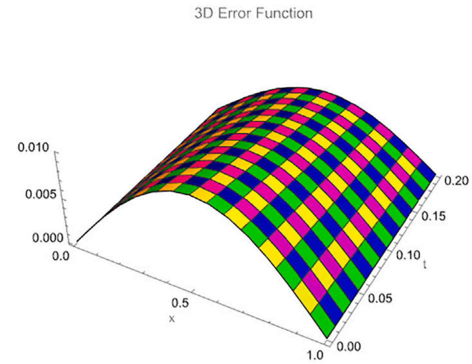


Fig. 3. 3D error profile for Example 1 when $\beta = 1.5$, $dt = 0.01$, $\gamma = 0$, $h = 0.0125$ and $T = 0.2$.

$$v(s,0.5) = \begin{cases} 0.5 + (0.2378)z - 0.5\cosh(z) + 0.0122\sinh(z), & s \in \left[0, \frac{1}{20}\right) \\ 0.5012 + (0.2127)z - 0.5012\cosh(z) + 0.0373\sinh(z), & s \in \left[\frac{1}{20}, \frac{1}{10}\right) \\ 0.5038 + (0.1877)z - 0.5038\cosh(z) + 0.06242\sinh(z), & s \in \left[\frac{1}{10}, \frac{3}{20}\right) \\ \vdots & \vdots \\ 0.6914 - (0.1877)z - 0.7040\cosh(z) + 0.4957\sinh(z), & s \in \left[\frac{17}{20}, \frac{9}{10}\right) \\ 0.7140 - (0.2127)z - 0.7297\cosh(z) + 0.5316\sinh(z), & s \in \left[\frac{9}{10}, \frac{19}{20}\right) \\ 0.7378 - (0.2378)z - 0.7572\cosh(z) + 0.5688\sinh(z), & s \in \left[\frac{19}{20}, 1\right) \end{cases}$$

Similarly, the approximate solution for $T = 1$, $\tau = 0.01$ and $M = 20$ is:

$$v(s,1) = \begin{cases} 2 + (0.9504)z - 2\cosh(z) + 0.0494\sinh(z), & s \in \left[0, \frac{1}{20}\right) \\ 2.005 + (0.8504)z - 2.005\cosh(z) + 0.1496\sinh(z), & s \in \left[\frac{1}{20}, \frac{1}{10}\right) \\ 2.0150 + (0.7503)z - 2.0150\cosh(z) + 0.2501\sinh(z), & s \in \left[\frac{1}{10}, \frac{3}{20}\right) \\ \vdots & \vdots \\ \vdots & \vdots \\ 2.7653 - (0.7503)z - 2.8154\cosh(z) + 1.9821\sinh(z), & s \in \left[\frac{17}{20}, \frac{9}{10}\right) \\ 2.8554 - (0.8504)z - 2.9181\cosh(z) + 2.1254\sinh(z), & s \in \left[\frac{9}{10}, \frac{19}{20}\right) \\ 2.9504 - (0.9504)z - 3.0281\cosh(z) + 2.2741\sinh(z), & s \in \left[\frac{19}{20}, 1\right) \end{cases}$$

Example 3. Consider the time-dependent fractional-order diffusion-wave equation:

$$\frac{\partial^\beta}{\partial t^\beta} v(z,t) + v(z,t) = \frac{\partial^2}{\partial z^2} v(z,t) + f(z,t), \quad (z,t) \in [0,1] \times [0,T],$$

with initial conditions (ICs):

$$v(z,0) = 0, \quad v_t(z,0) = 0,$$

and boundary condiitions (BCs):

$$v(0,t) = 0, \quad v(1,t) = t^2 \sinh(1).$$

The associated source term is [22]:

$$f(z,t) = \frac{2\sinh(z) t^{2-\beta}}{\Gamma[3-\beta]}.$$

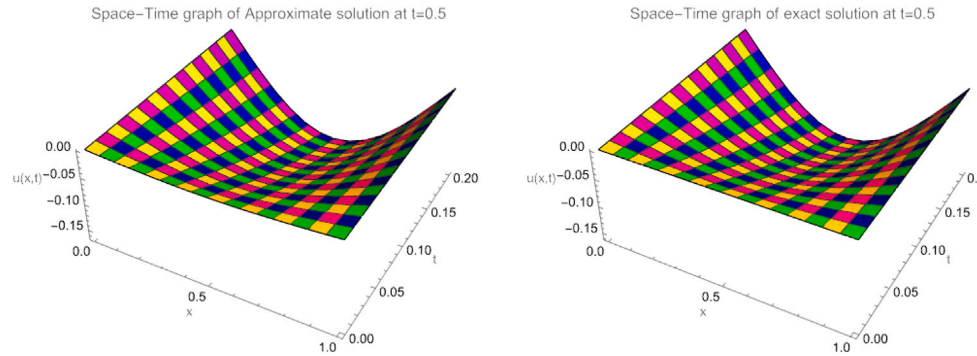


Fig. 4. Exact (right) and approximate (left) solutions for Example 1 with $\beta = 1.5$, $dt = 0.01$, $\gamma = 0$, $h = 0.0125$ and $T = 0.2$.

