Electronic Research Archive

https://www.aimspress.com/journal/era

ERA, 33(9): 5518–5535.
DOI: 10.3934/era.2025247
Received: 19 May 2025
Revised: 03 August 2025
Accepted: 01 September 2025
Published: 16 September 2025

Research article

Linear barycentric rational collocation method for solving nonlinear time-fractional Cable equation

Bo Liu* and Di Liang

School of Science, Shandong Jianzhu University, Jinan 250101, China

* **Correspondence:** Email: liuboyp@sdjzu.edu.cn.

Abstract: The time-fractional Cable (TFC) equation constitutes a significant advancement of the classical Cable equation within the context of fractional calculus. In contrast to conventional numerical techniques such as finite difference and finite element methods, the barycentric interpolation method exhibits superior capability in handling the nonlocal properties inherent in fractional operators. This study thoroughly examines the principles and attributes of the barycentric interpolation method, integrating them with the unique aspects of the TFC equation to develop an appropriate matrix formulation. This transformation facilitates the conversion of the TFC equation into a solvable algebraic system. The proposed algorithm's accuracy and convergence rate are meticulously evaluated. Numerical experiments are performed to underscore the benefits of this approach in solving the TFC equation, thereby confirming the efficacy of the error analysis and convergence assessment.

Keywords: numerical calculation; barycentric collocation method; time-fractional Cable equation; nonlinear; error estimation

1. Introduction

The TFC equation, which extends the classical Cable equation through the lens of fractional calculus, plays a crucial role in advancing the study of neural dynamics. While the traditional Cable equation has limitations in accurately modeling nerve fiber activity, its fractional-order counterpart excels in simulating neuronal behavior, particularly in capturing the anomalous electrical diffusion of ions within spiny dendrites. This extension establishes a robust mathematical framework for neuroscience, thereby facilitating a deeper understanding of neural mechanisms and promoting advancements in related disciplines such as neuroscience and cognitive science.

The barycentric interpolation method demonstrates significant potential in solving TFC equations. Traditional numerical methods often struggle with computational complexity and accuracy due to the inherent intricacies of these equations. In contrast, the barycentric interpolation method offers high precision and efficiency, making it well-suited for handling fractional derivatives and complex boundary conditions. This approach provides a novel and accurate solution to TFC equations, thereby promising more reliable numerical simulations for neuroscience research and enhancing both theoretical and experimental studies.

A considerable body of research has been conducted on solving TFC equations, with scholars [1] proposing various numerical methods, each possessing unique advantages and limitations. ADI difference, implicit difference scheme, and extrapolation methods have been developed to solve the integro-partial differential equation [2], evolution equations [3], hyperbolic equations [4], subdiffusion equations [5, 6], and supergeneralized viscous Burgers' equations [7, 8]. Early approaches included the finite difference method [9], where researchers developed schemes such as explicit, implicit, and Crank–Nicolson methods [10] tailored to the characteristics of TFC equations. The finite element method [11] discretizes the solution domain into finite elements, representing the solution as a linear combination of interpolation functions defined on each element. The spectral method [12] and Galerkin spectral method [13–15] use orthogonal polynomials as basis functions to expand the solution into a series form, excelling in handling periodic boundary conditions [16, 17] but encountering difficulties with non-periodic conditions and complex geometries. Other methods [18, 19], including the finite volume method [20, 21], boundary element method [22, 23], and meshless methods [24, 25], have also been applied to solve TFC equations.

As an emerging numerical technique [26–28], the barycentric interpolation method [29–32] has received widespread attention and has been increasingly applied across various fields in recent years, including Volterra integro-differential equations [33] and Volterra differential equations [34–37] in recent years. In structural mechanics, it addresses problems [38–42] like beam bending and vibration and the EFK equation [43] and ZK-MEW equation [44] by constructing differential matrices of unknown functions and discretizing control equations into algebraic systems using point matching, resulting in high-precision numerical solutions. In heat conduction problems, the method accurately models temperature distributions and changes. Space fractional diffusion equations have been studied using multilevel tau preconditioners [45], finite volume methods [46], algebraic preconditioners [47], and spectral methods [48]. Although research on applying the barycentric interpolation method to fractional differential equations, such as the Riesz fractional derivative [49], is still in its early stages, preliminary results are promising. Scholars have successfully applied this method to fractional-order diffusion and wave equations, validating its effectiveness and high accuracy through numerical experiments. However, research on its application to time-fractional Cable (TFC) equations remains limited, with incomplete theoretical analysis and numerical experiments. Improving computational accuracy and stability for fractional derivatives and complex boundary conditions, as well as optimizing algorithms for enhanced efficiency, are critical issues that require further investigation.

In this paper, we focus on the numerical solution of the nonlinear time-fractional Cable equation given by

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = -\,{}_0C_t^{\alpha} u(\mathbf{x},t) + {}_0C_t^{\beta} \Delta u(\mathbf{x},t) - F(u) + g(\mathbf{x},t), \quad \mathbf{x} \in \Omega, \quad 0 \leqslant t \leqslant T, \tag{1.1}$$

subject to initial and boundary conditions:

$$u(\mathbf{x},t) = 0, \quad \mathbf{x} \in \Omega, \tag{1.2}$$

$$u(\mathbf{x},0) = u_0(\mathbf{x}), \quad t \in [0,1], \tag{1.3}$$

where $F(u) = u^3 - u$, $\alpha, \beta \in (0, 1)$ and ${}_0C_t^\alpha u(\mathbf{x}, t)$ is the fractional part.

The nonlinear time-fractional Cable equation represents an interdisciplinary research topic at the intersection of neurodynamics and fractional calculus. By incorporating both memory effects and nonlinear properties, it offers a more precise characterization of the complex evolution of neuronal electrical signals, thereby serving as a valuable mathematical tool for investigating neural functions and pathological mechanisms. Theoretical analysis and numerical methods associated with this equation constitute currently active research directions within the field of biomathematics.

This paper is structured as follows. Initially, we present a numerical algorithm for solving TFC equations using the barycentric interpolation method: This section delves into the principles and attributes of the barycentric interpolation technique, aligning them with the unique aspects of TFC equations to construct a robust numerical solution framework. The algorithm is designed to accurately manage fractional derivatives and intricate boundary conditions by converting TFC equations into solvable algebraic equations through strategic node placement and differential matrix formulation. Next, we perform an in-depth analysis of the algorithm's accuracy, convergence, and stability. A thorough theoretical evaluation assesses the algorithm's performance under diverse conditions, ensuring its reliability and robustness. This analysis provides a comprehensive understanding of the algorithm's strengths and limitations. Finally, we validate the algorithm's effectiveness through a series of numerical experiments. By applying the developed algorithm to a representative TFC equation, we demonstrate its advantages in solving such equations. These experiments offer compelling evidence that supports the practical utility of the barycentric interpolation method in both engineering and scientific research. In summary, this paper systematically develops and evaluates a novel numerical algorithm for TFC equations, showcasing its potential for real-world applications.

2. Differentiation matrices of the TFC equation

For the fractional part of the TFC equation, we express the Caputo fractional derivative as follows:

$$D_t^\alpha u(\mathbf{x}, t) = \frac{1}{\Gamma(\xi - \alpha)} \int_0^t \frac{\partial^\xi u(\mathbf{x}, s)}{\partial s^\xi} \frac{ds}{(t - s)^{\alpha + 1 - \xi}}, \tag{2.1}$$

which can be rewritten using integration by parts as:

$$D_t^\alpha u(\mathbf{x}, t) = \Gamma_\alpha^\xi \left[\frac{\partial^\xi u(\mathbf{x}, 0)}{\partial t^\xi} t^{\xi - \alpha} + \int_0^t \frac{\partial^{\xi+1} u(\mathbf{x}, s)}{\partial s^{\xi+1}} \frac{ds}{(t - s)^{\alpha - \xi}} \right], \tag{2.2}$$

where $\Gamma_\alpha^\xi = \frac{1}{(\xi - \alpha)\Gamma(\xi - \alpha)}$.

