Electronic
Research Archive

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

ERA, 34(9): 6825–6845.
DOI: 10.3934/era.2026297
Received: 23 June 2026
Revised: 23 July 2026
Accepted: 05 August 2026
Published: 12 August 2026

Research article

Multi-term normalized time-fractional diffusion equations

Yongho Choi[1], **Seungjae Lee**[2] **and Junseok Kim**[2,*]

[1] Department of Computer & Information Engineering, Daegu University, Gyeongsan-si, Gyeongsangbuk-do 38453, Republic of Korea

[2] Department of Mathematics, Korea University, Seoul 02841, Republic of Korea

* **Correspondence:** Email: cfdkim@korea.ac.kr.

Abstract: This paper proposes a new multi-term normalized time-fractional diffusion equation. The proposed model extends the normalized time-fractional derivative to include several fractional orders. It keeps the normalization property and uses weighting coefficients to control short- and long-term memory effects. An implicit finite difference method is developed, and the tridiagonal linear system is solved by the Thomas algorithm. The proposed algorithm is unconditionally stable and converges with a temporal accuracy of $2 - \alpha_{\max}$ and second-order spatial accuracy. Numerical convergence tests agree with the theoretical results. Numerical examples show that the fractional orders and weighting coefficients control the diffusion rate, memory strength, and frequency-dependent decrease. Additionally, the computational results show that the proposed model is more flexible than the single-term normalized equation for describing anomalous diffusion with multiple memory scales. Finally, a simple two-dimensional extension of the proposed model is presented. MATLAB codes for both the single-term and multi-term normalized time-fractional diffusion models are provided in the Appendix.

Keywords: multi-term normalized time-fractional; numerical method; Caputo derivative

1. Introduction

Single-term time-fractional diffusion equations provide a fundamental model to describe anomalous transport phenomena through a single memory kernel [1], thereby effectively capturing subdiffusive behaviors in complex media. However, many real-world systems exhibit multiple memory mechanisms that cannot be represented by a single fractional order. Consequently, multi-term time-fractional diffusion equations have become an active and promising research area because they allow a more faithful mathematical representation of processes that involve a superposition of memory effects. Additionally, these models enable rigorous theoretical analyses, efficient numerical computations, and inverse problem formulations. The diffusion model with multi-term time-fractional derivatives was analyzed by Hu et al. [2], who proved the positivity of its solution using the subordination principle. Additionally, they also examined an inverse problem for finding a time-dependent source term and showed through numerical experiments that the proposed algorithm is reliable and efficient. Derbissaly and Sadybekov [3] investigated an inverse problem for the diffusion model; the authors determined a source term that does not depend on the spatial coordinate or the temperature field. Dwivedi and Rajeev [4] developed a fast numerical method for a multi-term time-fractional advection-diffusion equation; the proposed algorithm, based on the summation of exponentials technique for efficient computation of the Caputo derivative, was proven to be unconditionally stable and convergent while significantly reducing the CPU time and storage compared to conventional methods. Mostajeran and Hosseini [5] developed a high-order finite difference method (FDM) for multi-term time-fractional diffusion equations and used the L1-2-3 formula to approximate temporal derivatives with a higher accuracy; they used neural network-based automatic differentiation for spatial derivatives and demonstrated the method's unconditional stability, convergence, accuracy, and robustness. Liu and Yamamoto [6] showed that the fractional orders and parameters in time-fractional evolution models are uniquely determined from the principal terms of the asymptotic expansions near ($t = 0$) observed at a single spatial point. Liu et al. [7] presented a computational algorithm that combines the quadratic spline collocation method with the $L1^{+}$ formula for multi-term variable-order time-fractional mobile-immobile diffusion models and proved its unconditional stability and convergence. Hou and Hu [8] presented an FDM that combines the L1 method, the central difference method, and a backward formula for the two-dimensional (2D) nonlinear multi-term time-fractional subdiffusion equation with weakly singular solutions, and analyzed its stability and error estimates. Zhang et al. [9] studied the anti-periodic boundary value problem for multi-term fractional differential models with the p-Laplacian operator and proved the existence of solutions using the Leray-Schauder nonlinear alternative and Krasnosel'skii's fixed point theorem. Xiong and Li [10] studied an inverse problem for a multi-term time-fractional diffusion model. They proved the uniqueness of the inverse problem and proposed a trust-region algorithm for parameter reconstruction. The authors [11, 12] proposed an improved shifted Jacobi collocation method for the nonlinear time-fractional FitzHugh-Nagumo model and developed an ultraspherical collocation scheme based on Chebyshev-Gauss-Lobatto nodes for the time-fractional FitzHugh-Nagumo equation. Alsuyuti et al. [13] presented a shifted Jacobi Galerkin spectral method for one- and 2D multi-term time-fractional diffusion and diffusion-wave models; they demonstrated the effectiveness and accuracy of the proposed algorithm through numerical tests. Hafez et al. [14] constructed fractional and exponential Jacobi spectral Galerkin and collocation methods for multidimensional time-fractional diffusion models on semi-infinite domains. Sun and Chang [15] proposed a Galerkin spectral method for multi-term time-fractional diffusion models; they verified its stability, convergence, and effectiveness for inverse source problems. Kamel et al. [16] proposed an explicit spectral tau scheme based on Delannoy polynomials for the time-fractional diffusion model. Youssri et al. [17] applied a shifted first-kind Chebyshev polynomial collocation scheme to the time-fractional Korteweg-de Vries-Burgers model. Fan [18] proposed a fast numerical algorithm for multi-term Caputo-Fabrizio time-fractional diffusion models; the proposed method provided unconditional stability and second-order temporal accuracy. Furthermore, many studies have investigated diffusion-type partial differential equations with multi-term time-fractional derivatives. Santra [19] presented a higher-order hybrid computational method for multi-term time-fractional integro-partial differential models. Additionally, the author also analyzed the stability, convergence, and higher-order accuracy of the proposed scheme for multi-term time-fractional diffusion models with time singularities. Sabir et al. [20] presented a numerical algorithm for multi-term time-fractional advection-diffusion equations. The proposed method provided high-order accuracy through stability and convergence analyses. Mehri et al. [21] developed a finite element method for 2D multi-term nonlinear time-fractional convection-diffusion equations with the Caputo-Fabrizio derivative. The proposed scheme provided second-order accuracy in both space and time. Zhang et al. [22] proposed a barycentric rational interpolation collocation method for multi-term time-fractional convection-diffusion models. The proposed scheme provided high accuracy with only a few temporal partitioning points.