Table 7
Absolute errors for Example 2 at different (z, t) for varying β values.

(z,t)	$\beta = 1.1$	$\beta = 1.3$	$\beta = 1.5$	$\beta = 1.7$	$\beta = 1.9$
(0.1, 0.1)	3.275×10^{-9}	2.113×10^{-9}	1.300×10^{-9}	7.591×10^{-10}	4.209×10^{-10}
(0.2, 0.2)	3.225×10^{-8}	2.254×10^{-8}	1.479×10^{-8}	9.213×10^{-9}	5.480×10^{-9}
(0.3, 0.3)	1.158×10^{-7}	8.776×10^{-8}	6.133×10^{-8}	4.026×10^{-8}	2.531×10^{-8}
(0.4, 0.4)	2.693×10^{-7}	2.193×10^{-7}	1.648×10^{-7}	1.146×10^{-7}	7.558×10^{-8}
(0.5, 0.5)	4.840×10^{-7}	4.179×10^{-7}	3.375×10^{-7}	2.524×10^{-7}	1.764×10^{-7}
(0.6, 0.6)	7.228×10^{-7}	6.528×10^{-7}	5.609×10^{-7}	4.511×10^{-7}	3.401×10^{-7}
(0.7, 0.7)	9.171×10^{-7}	8.560×10^{-7}	7.727×10^{-7}	6.618×10^{-7}	5.360×10^{-7}
(0.8, 0.8)	9.636×10^{-7}	9.206×10^{-7}	8.623×10^{-7}	7.774×10^{-7}	6.678×10^{-7}
(0.9, 0.9)	7.198×10^{-7}	6.991×10^{-7}	6.722×10^{-7}	6.305×10^{-7}	5.683×10^{-7}

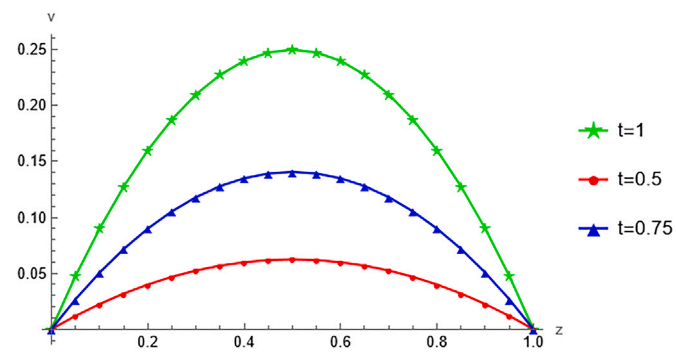


Fig. 5. Comparison between the exact solution (solid lines) and the approximate solution (symbols: circles, stars, triangles) for Example 2 at different time levels with $\beta = 1.5$, $\tau = 0.01$ and $h = 0.0125$.

Table 8
Spatial error norms and convergence rates p_h for various spatial resolutions M, with fixed time step $\tau = 1/160$ and fractional order $\beta = 1.5$.

M	L^∞ Error	L^2 Error	p_h
10	1.4676×10^{-4}	1.0845×10^{-4}	–
20	3.6624×10^{-5}	1.1914×10^{-5}	2.0025
40	9.1520×10^{-6}	3.3816×10^{-6}	2.0006
80	2.2878×10^{-6}	5.9773×10^{-7}	2.0002
160	5.7190×10^{-7}	1.0566×10^{-7}	2.0001

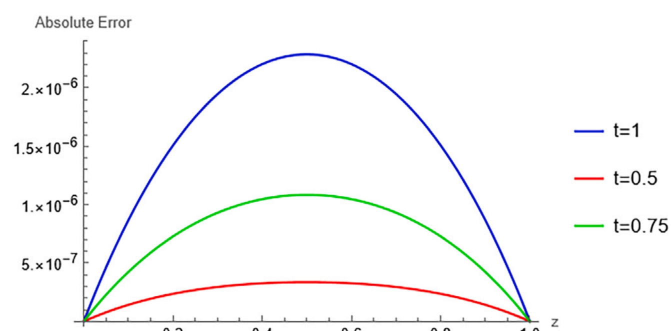


Fig. 6. Absolute errors for Example 2 at different time levels with $\beta = 1.5$, $\tau = 0.01$, $\gamma = 0$ and $h = 0.0125$.