Similarly, for another fractional order γ, we have:

$$D_t^\gamma u(\mathbf{x}, t) = \Gamma_\gamma^\xi \left[\frac{\partial^\xi u(\mathbf{x}, 0)}{\partial t^\xi} t^{\xi - \gamma} + \int_0^t \frac{\partial^{\xi+1} u(\mathbf{x}, s)}{\partial s^{\xi+1}} \frac{ds}{(t - s)^{\gamma - \xi}} \right], \tag{2.3}$$

where $\Gamma_\gamma^\xi = \frac{1}{(\xi - \gamma)\Gamma(\xi - \gamma)}$.

2.1. Integral term approximation

The integral term in Eq (2.1) can be expressed as:

$$\int_0^{t_i} \frac{R_j^{(\xi+1)}(\tau) d\tau}{(t_i - \tau)^{\alpha - \xi}} = Q_j^\alpha(t_i) = Q_{ji}^\alpha, \tag{2.4}$$

and similarly,

$$\int_0^{t_i} \frac{R_j^{(\xi+1)}(\tau)d\tau}{(t_i-\tau)^{\gamma-\xi}} = Q_j^\gamma(t_i) = Q_{ji}^\gamma. \tag{2.5}$$

These integrals are computed using Gaussian quadrature with weights $\rho(\tau) = (t_\theta - \tau)^{\xi-\alpha}$ and $\rho(\tau) = (t_\theta - \tau)^{\xi-\gamma}$, respectively:

$$Q_j^\alpha(t_i) = \sum_{k=1}^{g} R_k^{(\xi+1)}(\tau_k^{\theta,\alpha})G_k^{\theta,\alpha}, \tag{2.6}$$

$$Q_j^\gamma(t_i) = \sum_{k=1}^{g} R_k^{(\xi+1)}(\tau_k^{\theta,\gamma})G_k^{\theta,\gamma}, \tag{2.7}$$

where $G_k^{\theta,\alpha}$ and $G_k^{\theta,\gamma}$ are the Gauss weights, and $\tau_k^{\theta,\alpha}$ and $\tau_k^{\theta,\gamma}$ are the Gauss points.

The TFC equation system can be written as:

$$\begin{aligned} u_t \; &- \; \Gamma_\alpha^\xi\left[\frac{\partial^\xi u(\mathbf{x},0)}{\partial t^\xi}t^{\xi-\alpha} + \int_0^t \frac{\partial^{\xi+1}u(\mathbf{x},s)}{\partial s^{\xi+1}}\frac{ds}{(t-s)^{\alpha-\xi}}\right] \\ &+ \; \Gamma_\gamma^\xi\left[\frac{\partial^\xi u(\mathbf{x},0)}{\partial t^\xi}t^{\xi-\gamma} + \int_0^t \frac{\partial^{\xi+1}u(\mathbf{x},s)}{\partial s^{\xi+1}}\frac{ds}{(t-s)^{\gamma-\xi}}\right] \\ &- \; F(\mathbf{x},t) = g(\mathbf{x},t), \quad (\mathbf{x},t) \in \Omega\times(0,T). \end{aligned} \tag{2.8}$$

2.2. Spatial and temporal discretization

Considering the domain $(\mathbf{x}, t) \in \Omega \times (0, T)$, where $\Omega = [a, b] \times [c, d]$ and $[0, T]$ is discretized into subdomains $\Omega_{ij} = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$, with $0 = t_0 < t_1 < \cdots < t_l = T$. The mesh points can be uniformly or quasi-uniformly partitioned (e.g., using Chebyshev points).

Using barycentric interpolation, the function $u(x, y, t)$ can be approximated as:

$$u(x,y,t) = \sum_{i=0}^{m}\sum_{j=0}^{n}\sum_{k=0}^{l} \Phi_i(x)\Phi_j(y)\Phi_k(t)u_{ijk}, \tag{2.9}$$

where

$$u_{ijk} = u(x_i, y_j, t_k), \tag{2.10}$$

and the basis functions are defined as:

$$\Phi_i(x) = \frac{\frac{w_i}{x-x_i}}{\sum_{s=0}^{m}\frac{w_s}{x-x_s}}, \quad w_i = \sum_{j\in J_i}(-1)^j \prod_{k=j,k\neq i}^{j+d_1}\frac{1}{x_i - x_k}, \tag{2.11}$$

with $J_i = \{j \in \{0, 1, \cdots, m - d_1\} : i - d_1 \le j \le i\}, d_1 \in N$.

Similarly,

$$\Phi_j(y) = \frac{\frac{v_j}{y-y_j}}{\sum_{s=0}^{n}\frac{v_s}{y-y_s}}, \quad v_j = \sum_{i\in J_j}(-1)^i \prod_{k=i,k\neq j}^{i+d_2}\frac{1}{y_j - y_k}, \tag{2.12}$$

with $J_j = \{i \in \{0, 1, \cdots, n - d_2\} : j - d_2 \le i \le j\}, d_2 \in N$.

And,

$$\Phi_k(t) = \frac{\frac{\lambda_k}{t-t_k}}{\sum_{s=0}^{l} \frac{\lambda_s}{t-t_s}}, \quad \lambda_k = \sum_{j\in J_k} (-1)^j \prod_{i=j, i\neq k}^{j+d_3} \frac{1}{t_k - t_i}, \tag{2.13}$$

with $J_k = \{j \in \{0, 1, \cdots, l - d_3\} : k - d_3 \leq j \leq k\}, d_3 \in N$.

2.3. Higher-order derivatives

The higher-order derivatives of $u(x, y, t)$ can be expressed as:

$$\frac{\partial^{(g_1+g_2+g_3)} u_{(m,n,l)}(x, y, t)}{\partial x^{g_1} \partial y^{g_2} \partial t^{g_3}} = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{l} \Phi_i^{(g_1)}(x) \Phi_j^{(g_2)}(y) \Phi_k^{(g_3)}(t) u_{ijk}, \tag{2.14}$$

where $g_1, g_2, g_3 > 0$ are integers. When $g_1 = g_2 = g_3 = 0$, this reduces to Eq (2.9).

Evaluating at specific points $(x_\alpha, y_\beta, t_\gamma)$, we get:

$$\frac{\partial^{(g_1+g_2+g_3)} u_{(m,n,l)}(x_\alpha, y_\beta, t_\gamma)}{\partial x^{g_1} \partial y^{g_2} \partial t^{g_3}} = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{l} \Phi_i^{(g_1)}(x_\alpha) \Phi_j^{(g_2)}(y_\beta) \Phi_k^{(g_3)}(t_\gamma) u_{ijk}. \tag{2.15}$$