These developments confirmed that normalized single-term and multi-term time-fractional diffusion models together form a unified method that balances physical interpretability, analytical tractability, and computational feasibility. Since the multi-term formulation includes the single-term model as a special case, it provides a more general and flexible representation of anomalous diffusion processes. These features make such models highly relevant for the analysis of diffusion phenomena in heterogeneous materials, biological systems, and inverse identification applications.

Now, we propose a multi-term normalized time-fractional diffusion equation:

$$\begin{aligned} \frac{\partial^{\boldsymbol{\alpha},\boldsymbol{\beta}} u(x,t)}{\partial t^{\boldsymbol{\alpha},\boldsymbol{\beta}}} &= \frac{\partial^2 u(x,t)}{\partial x^2} \quad \text{for } (x,t) \in \Omega \times (0,\ \infty), && (1.1)\\ u(x,0) &= u_0(x), \quad x \in \Omega, && (1.2)\\ u(0,t) &= u(1,t) = 0, \quad t \geq 0, && (1.3) \end{aligned}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_N)$ with $0 \leq \alpha_k < 1$ for $k = 1, 2, \ldots, N$, $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_N)$ with $0 < \beta_k < 1$ and $\sum_{k=1}^{N} \beta_k = 1$, and $\Omega = (L_x, R_x)$ denotes a general one-dimensional interval. $u(x,t)$ is a space- and time-dependent variable, and $u_0(x)$ is the initial condition and must satisfy the boundary condition (1.3). The multi-term normalized time-fractional derivative is defined as follows:

$$\begin{aligned} \frac{\partial^{\boldsymbol{\alpha},\boldsymbol{\beta}} u(x,t)}{\partial t^{\boldsymbol{\alpha},\boldsymbol{\beta}}} &= \sum_{k=1}^{N} \beta_k \int_0^t \frac{1-\alpha_k}{t^{1-\alpha_k}(t-s)^{\alpha_k}} \frac{\partial u(x,s)}{\partial s} ds \\ &= \sum_{k=1}^{N} \beta_k \int_0^t w_{\alpha_k}^t(s) \frac{\partial u(x,s)}{\partial s} ds = \int_0^t \left[\sum_{k=1}^{N} \beta_k w_{\alpha_k}^t(s) \right] \frac{\partial u(x,s)}{\partial s} ds \\ &= \int_0^t w_{\boldsymbol{\alpha},\boldsymbol{\beta}}^t(s) \frac{\partial u(x,s)}{\partial s} ds, \end{aligned} \tag{1.4}$$

where the normalized weighting functions and their weighted sum are given by

$$w_{\alpha_k}^t(s) = \frac{1-\alpha_k}{t^{1-\alpha_k}(t-s)^{\alpha_k}} \quad \text{and} \quad w_{\boldsymbol{\alpha},\boldsymbol{\beta}}^t(s) = \sum_{k=1}^{N} \beta_k w_{\alpha_k}^t(s), \quad \text{respectively.}$$

Equation (1.4) has the same integral structure as the standard Caputo fractional derivative. The only difference is that the weighting function is the normalized formulation. We note that the normalized weighting function $w_{\alpha_k}^t(s)$ satisfies the following for all $0 \leq \alpha_k < 1$ and $t > 0$:

$$\int_0^t w_{\alpha_k}^t(s)\,ds = \int_0^t \frac{1-\alpha_k}{t^{1-\alpha_k}(t-s)^{\alpha_k}}\,ds = \frac{1-\alpha_k}{t^{1-\alpha_k}}\int_0^t (t-s)^{-\alpha_k}\,ds = \frac{1-\alpha_k}{t^{1-\alpha_k}}\left[-\frac{(t-s)^{1-\alpha_k}}{1-\alpha_k}\right]_0^t$$
$$= -\frac{1}{t^{1-\alpha_k}}\left[(t-s)^{1-\alpha_k}\right]_0^t = -\frac{1}{t^{1-\alpha_k}}\left(0^{1-\alpha_k}-t^{1-\alpha_k}\right) = 1, \tag{1.5}$$

which implies for any $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $t > 0$,

$$\int_0^t w_{\boldsymbol{\alpha},\boldsymbol{\beta}}^t(s)ds = 1. \tag{1.6}$$

Moreover, because β_k and $0 \le \alpha_k < 1$, the combined weighting function is nondecreasing in s for $0 \le s < t$:

$$\frac{dw_{\boldsymbol{\alpha},\boldsymbol{\beta}}^t(s)}{ds} = \frac{d}{ds}\left(\sum_{k=1}^N \beta_k \frac{1-\alpha_k}{t^{1-\alpha_k}(t-s)^{\alpha_k}}\right) = \sum_{k=1}^N \beta_k \frac{\alpha_k(1-\alpha_k)}{t^{1-\alpha_k}(t-s)^{\alpha_k+1}} \ge 0.$$

Thus, the operator $\partial^{\boldsymbol{\alpha},\boldsymbol{\beta}}/\partial t^{\boldsymbol{\alpha},\boldsymbol{\beta}}$ represents a multi-term normalized fractional derivative that preserves the integral weight over $[0, t]$. Additionally, we note that when $N = 1$, the proposed definition reduces to the single-term normalized time-fractional derivative [23].