Table 9
Temporal error norms and convergence rates p_τ for various time step sizes τ, with fixed spatial resolution $M = 50$ and fractional order $\beta = 1.5$.

τ	L^∞ Error	L^2 Error	p_τ
$\frac{1}{2}$	4.0972×10^{0}	1.3396×10^{0}	–
$\frac{1}{4}$	1.0122×10^{0}	3.3106×10^{-1}	2.0171
$\frac{1}{8}$	2.4458×10^{-1}	8.0076×10^{-2}	2.0492
$\frac{1}{16}$	5.4980×10^{-2}	1.8063×10^{-2}	2.1581
$\frac{1}{32}$	9.7759×10^{-3}	3.2566×10^{-3}	2.4868

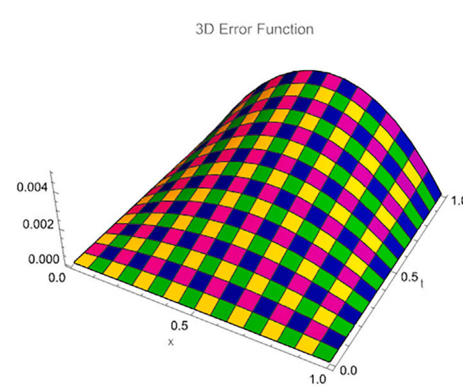


Fig. 7. 3D error profile for Example 2 with $\beta = 1.5$, $\tau = 0.01$, $\gamma = 0$, $T = 1$ and $h = 0.0125$..

The exact solution for Example 3 is $v(z, t) = t^2 \sinh(z)$. We apply the proposed numerical scheme to obtain numerical results. Table 10 presents the absolute errors at different points for varying values of β. Tables 11, 12 and 13 compare the correct results with those obtained using radial basis functions [15]. The spatial refinement results

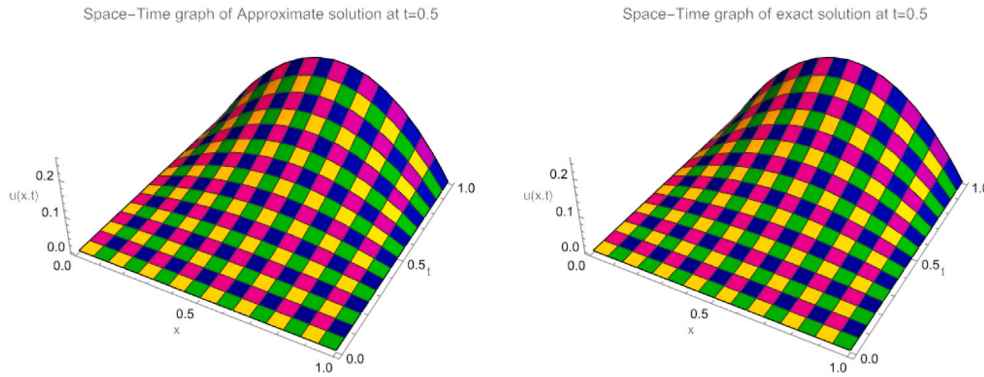


Fig. 8. Approximate (left) and exact (right) and solutions for Example 2 with $\beta = 1.5$, $\tau = 0.01$, $\gamma = 0$, $T = 1$ and $h = 0.0125$.

Table 10
Absolute errors for Example 3 at different (z, t) points across various β values.

(z,t)	$\beta = 1.1$	$\beta = 1.3$	$\beta = 1.5$	$\beta = 1.7$	$\beta = 1.9$
(0.1, 0.1)	5.320×10^{-15}	3.597×10^{-16}	2.452×10^{-15}	2.993×10^{-15}	3.207×10^{-15}
(0.2, 0.2)	3.252×10^{-15}	2.121×10^{-14}	2.648×10^{-14}	3.007×10^{-14}	3.075×10^{-14}
(0.3, 0.3)	6.592×10^{-14}	9.720×10^{-14}	1.054×10^{-13}	1.097×10^{-13}	1.099×10^{-13}
(0.4, 0.4)	2.142×10^{-13}	2.563×10^{-13}	2.695×10^{-13}	2.729×10^{-13}	2.694×10^{-13}
(0.5, 0.5)	4.744×10^{-13}	5.239×10^{-13}	5.463×10^{-13}	5.536×10^{-13}	5.399×10^{-13}
(0.6, 0.6)	5.330×10^{-12}	5.274×10^{-12}	5.235×10^{-12}	5.222×10^{-12}	5.255×10^{-12}
(0.7, 0.7)	8.640×10^{-12}	8.570×10^{-12}	8.531×10^{-12}	8.512×10^{-12}	8.571×10^{-12}
(0.8, 0.8)	1.787×10^{-11}	1.797×10^{-11}	1.799×10^{-11}	1.801×10^{-11}	1.794×10^{-11}
(0.9, 0.9)	2.630×10^{-11}	2.637×10^{-11}	2.638×10^{-11}	2.640×10^{-11}	2.634×10^{-11}