This can be represented in matrix form as:

$$\frac{\partial^{(g_1+g_2+g_3)} u_{(m,n,l)}(x_\alpha, y_\beta, t_\gamma)}{\partial x^{g_1} \partial y^{g_2} \partial t^{g_3}} = (\Phi^{(g_1 00)} \otimes \Phi^{(0 g_2 0)} \otimes \Phi^{(00 g_3)}) u, \tag{2.16}$$

where $\otimes$ denotes the Kronecker product. To represent each element in the differential matrices, we define:

$$\Phi_{ij}^{(g_1 00)} = \Phi_j^{(g_1)}(x_i), \quad \Phi_{ij}^{(0 g_2 0)} = \Phi_j^{(g_2)}(y_i), \quad \Phi_{ij}^{(00 g_3)} = \Phi_j^{(g_3)}(t_i), \tag{2.17}$$

$$\Phi^{(g_1 00)} = \begin{pmatrix} \Phi_1^{(g_1)}(x_1) & \Phi_2^{(g_1)}(x_1) & \cdots & \Phi_m^{(g_1)}(x_1) \\ \Phi_1^{(g_1)}(x_2) & \Phi_2^{(g_1)}(x_2) & \cdots & \Phi_m^{(g_1)}(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_1^{(g_1)}(x_m) & \Phi_2^{(g_1)}(x_m) & \cdots & \Phi_m^{(g_1)}(x_m) \end{pmatrix}, \ \Phi^{(0 g_2 0)} = \begin{pmatrix} \Phi_1^{(g_2)}(y_1) & \Phi_2^{(g_2)}(y_1) & \cdots & \Phi_n^{(g_2)}(y_1) \\ \Phi_1^{(g_2)}(y_2) & \Phi_2^{(g_2)}(y_2) & \cdots & \Phi_n^{(g_2)}(y_2) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_1^{(g_2)}(y_n) & \Phi_2^{(g_2)}(y_n) & \cdots & \Phi_n^{(g_2)}(y_n) \end{pmatrix},$$

$$\Phi^{(00 g_3)} = \begin{pmatrix} \Phi_1^{(g_3)}(t_1) & \Phi_2^{(g_3)}(t_1) & \cdots & \Phi_l^{(g_3)}(t_1) \\ \Phi_1^{(g_3)}(t_2) & \Phi_2^{(g_3)}(t_2) & \cdots & \Phi_l^{(g_3)}(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_1^{(g_3)}(t_l) & \Phi_2^{(g_3)}(t_l) & \cdots & \Phi_l^{(g_3)}(t_l) \end{pmatrix}, \tag{2.18}$$

and

$$u = (u_{111}, \cdots, u_{11l}, u_{121}, \cdots, u_{12l}, \cdots, \cdots, u_{mn1}, \cdots, u_{mnl})^T, \tag{2.19}$$

here, $\Phi^{(g_1 00)}$ is the g_1-order differential matrix with space variable x; $\Phi^{(0 g_2 0)}$ is the g_2-order differential matrix with space variable y; $\Phi^{(00 g_3)}$ is the g_3-order differential matrix with time variable t, when $g_1 = g_2 = g_3 = 0$, $M^{(g_1 00)} = I_m, M^{(0 g_2 0)} = I_n, M^{(00 g_3)} = I_l$, where I_m, I_n, and I_l are m, n and l-order identity matrix, respectively. In order to conveniently represent each element in the differential matrix, we let $\Phi_{ij}^{(g_1 00)} = \Phi_j^{(g_1)}(x_i)$; $\Phi_{ij}^{(0g_2 0)} = \Phi_j^{(g_2)}(y_i)$; $\Phi_{ij}^{(00g_3)} = \Phi_j^{(g_3)}(t_i)$, the 1-order differential matrix as follows (see Ref. [38])

$$\Phi_{ij}^{(100)} = \begin{cases} \dfrac{\omega_j/\omega_i}{x_i - x_j}, & i \neq j, \\ -\sum\limits_{j \neq i} \Phi_{ij}^{(100)}, & i = j, \end{cases} \quad \Phi_{ij}^{(010)} = \begin{cases} \dfrac{\nu_j/\nu_i}{y_i - y_j}, & i \neq j, \\ -\sum\limits_{j \neq i} \Phi_{ij}^{(010)}, & i = j, \end{cases} \quad \Phi_{ij}^{(001)} = \begin{cases} \dfrac{\lambda_j/\lambda_i}{t_i - t_j}, & i \neq j, \\ -\sum\limits_{j \neq i} \Phi_{ij}^{(001)}, & i = j, \end{cases} \tag{2.20}$$

and the recurrence formulas of higher-order differential matrices as follows (see Ref. [38])

$$\Phi_{ij}^{(g_1 00)} = \begin{cases} g_1(\Phi_{ii}^{((g_1-1)00)} \Phi_{ij}^{(100)} - \dfrac{\Phi_{ij}^{((g_1-1)00)}}{x_i - x_j}), & i \neq j, \\ -\sum\limits_{j \neq i} \Phi_{ij}^{(g_1 00)}, & i = j, \end{cases} \quad g_1 = 2, 3, \cdots, \tag{2.21}$$

$$\Phi_{ij}^{(0g_2 0)} = \begin{cases} g_2(M_{ii}^{(0(g_2-1)0)} \Phi_{ij}^{(010)} - \dfrac{\Phi_{ij}^{(0(g_2-1)0)}}{y_i - y_j}), & i \neq j, \\ -\sum\limits_{j \neq i} \Phi_{ij}^{(0g_2 0)}, & i = j, \end{cases} \quad g_2 = 2, 3, \cdots, \tag{2.22}$$

$$\Phi_{ij}^{(00g_3)} = \begin{cases} g_3(\Phi_{ii}^{(00(g_3-1))} \Phi_{ij}^{(001)} - \dfrac{\Phi_{ij}^{(00(g_3-1))}}{t_i - t_j}), & i \neq j, \\ -\sum\limits_{j \neq i} \Phi_{ij}^{(00g_3)}, & i = j, \end{cases} \quad g_3 = 2, 3, \cdots, \tag{2.23}$$

The matrices $\Phi^{(g_1 00)}$, $\Phi^{(0g_2 0)}$, and $\Phi^{(00g_3)}$ are the g_1-, g_2-, and g_3-order differential matrices for the spatial variables x and y, and the time variable t, respectively. When $g_1 = g_2 = g_3 = 0$, these matrices reduce to identity matrices of appropriate dimensions.

3. Linearized of TFC equation

In this section, we introduce three discretization methods for solving the nonlinear Cable equation. Each method is described in detail below.

3.1. Direct linearized iterative scheme (DLIS)

By taking (2.9) into (2.7), we get

$$\begin{aligned}
&\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\Phi_i(x)\Phi_j(y)\Phi_k'(t)u_{lmn}\\
&+\Gamma_\alpha^\xi\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\left[\Phi_i(x)\Phi_j(y)\Phi_k^{(\xi)}(0)t^{\xi-\alpha}+\Phi_i(x)\Phi_j(y)\int_0^t\frac{\partial^{\xi+1}\Phi_k(s)}{\partial s^{\xi+1}}\frac{ds}{(t-s)^{\alpha-\xi}}\right]u_{lmn}\\
&-\Gamma_\gamma^\xi\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\left[\Phi_i''(x)\Phi_j(y)\Phi_k^{(\xi)}(0)t^{\xi-\alpha}+\Phi_i(x)\Phi_j(y)\int_0^t\frac{\partial^{\xi+1}\Phi_k(s)}{\partial s^{\xi+1}}\frac{ds}{(t-s)^{\alpha-\xi}}\right]u_{lmn}\\
&-\Gamma_\gamma^\xi\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\left[\Phi_i(x)\Phi_j(y)\Phi_k^{(\xi)}(0)t^{\xi-\alpha}+\Phi_i(x)\Phi_j''(y)\int_0^t\frac{\partial^{\xi+1}\Phi_k(s)}{\partial s^{\xi+1}}\frac{ds}{(t-s)^{\alpha-\xi}}\right]u_{lmn}\\
&-\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\Phi_i(x)\Phi_j(y)\Phi_k(t)u_{lmn}=f(x,y,t)-(\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\Phi_i(x)\Phi_j(y)\Phi_k(t)u_{lmn})^3.
\end{aligned}\tag{3.1}$$