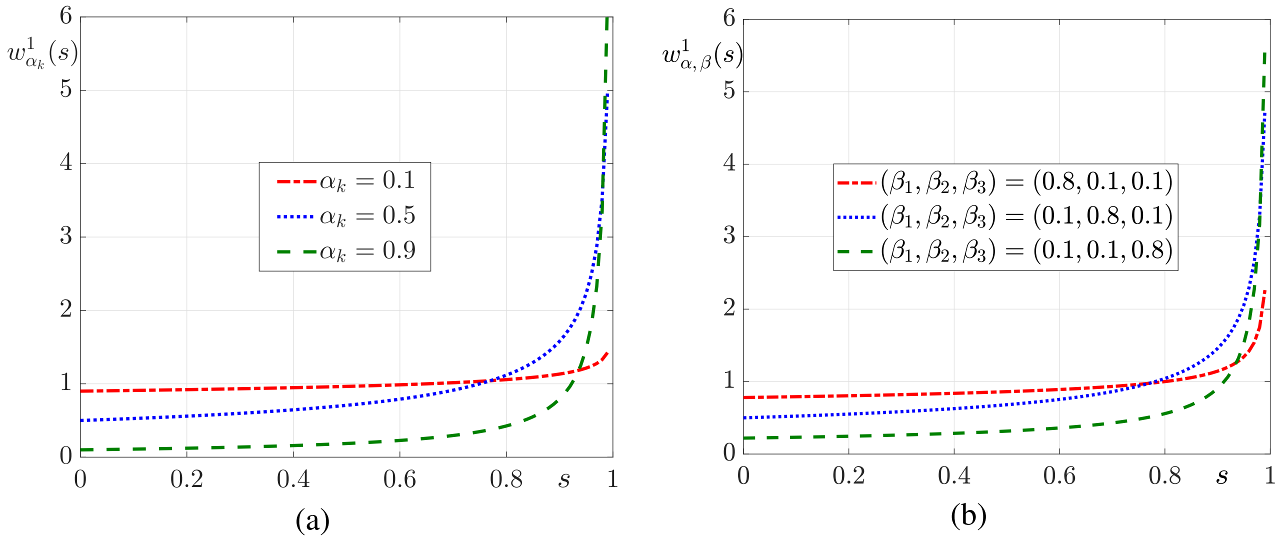


Figure 1. (a) Single-term weighting functions $w^1_{\alpha_k}(s)$ for $\alpha_k = 0.1$, 0.5, and 0.9. (b) Multi-term weighting functions $w^1_{\boldsymbol{\alpha},\boldsymbol{\beta}}(s)$ for three sets of parameters $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3) = (0.8, 0.1, 0.1)$, $(0.1, 0.8, 0.1)$, and $(0.1, 0.1, 0.8)$. The common parameter $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3) = (0.1, 0.5, 0.9)$ is used for all cases.

Figure 1 displays the behavior of the normalized weighting functions that define the multi-term normalized time-fractional derivative. In Figure 1(a), the single-term functions $w^1_{\alpha_k}(s)$ are shown for three fractional orders: $\alpha_k = 0.1$, 0.5, and 0.9. As α_k increases, the functions are more sharply concentrated near $s = 1$. Larger fractional orders assign greater weight to recent temporal history, whereas smaller α_k values distribute the memory effect more evenly over the interval $[0, 1]$.

Figure 1(b) displays the composite weighting functions $w^1_{\boldsymbol{\alpha},\boldsymbol{\beta}}(s)$ for three different sets of weighting coefficients $\boldsymbol{\beta}$, with $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3) = (0.1, 0.5, 0.9)$. These curves show that variations in $\boldsymbol{\beta}$ modify the contribution of each fractional order and control the balance between short- and long-term memory effects. The behavior shown in Figure 1 is also consistent with the complete monotonicity of the memory kernels. Although the weighting functions increase with s as s approaches the current time, they are positive and completely monotone with respect to the variable $t - s$. Because the coefficients β_k are positive, the multi-term kernel preserves this property. Note that if $N = 1$, Eq (1.4) reduces to the single-term normalized time-fractional derivative, which has been previously analyzed in [24, 25]. In this special case, the weighting coefficient becomes unity, the operator simplifies to represent a single memory kernel, and it recovers the standard normalized fractional dynamics without the interaction among multiple fractional orders.