Table 11
Comparison of numerical results for Example 3 at $t = 1$ with $\beta = 1.5$ using different grid resolutions (M) contrasted with [15].

z	$M = 10$		$M = 20$		$M = 50$		Exact
	Present	[15]	Present	[15]	Present	[15]	
0.1	0.10016675	0.09950933	0.10016675	0.09951043	0.10016675	0.09951107	0.100167
0.2	0.20133600	0.20079972	0.20133600	0.20080274	0.20133600	0.20080391	0.201336
0.3	0.30452029	0.30402685	0.30452029	0.30402960	0.30452029	0.30403116	0.304520
0.4	0.41075232	0.41029949	0.41075232	0.41030268	0.41075232	0.41030448	0.410752
0.5	0.52109530	0.52065615	0.52109530	0.52065950	0.52109530	0.52066138	0.521095
0.6	0.63665358	0.63621617	0.63665358	0.63621936	0.63665358	0.63622115	0.636654
0.7	0.75858370	0.75812244	0.75858370	0.75812517	0.75858370	0.75812672	0.758584
0.8	0.88810598	0.88761746	0.88810598	0.88761947	0.88810598	0.88762062	0.888106
0.9	1.02651672	1.02593218	1.02651672	1.02593326	1.02651672	1.02593389	1.026520
1.0	1.17520119	1.17520119	1.17520119	1.17520119	1.17520119	1.17520119	1.175200

Table 12
Comparison of numerical results for Example 3 at $t = 1$ with $\beta = 1.25$ using different grid resolutions (M), contrasted with [15].

z	$M = 10$		$M = 50$		Exact
	Present	[15]	Present	[15]	
0.1	0.10016675	0.09950226	0.10016675	0.09950368	0.100167
0.2	0.20133600	0.20078691	0.20133600	0.20078954	0.201336
0.3	0.30452029	0.30400968	0.30452029	0.30401323	0.304520
0.4	0.41075232	0.41027962	0.41075232	0.41028373	0.410752
0.5	0.52109530	0.52063540	0.52109530	0.52063970	0.521095
0.6	0.63665358	0.63619639	0.636653582	0.63620048	0.636654
0.7	0.75858370	0.75810541	0.75858370	0.75810893	0.758584
0.8	0.88810598	0.88760481	0.88810598	0.88760740	0.888106
0.9	1.02651672	1.02592522	1.02651672	1.02592662	1.026520
1.0	1.17520119	1.17520119	1.17520119	1.17520119	1.175200

in Table 14 show a gradual increase in convergence rate with spatial convergence rate approaching second order. The temporal refinement study in Table 15 indicates that temporal convergence rate remains close to 2 demonstrating the robustness of the scheme. Fig. 9 visually compares the approximate solutions (symbols) and exact solutions (solid lines) at different time levels. Fig. 10 displays the absolute errors for

Table 13
Comparison of numerical results for Example 3 at $t = 1$ with $\beta = 1.75$ using different grid resolutions (M) contrasted with [15].

z	$M = 10$		$M = 50$		Exact
	Present	[15]	Present	[15]	
0.1	0.10016675	0.09951844	0.10016675	0.09952297	0.100167
0.2	0.20133600	0.20081683	0.20133600	0.09952297	0.201336
0.3	0.30452029	0.30405039	0.30452029	0.30406191	0.304520
0.4	0.41075232	0.41032718	0.41075232	0.41034071	0.410752
0.5	0.52109530	0.52068530	0.52109530	0.52069953	0.521095
0.6	0.63665358	0.63624390	0.63665358	0.63625745	0.636654
0.7	0.75858370	0.75814606	0.75858370	0.75815761	0.758584
0.8	0.88810598	0.88763465	0.88810598	0.88764309	0.888106
0.9	1.02651672	1.02594135	1.02651672	1.02594589	1.026520
1.0	1.17520119	1.17520119	1.17520119	1.17520119	1.175200