Let $x=x_i, y=y_j, t=t_k, i,j,k=1,2,\cdots,m$, we get

$$\begin{aligned}
&\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\Phi_i(x_i)\Phi_j(y_j)\Phi_k'(t_k)u_{lmn}\\
&+\Gamma_\alpha^\xi\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\left[\Phi_i(x_i)\Phi_j(y_j)\Phi_k^{(\xi)}(0)t_k^{\xi-\alpha}+\Phi_i(x_i)\Phi_j(y_j)\int_0^{t_k}\frac{\partial^{\xi+1}\Phi_k(s)}{\partial s^{\xi+1}}\frac{ds}{(t-s)^{\alpha-\xi}}\right]u_{lmn}\\
&-\Gamma_\gamma^\xi\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\left[\Phi_i''(x_i)\Phi_j(y_j)\Phi_k^{(\xi)}(0)t_k^{\xi-\alpha}+\Phi_i(x_i)\Phi_j(y_j)\int_0^{t_k}\frac{\partial^{\xi+1}\Phi_k(s)}{\partial s^{\xi+1}}\frac{ds}{(t-s)^{\alpha-\xi}}\right]u_{lmn}\\
&-\Gamma_\gamma^\xi\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\left[\Phi_i(x_i)\Phi_j(y_j)\Phi_k^{(\xi)}(0)t_k^{\xi-\alpha}+\Phi_i(x_i)\Phi_j''(y_j)\int_0^{t_k}\frac{\partial^{\xi+1}\Phi_k(s)}{\partial s^{\xi+1}}\frac{ds}{(t-s)^{\alpha-\xi}}\right]u_{lmn}\\
&-\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\Phi_i(x_i)\Phi_j(y_j)\Phi_k(t_k)u_{lmn}=f(x_i,y_j,t_k)-(\sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\Phi_i(x_i)\Phi_j(y_j)\Phi_k(t_k)u_{lmn})^3.
\end{aligned}\tag{3.2}$$

Then we get the matrix equation as

$$\begin{aligned}
&\Gamma_\alpha^\xi\left(I_m\otimes I_n\otimes M_1^{(\xi 0)}+I_m\otimes I_n\otimes Q^\alpha\right)u-\Gamma_\gamma^\xi\left(M^{(002)}\otimes I_n\otimes M_1^{(\xi 0)}+I_n\otimes I_m\otimes Q^\gamma\right)u\\
&+(I_m\otimes I_n\otimes M^{(001)})u-\Gamma_\gamma^\xi\left(I_n\otimes I_m\otimes M_1^{(\xi 0)}+I_m\otimes M^{(020)}\otimes Q^\gamma\right)u-u=f-u_0^3,
\end{aligned}\tag{3.3}$$

then

$$\begin{aligned}
&[\Gamma_\alpha^\xi\left(I_m\otimes I_n\otimes M_1^{(\xi 0)}+I_m\otimes I_n\otimes Q^\alpha\right)-\Gamma_\gamma^\xi\left(M^{(002)}\otimes I_n\otimes M_1^{(\xi 0)}+I_n\otimes I_m\otimes Q^\gamma\right)\\
&+(I_m\otimes I_n\otimes M^{(001)})-\Gamma_\gamma^\xi\left(I_n\otimes I_m\otimes M_1^{(\xi 0)}+I_m\otimes M^{(020)}\otimes Q^\gamma\right)-I_{mnl}]u=f-u_0^3,
\end{aligned}\tag{3.4}$$

where $I_{mnl} = I_m \otimes I_n \otimes I_l$, h is iterative number, and $U = \begin{bmatrix} \phi_{111} \\ \vdots \\ \phi_{11n} \\ \phi_{l11} \\ \vdots \\ \phi_{lmn} \end{bmatrix}, F = \begin{bmatrix} f_{111} \\ \vdots \\ f_{11n} \\ f_{l11} \\ \vdots \\ f_{lmn} \end{bmatrix}$. Then the DLIS is constructed as follows

$$\begin{aligned}&[\Gamma_\alpha^\xi \left(I_m \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes I_n \otimes Q^\alpha\right) - \Gamma_\gamma^\xi \left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_n \otimes I_m \otimes Q^\gamma\right) \\ &+ (I_m \otimes I_n \otimes M^{(001)}) - \Gamma_\gamma^\xi \left(I_n \otimes I_m \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes Q^\gamma\right) - I_{mnl}]u_h = f - u_{h-1}^3. \end{aligned} \tag{3.5}$$

We let

$$\begin{aligned}L = &[\Gamma_\alpha^\xi \left(I_m \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes I_n \otimes Q^\alpha\right) - \Gamma_\gamma^\xi \left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_n \otimes I_m \otimes Q^\gamma\right) \\ &+ (I_m \otimes I_n \otimes M^{(001)}) - \Gamma_\gamma^\xi \left(I_n \otimes I_m \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes Q^\gamma\right) - I_{mnl}],\end{aligned} \tag{3.6}$$

then DLIS can be written as follows:

$$Lu_h = f - u_{h-1}^3, \tag{3.7}$$

where $h \geq 1$, $h \in N^+$, N^+ is a positive integer set.

3.2. Partial linearized iterative scheme (PLIS)

We separate u^3 into u^2u, u_0^2u is obtained by substituting u_0 into the nonlinear part u^2, then we put the u_0^2u into Eq (1.1)

$$\begin{aligned}&\Gamma_\alpha^\xi \left(I_m \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes I_n \otimes Q^\alpha\right) u - \Gamma_\gamma^\xi \left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_n \otimes I_m \otimes Q^\gamma\right) u \\ &+ (I_m \otimes I_n \otimes M^{(001)})u - \Gamma_\gamma^\xi \left(I_n \otimes I_m \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes Q^\gamma\right) u - u + diag(u_0^2)u = f,\end{aligned} \tag{3.8}$$

where $diag(u_0^2)$ is a diagonal matrix whose diagonal elements are elements in u_0^2, then

$$\begin{aligned}&[\Gamma_\alpha^\xi \left(I_m \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes I_n \otimes Q^\alpha\right) - \Gamma_\gamma^\xi \left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_n \otimes I_m \otimes Q^\gamma\right) \\ &+ [\Gamma_\alpha^\xi \left(I_m \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes I_n \otimes Q^\alpha\right) - \Gamma_\gamma^\xi \left(I_n \otimes I_m \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes Q^\gamma\right) \\ &-I_{mnl} + diag(u_0^2)]u = f,\end{aligned} \tag{3.9}$$

the PLIS is constructed as

$$\begin{aligned}&[\Gamma_\alpha^\xi \left(I_m \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes I_n \otimes Q^\alpha\right) - \Gamma_\gamma^\xi \left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_n \otimes I_m \otimes Q^\gamma\right) \\ &+ (I_m \otimes I_n \otimes M^{(001)}) - \Gamma_\gamma^\xi \left(I_n \otimes I_m \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes Q^\gamma\right) - I_{mnl} + diag(u_{h-1}^2)]u_h = f.\end{aligned} \tag{3.10}$$