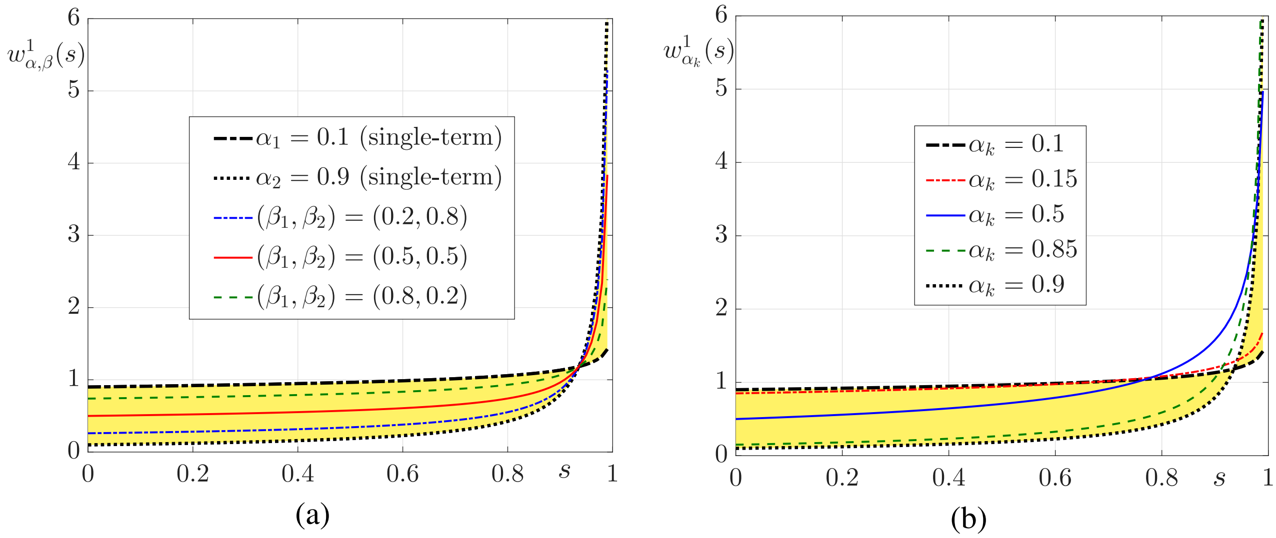


Figure 2. (a) Single-term weighting functions $w^1_{\alpha_k}(s)$ for $\alpha_1 = 0.1$ and $\alpha_2 = 0.9$, and multi-term weighting functions $w^1_{\boldsymbol{\alpha},\boldsymbol{\beta}}(s)$ for $\boldsymbol{\beta} = (\beta_1, \beta_2) = (0.2, 0.8)$, $(0.5, 0.5)$, and $(0.8, 0.2)$ with $\boldsymbol{\alpha} = (\alpha_1, \alpha_2) = (0.1, 0.9)$. The shaded area denotes the range of multi-term weighting functions for various values of $\boldsymbol{\beta}$. (b) Single-term weighting functions $w^1_{\alpha_k}(s)$ for different fractional orders $\alpha_k = 0.1$, 0.15, 0.5, 0.85, and 0.9. The shaded region is identical to that in (a).

Figure 2 compares the weighting functions of the single-term and multi-term models. In Figure 2(a), the shaded region is between the single-term weighting functions for $\alpha_1 = 0.1$ and $\alpha_2 = 0.9$. For the fixed fractional orders $\boldsymbol{\alpha} = (0.1, 0.9)$, the multi-term weighting functions for three different weighting coefficients all stay inside the shaded region. As the weighting coefficients change, the weighting function changes from one boundary to the other. This result shows that the memory effect can be controlled by the weighting coefficients. Figure 2(b) displays the single-term weighting functions for $\alpha_k = 0.1, 0.15, 0.5, 0.85$, and 0.9. Most curves inside the shaded region are different from these single-term weighting functions. This result shows that a single fractional order cannot produce most of the weighting functions inside the shaded region. Therefore, the multi-term model gives more choices for the weighting function and can describe memory effects that the single-term model cannot describe.

In this study, we propose a novel multi-term normalized time-fractional diffusion equation, in which the normalized time-fractional derivative is extended to a multi-term formulation. This extension provides a more flexible method to model diffusion processes with multiple memory scales than the conventional single-term normalized fractional model. Additionally, we develop an efficient FDM, prove its stability and convergence properties, and perform computational experiments to demonstrate the capability of the proposed model to describe anomalous diffusion with flexible short- and long-term memory effects. For further details on advanced numerical methods for time-fractional diffusion equations, including stability and convergence analyses [26–28], inverse problems [29], unconditionally stable schemes [30, 31], high-order schemes [32, 33], and efficient computational techniques [34, 35], see the relevant references.