Example 3. Fig. 11 shows the 3D error profile, while Fig. 12 provides a 3D comparison between the approximate and exact solutions. The approximate solution for Example 3 with $T = 0.5$, $\tau = 0.01$ and $M = 20$ is:

$$V(s,0.5)=\begin{cases}6.564\times10^{-12}-(1.432\times10^{-10})z-6.561\times10^{-12}\cosh(z)+0.25\sinh(z), & s\in\left[0,\frac{1}{20}\right)\\ -1.042\times10^{-12}+(8.813\times10^{-12})z+1.044\times10^{-12}\cosh(z)+0.25\sinh(z), & s\in\left[\frac{1}{20},\frac{1}{10}\right)\\ -1.795\times10^{-13}+(1.789\times10^{-13})z+1.821\times10^{-13}\cosh(z)+0.25\sinh(z), & s\in\left[\frac{1}{10},\frac{3}{20}\right)\\ \vdots \\ \vdots \\ -4.796\times10^{-12}+(5.686\times10^{-12})z+5.698\times10^{-12}\cosh(z)+0.25\sinh(z), & s\in\left[\frac{17}{20},\frac{9}{10}\right)\\ 5.670\times10^{-12}-(6.412\times10^{-12})z-6.69953\times10^{-12}\cosh(z)+0.25\sinh(z), & s\in\left[\frac{9}{10},\frac{19}{20}\right)\\ -1.808\times10^{-11}+(1.929\times10^{-11})z+2.156\times10^{-11}\cosh(z)+0.25\sinh(z), & s\in\left[\frac{19}{20},1\right).\end{cases}$$

Similarly, the approximate solution for $T = 1$, $\tau = 0.01$ and $M = 20$ is:

$$V(s,1)=\begin{cases}-5.183\times10^{-12}+(6.607\times10^{-11})z+5.194\times10^{-12}\cosh(z)+\sinh(z), & s\in\left[0,\frac{1}{20}\right)\\ -4.460\times10^{-12}+(5.217\times10^{-11})z+4.496\times10^{-12}\cosh(z)+\sinh(z), & s\in\left[\frac{1}{20},\frac{1}{10}\right)\\ 2.152\times10^{-12}-(1.414\times10^{-11})z-2.150\times10^{-12}\cosh(z)+\sinh(z), & s\in\left[\frac{1}{10},\frac{3}{20}\right)\\ \vdots \\ \vdots \\ -2.650\times10^{-11}+(2.999\times10^{-11})z+2.871\times10^{-11}\cosh(z)+\sinh(z), & s\in\left[\frac{17}{20},\frac{9}{10}\right)\\ 2.464\times10^{-11}-(2.565\times10^{-11})z-2.839\times10^{-11}\cosh(z)+\sinh(z), & s\in\left[\frac{9}{10},\frac{19}{20}\right)\\ 4.785\times10^{-11}-(5.027\times10^{-11})z-5.541\times10^{-11}\cosh(z)+\sinh(z), & s\in\left[\frac{19}{20},1\right).\end{cases}$$

Table 14
Spatial error norms and convergence rates p_h for Example 3 with fractional order $\beta = 1.5$ and fixed time step $\tau = 1/160$.

M	L_2 Norm	L_∞ Norm	p_h
50	1.2367×10^{-4}	1.8304×10^{-4}	–
100	5.0019×10^{-5}	7.5480×10^{-5}	1.2780
200	1.8730×10^{-5}	2.8810×10^{-5}	1.3895
400	6.8398×10^{-6}	1.0705×10^{-5}	1.4283
800	1.5720×10^{-6}	2.5123×10^{-6}	2.0912

Table 15
Temporal error norms and convergence rates p_τ for Example 3 with fractional order $\beta = 1.5$ and fixed spatial resolution $M = 160$.

τ	L_2 Norm	L_∞ Norm	p_τ
$\frac{1}{2}$	0.8069	1.1254	–
$\frac{1}{4}$	0.1995	0.2782	2.0163
$\frac{1}{8}$	0.0483	0.0673	2.0468
$\frac{1}{16}$	0.0109	0.0152	2.1447
$\frac{1}{32}$	0.0020	0.0028	2.4506

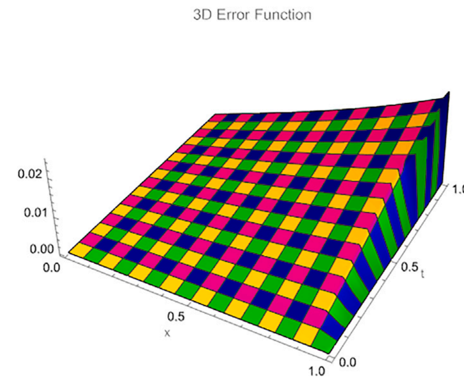


Fig. 11. 3D error profile for Example 3 when $\tau = 0.01$, $\beta = 1.5$, $T = 1$ and $h = 0.0125$.