Similarly, the can be expressed as

$$[L + diag(u_{h-1}^2)]u_h = f. \tag{3.11}$$

3.3. Newton iterative scheme (NIS)

The nonlinear term u^3 of (1.1) is expanded at u_0 by use of the Taylor formula; we have

$$u^3 \approx u_0^3 + 3u_0^2(u - u_0) + o(u - u_0) + \cdots, \tag{3.12}$$

we ignore $o(u - u_0)$ and keep the linear term $u^3 \cong u_0^3 + 3u_0^2(u - u_0) = 3u_0^2 u - 2u_0^3$, put it into (1.1); we can get

$$\Gamma_\gamma^\xi\left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_n \otimes I_m \otimes Q^\gamma\right)u - \Gamma_\gamma^\xi\left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes M_1^{(\xi 0)}\right)u$$
$$+ (I_m \otimes I_n \otimes M^{(001)})u - \Gamma_\gamma^\xi\left(I_n \otimes I_m \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes Q^\gamma\right)u - u + 3diag(u_0^2)u = f + 2u_0^3, \tag{3.13}$$

similarly, we have

$$[\Gamma_\gamma^\xi\left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_n \otimes I_m \otimes Q^\gamma\right) - \Gamma_\gamma^\xi\left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes M_1^{(\xi 0)}\right)$$
$$+ \Gamma_\alpha^\xi\left(I_m \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes I_n \otimes Q^\alpha\right) - \Gamma_\gamma^\xi\left(I_n \otimes I_m \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes Q^\gamma\right)$$
$$- I_{mnl} + 3diag(u_0^2)]u = f + 2u_0^3, \tag{3.14}$$

then the NIS can be constructed

$$[\Gamma_\gamma^\xi\left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_n \otimes I_m \otimes Q^\gamma\right) - \Gamma_\gamma^\xi\left(M^{(002)} \otimes I_n \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes M_1^{(\xi 0)}\right)$$
$$+ (I_m \otimes I_n \otimes M^{(001)}) - \Gamma_\gamma^\xi\left(I_n \otimes I_m \otimes M_1^{(\xi 0)} + I_m \otimes M^{(020)} \otimes Q^\gamma\right)$$
$$- I_{mnl} + 3diag(u_{h-1}^2)]u_h = f + 2u_{h-1}^3. \tag{3.15}$$

The NIS can be written as

$$[L + 3diag(u_{h-1}^2)]u_h = f + 2u_{h-1}^3. \tag{3.16}$$

4. Error estimation of TFC equation with DLIS

For functions $u(x), u(y), u(t) \in C[-1, 1]$, we define the interpolation operators $\mathcal{L}_{x,l} : C[-1, 1] \to J_x$, $\mathcal{L}_{y,m} : C[-1, 1] \to J_y$, and $\mathcal{L}_{t,n} : C[-1, 1] \to J_t$ for the variables x, y, and t respectively. These operators are given by:

$$\mathcal{L}_{x,l}u(x) = \sum_{i=1}^{l} \Phi_i(x)u_i, \quad \mathcal{L}_{y,m}u(y) = \sum_{j=1}^{m} \Phi_j(y)u_j, \quad \mathcal{L}_{t,n}u(t) = \sum_{k=1}^{n} \Phi_k(t)u_k. \tag{4.1}$$

We also define the combined interpolation operator $\mathcal{L}_{x,l}\mathcal{L}_{y,m} : C([-1, 1] \times [-1, 1]) \to J_{xy}$ as:

$$\mathcal{L}_{x,l}\mathcal{L}_{y,m}u(x, y) = \sum_{i=1}^{l} \sum_{j=1}^{m} \Phi_i(x)\Phi_j(y)u_{ij}. \tag{4.2}$$

Similarly, we define the three-dimensional interpolation operator $\mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n} : C([-1,1]\times[-1,1]\times[-1,1]) \to J$ as:

$$\mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n}u(x,y,t) = \sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}\Phi_i(x)\Phi_j(y)\Phi_k(t)u_{ijk}. \tag{4.3}$$

It can be shown that $u_{lmn} := \mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n}u$ and the operators $\mathcal{L}_{x,l}$, $\mathcal{L}_{y,m}$, $\mathcal{L}_{t,n}$, $\mathcal{L}_{x,l}\mathcal{L}_{y,m}$, and $\mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n}$ are all linear.

The Lebesgue constant Λ_n is defined as:

$$\Lambda_n = \|\mathcal{L}_{x,l}\|_\infty = \max_{x\in[-1,1]}\sum_{i=1}^{l}|\Phi_i(x)|. \tag{4.4}$$

Lemma 1. *For Chebyshev nodes of the second kind, the Lebesgue constant satisfies:*

$$\Lambda_n \le \frac{2}{\pi}\log(n+1)+1, \tag{4.5}$$

as shown in [29].

Lemma 2. *For equidistant nodes with basis functions $\Phi_i(x)$, the Lebesgue constant satisfies:*

$$\Lambda_n \le 2^{d-1}(2+\ln n), \tag{4.6}$$

as derived in [30].

We now present a theorem regarding the error bound for the interpolation operators:

Theorem 1. *For the interpolation operators $\mathcal{L}_{x,l}$, $\mathcal{L}_{y,m}$, $\mathcal{L}_{t,n}$, $\mathcal{L}_{x,l}\mathcal{L}_{y,m}$, and $\mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n}$ defined above, the following inequality holds:*

$$|u(x,y,t) - u_{lmn}(x,y,t)| \le C(h_x^{d_1+1} + \Lambda_l h_y^{d_2+1} + \Lambda_l\Lambda_m h_t^{d_3+1}), \tag{4.7}$$

where $C = \max\{\frac{\|\phi_x^{(l)}(x_\xi,y,t)\|_\infty}{(l+1)!}, \frac{\|\phi_y^{(m)}(x,y_\xi,t)\|_\infty}{(m+1)!}, \frac{\|\phi_t^{(n)}(x,y,t_\xi)\|_\infty}{(n+1)!}\}$.

Proof. To prove this theorem, we start by considering the infinity norm of the difference between the original function and its interpolated version. By the triangle inequality, we have

$$\begin{aligned}
&\|u(x,y,t) - u_{lmn}(x,y,t)\|_\infty \\
&= \|u(x,y,t) - \mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n}u(x,y,t)\|_\infty \\
&= \|u(x,y,t) - \mathcal{L}_{x,l}u(x,y,t) + \mathcal{L}_{x,l}u(x,y,t) - \mathcal{L}_{x,l}\mathcal{L}_{y,m}u(x,y,t) \\
&\quad + \mathcal{L}_{x,l}\mathcal{L}_{y,m}u(x,y,t) - \mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n}u(x,y,t)\|_\infty \\
&\le \|u(x,y,t) - \mathcal{L}_{x,l}u(x,y,t)\|_\infty + \|\mathcal{L}_{x,l}u(x,y,t) - \mathcal{L}_{x,l}\mathcal{L}_{y,m}u(x,y,t)\|_\infty \\
&\quad + \|\mathcal{L}_{x,l}\mathcal{L}_{y,m}u(x,y,t) - \mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n}u(x,y,t)\|_\infty.
\end{aligned} \tag{4.8}$$

We analyze each term separately:

Error in x-direction:

$$\|u(x,y,t)-\mathcal{L}_{x,l}u(x,y,t)\|_\infty = h_x^{d_1+1}\frac{\|\phi_x^{(l)}(x_\xi,y,t)\|_\infty}{(l+1)!}. \tag{4.9}$$

Error in y-direction:

$$\begin{aligned}
&\|\mathcal{L}_{x,l}u(x,y,t)-\mathcal{L}_{x,l}\mathcal{L}_{y,m}u(x,y,t)\|_\infty\\
&\le \|\mathcal{L}_{x,l}[u(x,y,t)-\mathcal{L}_{y,m}u(x,y,t)]\|_\infty\\
&\le \|\mathcal{L}_{x,l}\|_\infty\|u(x,y,t)-\mathcal{L}_{y,m}u(x,y,t)\|_\infty\\
&\le \Lambda_l h_y^{d_2+1}\frac{\|\phi_y^{(m)}(x,y_\xi,t)\|_\infty}{(m+1)!}.
\end{aligned} \tag{4.10}$$

Error in t-direction:

$$\begin{aligned}
&\|\mathcal{L}_{x,l}\mathcal{L}_{y,m}u(x,y,t)-\mathcal{L}_{x,l}\mathcal{L}_{y,m}\mathcal{L}_{t,n}u(x,y,t)\|_\infty\\
&\le \|\mathcal{L}_{x,l}[\mathcal{L}_{y,m}u(x,y,t)-\mathcal{L}_{y,m}\mathcal{L}_{t,n}u(x,y,t)]\|_\infty\\
&\le \|\mathcal{L}_{x,l}\|_\infty\|\mathcal{L}_{y,m}\|_\infty\|u(x,y,t)-\mathcal{L}_{t,n}u(x,y,t)\|_\infty\\
&\le \Lambda_l\Lambda_m h_t^{d_3+1}\frac{\|\phi_t^{(n)}(x,y,t_\xi)\|_\infty}{(n+1)!}.
\end{aligned} \tag{4.11}$$

Combining these results, we obtain the desired error bound, thus completing the proof of Theorem 1.