6. Concluding remarks

This study presents a novel finite difference scheme that combines uniform hyperbolic polynomial B-spline basis functions for solving time-dependent fractional-order diffusion-wave equation with a Caputo fractional derivative in the reaction term. The introduced method exhibits high accuracy and computational efficiency and robustness, effectively capturing the complex dynamics inherent in fractional-order systems. The uniform hyperbolic polynomial B-spline basis ensures smooth solutions while maintaining numerical stability. The main advantages of the proposed method include its ability to deliver high accuracy on uniform grids, its unconditional stability and its suitability for capturing both local sharp gradients and global memory effects inherent in fractional

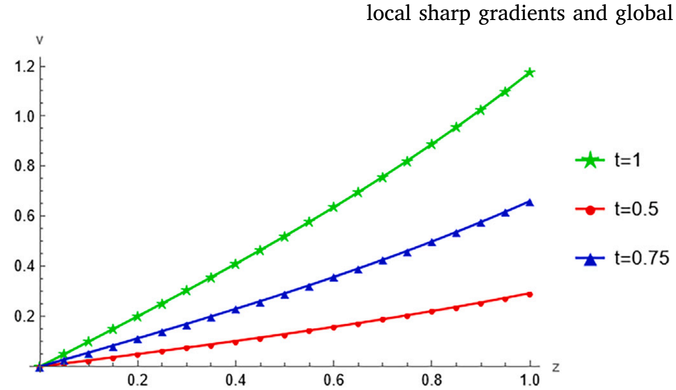


Fig. 9. Comparison between exact (solid lines) and approximate solution (symbols) for Example 3 at different time levels with $\tau = 0.01$, $\beta = 1.5$ and $h = 0.0125$.

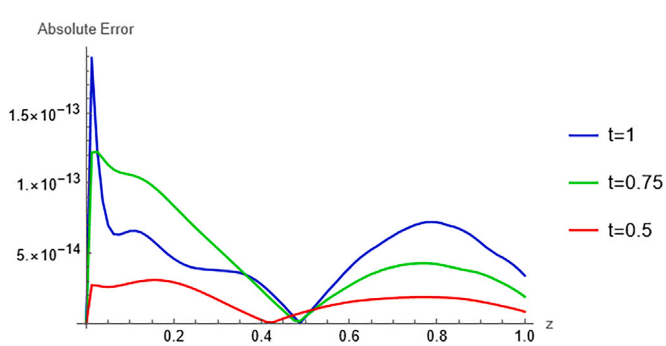


Fig. 10. Absolute errors for Example 3 at different time levels taking $\tau = 0.01$, $\beta = 1.5$ and $h = 0.0125$.

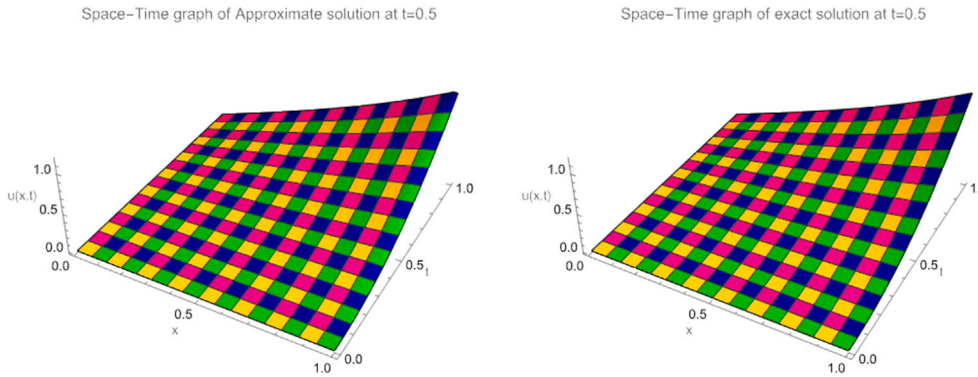


Fig. 12. Approximate (left) and exact(right) solutions for Example 3 when $\tau = 0.01$, $\beta = 1.5$, $T = 1$ and $h = 0.0125$.

models. The present analysis assumes smooth and bounded solutions and source terms, uniform spatial grids, and standard boundary conditions. In practical applications, these assumptions may not hold, for example, in problems with discontinuities, singularities, or irregular geometries. Extending the scheme to such cases, as well as to non-uniform meshes and adaptive grid strategies, constitutes a promising direction for future research.

CRediT authorship contribution statement

Aleeza Kiran: Writing – original draft, Software, Methodology, Formal analysis. **Muhammad Yaseen:** Writing – review & editing, Supervision, Conceptualization. **Aziz Khan:** Validation, Resources, Funding acquisition. **Thabet Abdeljawad:** Writing – review & editing, Validation, Data curation. **Manar A. Alqudah:** Writing – review & editing, Visualization, Investigation. **Rajermani Thinakaran:** Writing – review & editing, Validation, Investigation.

Funding

No funding was used in this study.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

Manar A. Alqudah would like to thank Princess Nourah bint Abdulrahman University researchers for supporting project number (PNURSP2025R14), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. A. Khan and T. Abduljawad would like to thank Prince Sultan University and TAS lab.

Data availability

All data that support the findings of this study are included in the article.

References

[1] Arqub OA. Numerical simulation of time-fractional partial differential equations arising in fluid flows VIA reproducing kernel method. Int J Numer Methods Heat Fluid Flow 2020;30(11):4711–33.
[2] Sadri K, Amilo D, Hncal E. A combination of classical and shifted Jacobi polynomials for two-dimensional time-fractional diffusion-wave equations. Chaos Solitons Fractals 2025;198:116569.
[3] Sadri K, Aminikhah H. A new efficient algorithm based on fifth-kind Chebyshev polynomials for solving multi-term variable-order time-fractional diffusion-wave equation. Int J Comput Math 2022;99:966–92.
[4] Zhou F, Xu X. Numerical solution of time-fractional diffusion-wave equations VIA Chebyshev wavelets collocation method. Adv Math Phys 2017;2017:1–17.
[5] Zeng F. Second-order stable finite difference schemes for the time-fractional diffusion-wave equation. J Sci Comput 2015;65:411–30.
[6] Khader MM, Adel MH. Numerical solutions of fractional wave equations using an efficient class of FDM based on the Hermite formula. Adv Differ Equ 2016;34:1–10.
[7] Jena SR, Senapati A. An improvised cubic b-spline collocation of fourth order and crank–nicolson technique for numerical solution of klein–gordon and sine–gordon equations. Iran J Sci 2025;49(2):383–407.
[8] Ali U, Iqbal A, Sohail M, Abdullah FA, Khan Z. Compact implicit difference approximation for time-fractional diffusion-wave equation. Alexandria Eng J 2022;61(5):4119–26.
[9] Salehi R. A meshless point collocation method for 2-d multi-term time fractional diffusion- wave equation. Numer Algor 2017;74:1145–68.
[10] Ali U, Sohail M, Usman M, Abdullah FA, Khan I, Nisar KS. Fourth-order difference approximation for time-fractional modified sub diffusion equation. Symmetry 2020;12(5):1–11.
[11] Fu ZJ, Yang LW, Zhu HQ, Xu WZ. A semi-analytical collocation Trefftz scheme for solving multi-term time fractional diffusion-wave equations. Eng Anal Bound Elem 2019;98:137–46.
[12] Jena SR, Senapati A. Explicit and implicit numerical investigations of one-dimensional heat equation based on spline collocation and Thomas algorithm. Soft Comput 2024:1–22.
[13] Jena SR, Senapati A. One-dimensional heat and advection-diffusion equation based on improvised cubic b-spline collocation, finite element method and Crank-Nicolson technique. Int Commun Heat Mass Transf 2023;147:106958.
[14] Arqub OA, Tayebi S, Baleanu D, Osman MS, Mahmoud W, Alsulami H. A numerical combined algorithm in cubic b-spline method and finite difference technique for the time-fractional nonlinear difusion-wave equation with reaction and damping terms. Results In Phys 2022;41(105912).
[15] Avazzadeh Z, Hosseini VR, Chen W. Radial basis functions and FDM for solving fractional diffusion-wave equation. Iran J Sci Technol 2014;2014:205–12.
[16] Heydari MH, Hooshmandasl MR, Ghaini FMM, Cattani C. Wavelets method for the time fractional diffusion-wave equation. Phys Lett A 2015;379(3):71–6.
[17] Ebadian A, Fazli HR, Khajehnasiri AA. Solution of nonlinear fractional diffusion-wave equation by traingular functions. sema J 2015;72:37–46.
[18] Nikan O, Machado JAT, Golbabai A. Numerical solution of time-fractional fourth-order reaction-diffusion model arising in composite environments. Appl Math Model 2021;87(1):819–36.
[19] Rashidinia J, Mohmedi E. Approximate solution of the multi-term time fractional diffusion and diffusion-wave equations. Comput Appl Math 2020;39(275):1–25.
[20] Dehghan M, Safarpoor M, Abbaszadeh M. Two high-order numerical algorithms for solving the multi-term time fractional diffusion-wave equations. J Comput Appl Math 2015;290:174–95.
[21] Wei Y, Zhao Y, Shi Z, Wang F, Tang Y. Spatial high accuracy analysis of FEM for two-dimensional multi-term time fractional diffusion-wave equations. Acta Math Appl Sin Engl Ser 2018;34(4):828–41.
[22] Yaseen M, Abbas M, Nazir T, Baleanu D. Afinite difference scheme based on cubic trigonometric b-splines for a time fractional diffusion-wave equation. Adv Differ Equ 2017;2017:1–18.
[23] Wang YM. A compact finite difference method for a class of time fractional convection diffusion-wave equations with variable coefficients. Numerical Algorithms 2015;70:625–51.
[24] Shah NA, Ahammad NA, Vieru D, Yook SJ, Hussam A. Analytical solutions for time-fractional diffusion equation with heat absorption in spherical domains. Ain Shams Eng J 2023;14(8):102031.
[25] Krishnaveni K, Kannan K, Balachandar SR, Venkatesh SG. An efficient fractional polynomial method for space fractional diffusion equations. Ain Shams Eng J 2018;9(4):767–76.
[26] Singh J, Swroop R, Kumar D. A computational approach for fractional convection-diffusion equation VIA integral transforms. Ain Shams Eng J 2018;9(4):1019–28.
[27] Farman M, Talib A, Nisar KS, Sambas A, Bayram M, Hafez M. Investigation of ABPV predict dynamics infection in honeybee colony production: soft patterns multiscale modeling with fractional approach. Ain Shams Eng J 2025;16(10):103626.