5. Numerical example

In this part, numerical examples are presented to illustrate our theorem.

Example 1. *We now choose the nonlinear term $F(u) = u^3 - u$ and the exact solution $u(x,y,t) = t^2 sin(2\pi x) sin(2\pi y)$, then the source function can be determined by*

$$g(x,y,t) = \left[2t - t^2 + \frac{2t^{2-\alpha}}{\Gamma(3-\alpha)} + 16\pi^2\frac{2t^{2-\beta}}{\Gamma(3-\alpha)}\right]\sin(2\pi x)\sin(2\pi y) + t^6\sin^3(2\pi x)\sin^3(2\pi y). \tag{5.1}$$

Table 1 presents the errors for three iterative schemes: DLIS, PLIS, and NIS, with parameters $\alpha = 0.6, \beta = 0.3$, $d_1 = d_2 = 5$, $d_3 = 5$ is the parameter in basis functions of Eq (2.9), and $e = 10^{-8}$. The data show that under identical error conditions, the number of iterations required to reach this error is 4 for DLIS, 5 for PLIS, and 6 for NIS. With the same error for three iterative schemes, the fewer the number of iterations , the easier it is more easy to solve Eq (5.1).

The performance of NIS with $\alpha = 0.6$, $\beta = 0.3$, $d_1 = d_2 = 5$, and $d_3 = 5$ was evaluated using linear barycentric rational and Lagrange bases across barycentric rational uniform (RUN), barycentric Lagrange uniform (LUN), barycentric rational nonuniform "Chebyshev nodes" (RNN), and barycentric Lagrange nonuniform (LNN), as detailed in Table 2. This table reveals that Lagrange interpolation yields smaller errors and exhibits faster convergence compared to rational interpolation.

Table 3 provides the errors for DLIS when $\beta = 0.4$, $m = n = l = 12$, and $d_1 = d_2 = d_3 = 6$, for varying values of α from 0.01 to 0.99. The results indicate that the error remains consistent across different α values.

Next, we examine the convergence conditions of DLIS under varying spatial and temporal variables.

Tables 4 and 5, along with Tables 6 and 7, display the errors for DLIS under different configurations. Specifically, for a fixed spatial variable ($d_1 = d_2 = 5$), the convergence rates of the temporal variable (d_3) are shown in Tables 4 and 6 for $\alpha = \beta = 0.5$ using uniform (Lagrange nodes) and nonuniform (Chebyshev nodes). Conversely, for a fixed temporal variable ($d_3 = 5$), the convergence rates of the spatial variables ($d_1 = d_2$) are presented in Tables 5 and 7 under the same node types.

Finally, Table 2 also includes additional error data for DLIS, further supporting the analysis of its convergence properties.

Table 1. Errors of the $\alpha = 0.6; \beta = 0.3, d_1 = d_2 = 5; d_3 = 5$.

	NIS		PLIS		DLIS	
8	6.3230e-02	4	6.3230e-02	5	6.3230e-02	6
10	6.6465e-03	4	6.6465e-03	5	6.6465e-03	6
12	6.6822e-04	4	6.6822e-04	5	6.6822e-04	6
14	3.3039e-05	4	3.3040e-05	5	3.3009e-05	6

Table 2. Errors of the NIS with $\alpha = 0.6; \beta = 0.3, d_1 = d_2 = 5; d_3 = 5$.

	RUN		RNN		LUN		LNN	
8	6.3230e-02		1.9901e-02		6.3230e-02		1.9901e-02	
10	6.6465e-03	7.8305	5.8995e-04	12.231	5.9584e-03	8.2104	4.0966e-04	13.498
12	6.6822e-04	10.295	1.5017e-05	16.451	1.5017e-05	13.465	6.6391e-06	18.474
14	3.3039e-05	16.492	1.8534e-05	-	1.8534e-05	18.024	1.3223e-07	21.479

Table 3. Errors of the DLIS with $\beta = 0.4; m = n = l = 14; d_1 = d_2 = d_3 = 6$.

α	RUN	LUN	RNN	LNN
0.01	4.1531e-04	1.1062e-05	2.2889e-05	1.3272e-07
0.1	4.1507e-04	1.1057e-05	2.2879e-05	1.3269e-07
0.2	4.1479e-04	1.1049e-05	2.2867e-05	1.3265e-07
0.3	4.1448e-04	1.1040e-05	2.2855e-05	1.3261e-07
0.4	4.1417e-04	1.1031e-05	2.2841e-05	1.3256e-07
0.5	4.1384e-04	1.1022e-05	2.2827e-05	1.3252e-07
0.6	4.1349e-04	1.1013e-05	2.2813e-05	1.3247e-07
0.7	4.1314e-04	1.1003e-05	2.2798e-05	1.3242e-07
0.8	4.1277e-04	1.0993e-05	2.2783e-05	1.3237e-07
0.9	4.1241e-04	1.0983e-05	2.2767e-05	1.3232e-07
0.99	4.1208e-04	1.0975e-05	2.2753e-05	1.3228e-07

Table 4. Errors of the DLIS with $d_3 = 5$.

n $d_1 = d_2$	1		2		3		4	
6	3.8583e-02		1.0186e-01		2.4241e-02		6.3906e-02	
8	1.8612e-02	2.5340	3.6325e-02	3.5841	1.4750e-04	17.735	1.7484e-02	4.5054
10	9.7067e-03	2.9175	1.5612e-02	3.7844	1.3611e-03	-	5.0815e-03	5.5375
12	6.0672e-03	2.5774	8.4169e-03	3.3885	9.8761e-04	1.7594	1.8865e-03	5.4347

Table 5. Errors of the DLIS with $d_1 = d_2 = 5$.

n d_3	1		2		3		4	
6	1.9953e-01		6.4108e-02		6.4017e-02		6.3906e-02	
8	3.2383e-02	6.3206	6.6809e-03	7.8604	6.6788e-03	7.8566	6.6771e-03	7.8514
10	1.7529e-01	-	6.7089e-04	10.300	6.7105e-04	10.298	6.7099e-04	10.297
12	5.9232e-01	-	3.3135e-05	16.498	3.3186e-05	16.491	3.3189e-05	16.490

Table 6. Errors of the direct of $d_1 = d_2 = 5$.

n d_3	1		2		3		4	
6	4.1777e-02		2.0234e-02		2.0231e-02		2.0227e-02	
8	3.6512e-02	0.4683	5.9476e-04	12.260	5.9473e-04	12.260	5.9472e-04	12.259
10	2.3414e-02	1.9912	1.5193e-05	16.435	1.5193e-05	16.435	1.5193e-05	16.435
12	1.2552e-02	3.4193	1.8688e-05	-	1.8687e-05	-	1.8687e-05	-

Table 7. Errors of the direct of $d_3 = 5$.

n $d_1 = d_2$	1		2		3		4	
6	2.3371e-03		4.7455e-02		6.1448e-03		2.0227e-02	
8	1.1093e-02	-	2.5560e-02	2.1508	9.0325e-04	6.6649	2.7929e-03	6.8825
10	1.4693e-02	-	9.7582e-03	4.3153	9.7301e-04	-	9.2683e-04	4.9432
12	8.5018e-03	3.0006	4.8465e-03	3.8385	5.3405e-04	3.2904	3.5514e-04	5.2615

6. Concluding remarks

This study presents an efficient and accurate solution algorithm for the TFC equation based on the barycentric interpolation collocation method (BICM). Through rigorous theoretical analysis, the accuracy, convergence, and stability of the algorithm are systematically investigated, thereby establishing a solid theoretical foundation for its practical implementation. Numerical experiments are conducted using representative cases of the TFC equation, with results compared against those obtained from established numerical methods to validate the effectiveness and advantages of the proposed approach.

Despite these advancements, certain limitations persist. While the current formulation effectively handles simple Dirichlet and Neumann boundary conditions, it encounters challenges when applied to more complex conditions, such as Robin or nonlinear boundary conditions. These difficulties arise due to the increased complexity of the resulting algebraic system, which becomes harder to solve accurately. In real-world applications, boundary conditions may involve nonlinear dependencies on time or spatial variables, which existing numerical formulations struggle to represent precisely in algebraic form, thereby affecting both the accuracy and stability of the solutions.

Future research will focus on refining the algorithm through improved node selection strategies and the exploration of more suitable node distributions to enhance interpolation accuracy and computational efficiency. Hybridizing the BICM with other numerical methods, such as the finite element method or spectral methods, may offer complementary advantages for solving more complex problems. Expanding the application scope of this method to more sophisticated TFC equation models, particularly those involving multi-physics coupling, can address a broader range of engineering and scientific challenges.

In conclusion, this study has successfully developed and validated an effective algorithm for solving the TFC equation using the BICM. Future work will aim to overcome current limitations and extend the method's applicability to more complex and realistic scenarios.

Use of AI tools declaration

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

Acknowledgments

This work received the support of the Water Conservancy Informatization Project for Data Acquisition and Digital Twin System, which is applied to both the "Grade-III to Grade-II Upgrading" of the Navigation Channel (Liuchang Reach) of the Beijing-Hangzhou Grand Canal (Jining to Dongping Section) and the Flood Diversion to the South as Appropriate in the Old Lake Area of Dongping Lake.

Conflicts of interest

The authors declare that they have no conflicts of interest.

References

1. N. R. Bayramov, J. K. Kraus, On the stable solution of transient convection-diffusion equations, *J. Comput. Appl. Math.*, **280** (2015), 275–293. https://doi.org/10.1016/j.cam.2014.12.001
2. T. Liu, H. Zhang, X. Yang, The ADI compact difference scheme for three-dimensional integro-partial differential equation with three weakly singular kernels, *J. Appl. Math. Comput.*, **71** (2025), 3861–3889. https://doi.org/10.1007/s12190-025-02386-3
3. J. Zhang, X. Yang, S. Wang, The ADI difference and extrapolation scheme for high-dimensional variable coefficient evolution equations, *Electron. Res. Arch.*, **33** (2025), 3305–3327. https://doi.org/10.3934/era.2025146

4. Z. Zhang, X. Yang, S. Wang, The alternating direction implicit difference scheme and extrapolation method for a class of three-dimensional hyperbolic equations with constant coefficients, *Electron. Res. Arch.*, **33** (2025), 3348–3377. https://doi.org/10.3934/era.2025148
5. X. Yang, Z. Zhang, Superconvergence analysis of a robust orthogonal Gauss collocation method for 2D fourth-order subdiffusion equations, *J. Sci. Comput.*, **100** (2024), 62. https://doi.org/10.1007/s10915-024-02616-z
6. X. Yang, Z. Zhang, Analysis of a new NFV scheme preserving DMP for two-dimensional sub-diffusion equation on distorted meshes, *J. Sci. Comput.*, **99** (2024), 80. https://doi.org/10.1007/s10915-024-02511-7
7. Y. Shi, X. Yang, The pointwise error estimate of a new energy-preserving nonlinear difference method for supergeneralized viscous Burgers' equation, *Comput. Appl. Math.*, **44** (2025), 257. https://doi.org/10.1007/s40314-025-03222-x
8. J. Wang, X. Jiang, H. Zhang, A BDF3 and new nonlinear fourth-order difference scheme for the generalized viscous Burgers' equation, *Appl. Math. Lett.*, **151** (2024), 109002. https://doi.org/10.1016/j.aml.2024.109002
9. Y. Wang, Y. Liu, H. Li, J. Wang, Finite element method combined with second-order time discrete scheme for nonlinear fractional Cable equation, *Eur. Phys. J. Plus*, **131** (2016), 61. https://doi.org/10.1140/epjp/i2016-16061-3
10. F. Zeng, C. Li, A new Crank-Nicolson finite element method for the time-fractional subdiffusion equation, *Appl. Numer. Math.*, **121** (2017), 82–95. https://doi.org/10.1016/j.apnum.2017.06.011
11. Y. Liu, Y. Du, H. Li, J. Wang, A two-grid finite element approximation for a nonlinear time-fractional Cable equation, *Nonlinear Dyn.*, **85** (2016), 2535–2548. https://doi.org/10.1007/s11071-016-2843-9
12. J. Shen, T. Tang, L. Wang, *Spectral Methods: Algorithms, Analysis and Applications*, Springer, Heidelberg, Germany, 2011. https://doi.org/10.1007/978-3-540-71041-7
13. A. G. Atta, Two spectral Gegenbauer methods for solving linear and nonlinear time fractional Cable problems, *Int. J. Mod. Phys. C*, **35** (2024), 1–22. https://doi.org/10.1142/S0129183124500700
14. H. Liu, S. Lü, Galerkin spectral method for nonlinear time fractional Cable equation with smooth and nonsmooth solutions, *Appl. Math. Comput.*, **350** (2019), 32–47. https://doi.org/10.1016/j.amc.2018.12.072
15. X. Yang, X. Jiang, H. Zhang, A time-space spectral tau method for the time fractional Cable equation and its inverse problem, *Appl. Numer. Math.*, **130** (2018), 95–111. https://doi.org/10.1016/j.apnum.2018.03.016
16. H. Liu, S. Lü, T. Jiang, Analysis of Legendre pseudospectral approximations for nonlinear time fractional diffusion-wave equations, *Int. J. Comput. Math.*, **98** (2021), 1769–1791. https://doi.org/10.1080/00207160.2020.1846731
17. Y. Chen, X. Lin, M. Zhang, Y. Huang, Stability and convergence of L1-Galerkin spectral methods for the nonlinear time fractional Cable equation, *East Asian J. Appl. Math.*, **13** (2023), 22–46. https://doi.org/10.4208/eajam.020521.140522