[28] Li X, Dong H. Analysis of a stabilized element-free Galerkin method for magnetohydrodynamic flow at very large Hartmann numbers. Appl Math Comput 2025;495:129334.
[29] Li X, Dong H. Analysis of a divergence-free element-free Galerkin method for the Navier-Stokes equations. Appl Numer Math 2025;217:73–95.
[30] Qing L, Li X. Analysis of a meshless generalized finite difference method for the time-fractional diffusion-wave equation. Comput Math Appl 2024;172:134–51.
[31] Logan JD. Applied partial Differential equations. Springer; 2012.
[32] Lu Y, Wang G, Yang X. Uniform hyperbolic polynomial B-spline curves. Comput Aided Geom Des 2002;19(6):379–93.

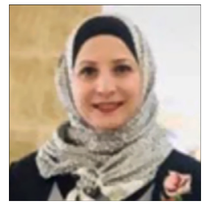
Author biography

Aleeza Kiran is a researcher at the Department of Mathematics, University of Sargodha, 40100, Sargodha,Pakistan. Her research focuses on finding numerical solutions of fractional differential equations using spline approaches.

Dr. Muhammad Yaseen is an accomplished mathematician and HEC Approved Supervisor currently serving as an Assistant Professor in the Department of Mathematics at the University of Sargodha, Pakistan. He has specialized in numerical analysis, fractional calculus, and computational mathematics. His research focuses on innovative numerical techniques, particularly cubic B-spline collocation methods, for solving fractional differential and integro-differential equations. Dr. Yaseen has an extensive publication record, with over 30 research articles in high-impact journals. Additionally, he has served as a guest editor for special issues in Fractal and Fractional and actively participates in academic committees. His contributions to mathematics education and research have earned him recognition.

Dr. Aziz Khan is a Researcher (Assistant Professor) in the Department of Mathematics and Sciences, Prince Sultan University. His research focuses on fractional order modeling, fluid mechanics, mathematical modeling, and artificial intelligence neural networking. He has published extensively on AI-driven fractional models and environmental simulations. Dr. Khan integrates mathematics and AI to develop sustainable, intelligent solutions for complex industrial and environmental systems through advanced computational modeling.

Prof. Thabet Abdeljawad is a distinguished Professor of Mathematics known for his pioneering work in fractional calculus, difference equations, and mathematical analysis. He has made significant contributions to the theory and applications of fractional differential and difference operators. Dr. Abdeljawad has authored numerous high impact publications and collaborates globally on advancing modern fractional calculus and its interdisciplinary applications in science and engineering.



Dr. Manar A. Alqudah is an accomplished researcher and academic specializing in applied mathematics, fractional calculus, and mathematical modeling. Her research focuses on developing and analyzing fractional-order differential equations and their applications in engineering and physical sciences. Dr. Alqudah has published several papers in reputable international journals and actively collaborates with scholars worldwide to advance modern analytical and numerical methods in fractional-order systems and dynamical analysis.

Dr. Rajermani Thinakaran is a Senior Lecturer at INTI International University, Malaysia. She holds a PhD in Computer Science from Universiti Teknologi Malaysia. Her research focuses on artificial intelligence, intelligent tutoring systems, assistive technologies, and e-learning innovation. Dr. Thinakaran has published widely in international journals and conferences and actively promotes inclusive education through AI-based learning systems and technology-enhanced teaching methodologies.