18. S. Kumar, D. Baleanu, Numerical solution of two-dimensional time fractional Cable equation with Mittag-Leffler kernel, *Math. Methods Appl. Sci.*, **43** (2020), 8348–8362. https://doi.org/10.1002/mma.6491
19. M. A. Arefin, U. Sadiya, M. Inc, M. H. Uddin, Adequate soliton solutions to the space-time fractional telegraph equation and modified third-order KdV equation through a reliable technique, *Opt. Quantum Electron.*, **54** (2022), 309. https://doi.org/10.1007/s11082-022-03640-9
20. N. H. Sweilam, S. M. AL-Mekhlafi, A novel numerical method for solving the 2-D time fractional Cable equation, *Eur. Phys. J. Plus*, **134** (2019), 323. https://doi.org/10.1140/epjp/i2019-12730-y
21. Y. Liu, Z. Yu, H. Li, F. Liu, J. Wang, Time two-mesh algorithm combined with finite element method for time fractional water wave model, *Int. J. Heat Mass Transfer*, **120** (2018), 1132–1145. https://doi.org/10.1016/j.ijheatmasstransfer.2017.12.118
22. Z. Wang, On Caputo-Type Cable equation: Analysis and computation, *CMES-Comput. Model. Eng. Sci.*, **123** (2020), 353–376. https://doi.org/10.32604/cmes.2020.08776
23. H. R. Ghehsareh, M. S. Seidzadeh, S. K. Etesami, Numerical simulation of a generalized anomalous electro-diffusion process in nerve cells by a localized meshless approach in Pseudospectral mode, *Int. J. Numer. Modell. Electron. Networks Devices Fields*, **33** (2020), e2756. https://doi.org/10.1002/jnm.2756
24. W. Zou, Y. Tang, V. R. Hosseini, The numerical meshless approach for solving the 2D time nonlinear multi-term fractional Cable equation in complex geometries, *Fractals*, **30** (2022), 2240170. https://doi.org/10.1142/S0218348X22401703
25. Y. Liu, Y. Du, H. Li, F. Liu, Y. Wang, Some second-order θ schemes combined with finite element method for nonlinear fractional Cable equation, *Numer. Algorithms*, **80** (2019), 533–555. https://doi.org/10.1007/s11075-018-0496-0
26. J. P. Berrut, M. S. Floater, G. Klein, Convergence rates of derivatives of a family of barycentric rational interpolants, *Appl. Numer. Math.*, **61** (2011), 989–1000. https://doi.org/10.1016/j.apnum.2011.05.001
27. J. P. Berrut, S. A. Hosseini, G. Klein, The linear barycentric rational quadrature method for Volterra integral equations, *SIAM J. Sci. Comput.*, **36** (2014), A105–A123. https://doi.org/10.1137/120904020
28. J. P. Berrut, G. Klein, Recent advances in linear barycentric rational interpolation, *J. Comput. Appl. Math.*, **259** (2014), 95–107. https://doi.org/10.1016/j.cam.2013.03.044
29. E. Cirillo, K. Hormann, On the Lebesgue constant of barycentric rational Hermite interpolants at equidistant nodes, *J. Comput. Appl. Math.*, **349** (2019), 292–301. https://doi.org/10.1016/j.cam.2018.06.011
30. M. S. Floater, K. Hormann, Barycentric rational interpolation with no poles and high rates of approximation, *Numer. Math.*, **107** (2007), 315–331. https://doi.org/10.1007/s00211-007-0093-y
31. G. Klein, J. P. Berrut, Linear rational finite differences from derivatives of barycentric rational interpolants, *SIAM J. Numer. Anal.*, **50** (2012), 643–656. https://doi.org/10.1137/110827156
32. G. Klein, J. P. Berrut, Linear barycentric rational quadrature, *BIT Numer. Math.*, **52** (2012), 407–424. https://doi.org/10.1007/s10543-011-0357-x

33. A. Abdi, J. P. Berrut, S. A. Hosseini, The linear barycentric rational method for a class of delay Volterra integro-differential equations, *J. Sci. Comput.*, **75** (2018), 1757–1775. https://doi.org/10.1007/s10915-017-0608-3

34. J. Pour-Mahmoud, M. Y. Rahimi-Ardabili, S. Shahmorad, Numerical solution of the system of Fredholm integro-differential equations by the tau method, *Appl. Math. Comput.*, **168** (2005), 465–478. https://doi.org/10.1016/j.amc.2004.09.026

35. S. M. Hosseini, S. Shahmorad, Numerical solution of a class of integro-differential equations by the tau method with an error estimation, *Appl. Math. Comput.*, **136** (2003), 559–570. https://doi.org/10.1016/S0096-3003(02)00081-4

36. S. Yousefi, M. Razzaghi, Legendre wavelets method for the nonlinear Volterra Fredholm integral equations, *Math. Comput. Simul.*, **70** (2005), 1–8. https://doi.org/10.1016/j.matcom.2005.02.035

37. S. Yalcinbas, M. Sezer, H. H. Sorkun, Legendre polynomial solutions of high-order linear Fredholm integro-differential equations, *Appl. Math. Comput.*, **210** (2009), 334–349. https://doi.org/10.1016/j.amc.2008.12.090

38. S. Li, Z. Wang, *High Precision Meshless barycentric Interpolation Collocation Method–Algorithmic Program and Engineering Application (in Chinese)*, Science Press, Beijing, China, 2012.

39. Z. Wang, S. Li, *Barycentric Interpolation Collocation Method for Nonlinear Problems (in Chinese)*, National Defense Industry Press, 2015.

40. Z. Wang, Z. Xu, J. Li, Mixed barycentric interpolation collocation method of displacement-pressure for incompressible plane elastic problems (in Chinese), *Chin. J. Appl. Mech.*, **35** (2018), 631–636+695.

41. Z. Wang, L. Zhang, Z. Xu, J. Li, Barycentric interpolation collocation method based on mixed displacement-stress formulation for solving plane elastic problems (in Chinese), *Chin. J. Appl. Mech.*, **35** (2018), 304–308+451.

42. K. Maleknejad, N. Aghazadeh, Numerical solutions of Volterra integral equations of the second kind with convolution kernel by using Taylor-series expansion method, *Appl. Math. Comput.*, **161** (2005), 915–922. https://doi.org/10.1016/j.amc.2003.12.075

43. J. Li, J. Qu, High precision barycentric interpolation collocation method to solve EFK equation, *Int. J. Comput. Math.*, **2025** (2025), 1–19. https://doi.org/10.1080/00207160.2025.2522993

44. Z. Li, J. Li, Numerical solutions for (2+1)-dimensional ZK-MEW equation using linear barycentric rational collocation method, *Comput. Math. Appl.*, **193** (2025), 332–345. https://doi.org/10.1016/j.camwa.2025.06.021

45. C. Li, S. Hon, Multilevel tau preconditioners for symmetrized multilevel Toeplitz systems with applications to solving space fractional diffusion equations, *SIAM J. Matrix Anal. Appl.*, **46** (2025), 487–508. https://doi.org/10.1137/24M1647096

46. J. Pan, M. Ng, H. Wang, Fast preconditioned iterative methods for finite volume discretization of steady-state space-fractional diffusion equations, *Numer. Algorithms*, **74** (2017), 153–173. https://doi.org/10.1007/s11075-016-0143-6

47. M. Mazza, S. Serra-Capizzano, R. L. Sormani, Algebra preconditionings for 2D Riesz distributed-order space-fractional diffusion equations on convex domains, *Numer. Linear Algebra Appl.*, **31** (2024), e2536. https://doi.org/10.1002/nla.2536

48. M. Donatelli, M. Mazza, S. Serra-Capizzano, Spectral analysis and multigrid methods for finite volume approximations of space-fractional diffusion equations, *SIAM J. Sci. Comput.*, **40** (2018), A4007–A4039. https://doi.org/10.1137/17M115164X

49. Z. Chen, H. Zhang, H. Chen, ADI compact difference scheme for the two-dimensional integro-differential equation with two fractional Riemann Liouville integral kernels, *Fractal Fract.*, **8** (2024), 707. https://doi.org/10.3390/fractalfract8120707