The remainder of this paper is organized as follows: in Section 2, an FDM is presented based on the multi-term normalized fractional operator, and the linear system is efficiently solved using the Thomas algorithm; Section 3 presents several numerical experiments designed to study the effects of the fractional orders and weighting coefficients on the diffusion dynamics, including the analysis of frequency-dependent memory behavior; and Section 4 presents a summary of the results and discusses possible extensions of the proposed method to higher-dimensional and variable-order fractional models.

2. Computational method

Let $\Omega = (L_x,\ R_x)$ be the domain and $\Omega_h = \{x_i | x_i = L_x + (i-1)h,\ i = 1, \dots, N_x\}$ be its discrete domain, where $h = (R_x - L_x)/(N_x - 1)$ for an integer N_x, see Figure 3.

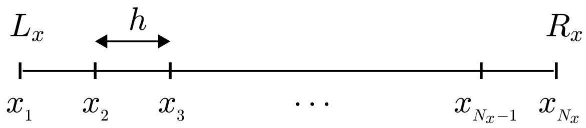


Figure 3. Discrete computational domain $\Omega_h = \{x_i \,|\, x_i = L_x + (i-1)h,\ i = 1, \dots, N_x\}$, where $h = (R_x - L_x)/(N_x - 1)$ is the uniform spatial step size. The points $x_1, x_2, \dots, x_{N_x}$ uniformly divide the domain (L_x, R_x) into $N_x - 1$ subintervals, providing the spatial grid for the FDM of the proposed model.

Let $u_i^n = u(x_i,\ t_n)$, where $t_n = (n-1)\Delta t$. The discrete form of Eq (1.4) is obtained through the following numerical quadrature scheme:

$$\begin{aligned}
\frac{\partial^{\alpha,\beta} u(x_i, t_{n+1})}{\partial t^{\alpha,\beta}} &= \int_0^t w_{\alpha,\beta}^t(s) \frac{\partial u(x_i, s)}{\partial s} ds = \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} w_{\alpha,\beta}^{t_{n+1}}(s) \frac{\partial u(x, s)}{\partial s} ds \\
&\approx \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} w_{\alpha,\beta}^{t_{n+1}}(s) \frac{u_i^{p+1} - u_i^p}{\Delta t} ds = \sum_{p=1}^{n} \left(\int_{t_p}^{t_{p+1}} w_{\alpha,\beta}^{t_{n+1}}(s) ds \right) \frac{u_i^{p+1} - u_i^p}{\Delta t} \\
&= \sum_{p=1}^{n} \left(\sum_{k=1}^{N} \beta_k \frac{(n+1-p)^{1-\alpha_k} - (n-p)^{1-\alpha_k}}{n^{1-\alpha_k}} \right) \frac{u_i^{p+1} - u_i^p}{\Delta t}. \qquad (2.1)
\end{aligned}$$

Then, we have

$$\sum_{p=1}^{n} w_p^n \frac{u_i^{p+1} - u_i^p}{\Delta t} = \frac{u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1}}{h^2}, \tag{2.2}$$

where

$$w_p^n = \sum_{k=1}^{N} \beta_k \frac{(n+1-p)^{1-\alpha_k} - (n-p)^{1-\alpha_k}}{n^{1-\alpha_k}}. \tag{2.3}$$

For any integer n, the weight w_p^n must obey the following condition:

$$\begin{aligned}
\sum_{p=1}^{n} w_p^n &= \sum_{p=1}^{n} \left(\sum_{k=1}^{N} \beta_k \frac{(n+1-p)^{1-\alpha_k} - (n-p)^{1-\alpha_k}}{n^{1-\alpha_k}} \right) \qquad (2.4)\\
&= \sum_{k=1}^{N} \beta_k \left[\sum_{p=1}^{n} \left(\frac{(n+1-p)^{1-\alpha_k} - (n-p)^{1-\alpha_k}}{n^{1-\alpha_k}} \right) \right]\\
&= \sum_{k=1}^{N} \beta_k \left[\frac{1}{n^{1-\alpha_k}} \sum_{p=1}^{n} \left((n+1-p)^{1-\alpha_k} - (n-p)^{1-\alpha_k} \right) \right]\\
&= \sum_{k=1}^{N} \beta_k \Bigg[\frac{1}{n^{1-\alpha_k}} \left(n^{1-\alpha_k} - (n-1)^{1-\alpha_k} \right) + \left((n-1)^{1-\alpha_k} - (n-2)^{1-\alpha_k} \right) + \cdots\\
&\quad + \left(2^{1-\alpha_k} - 1^{1-\alpha_k} \right) + \left(1^{1-\alpha_k} - 0^{1-\alpha_k} \right) \Bigg] = \sum_{k=1}^{N} \beta_k = 1. \qquad (2.5)
\end{aligned}$$

Hence, Eq (2.2) can be expressed as follows:

$$-\frac{1}{h^2} u_{i-1}^{n+1} + \left(\frac{w_n^n}{\Delta t} + \frac{2}{h^2} \right) u_i^{n+1} - \frac{1}{h^2} u_{i+1}^{n+1} = \frac{w_n^n}{\Delta t} u_i^n - \sum_{p=1}^{n-1} w_p^n \frac{u_i^{p+1} - u_i^p}{\Delta t}. \tag{2.6}$$

Equation (2.6) represents a tridiagonal linear system subject to homogeneous Dirichlet boundary conditions. At a fixed spatial grid point x_i, the direct evaluation of the summation term in Eq (2.6) requires the storage of $u_i^1, u_i^2, \ldots, u_i^n$, and hence, the memory storage cost is $O(n)$. For the entire computational domain with N_x spatial grid points, the total memory storage cost is $O(nN_x)$. This system can be efficiently solved through the Thomas algorithm [36], which provides a direct and computationally stable procedure for tridiagonal matrices. Accordingly, the unknown vector $\mathbf{u}^{n+1}$ is obtained by applying the Thomas algorithm:

$$A\mathbf{u}^{n+1} = \mathbf{f},$$

where

$$A = \begin{pmatrix} \frac{w_n^n}{\Delta t} + \frac{2}{h^2} & -\frac{1}{h^2} & 0 & \cdots & 0 & 0 & 0 \\ -\frac{1}{h^2} & \frac{w_n^n}{\Delta t} + \frac{2}{h^2} & -\frac{1}{h^2} & \cdots & 0 & 0 & 0 \\ 0 & -\frac{1}{h^2} & \frac{w_n^n}{\Delta t} + \frac{2}{h^2} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\frac{1}{h^2} & \frac{w_n^n}{\Delta t} + \frac{2}{h^2} & -\frac{1}{h^2} \\ 0 & 0 & 0 & \cdots & 0 & -\frac{1}{h^2} & \frac{w_n^n}{\Delta t} + \frac{2}{h^2} \end{pmatrix},$$

$$\mathbf{u}^{n+1} = \begin{pmatrix} u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_{N_x-1}^{n+1} \end{pmatrix} \text{ and } \mathbf{f} = \begin{pmatrix} w_n^n u_2^n/\Delta t - F \\ w_n^n u_3^n/\Delta t - F \\ \vdots \\ w_n^n u_{N_x-1}^n/\Delta t - F \end{pmatrix},$$

where $F = \sum_{p=1}^{n-1} w_p^n(u_i^{p+1} - u_i^p)/\Delta t$. We note that the matrix A is strictly diagonally dominant, and therefore, it is invertible and guarantees the stability of the proposed scheme.

Next, we briefly present the stability and convergence properties of the proposed numerical method. The complete proofs of the stability and convergence results for a numerical scheme based on a similar idea are available in the recently published paper [37]. Let $\alpha_{\max} = \max_{1\le k\le N} \alpha_k$. The multi-term discrete weight is given by the following:

$$w_p^n = \sum_{k=1}^{N} \beta_k \frac{(n+1-p)^{1-\alpha_k} - (n-p)^{1-\alpha_k}}{n^{1-\alpha_k}}, \qquad 1 \le p \le n.$$

Because $\beta_k > 0$ and $\sum_{k=1}^{N} \beta_k = 1$, the weights satisfy the following:

$$0 < w_1^n < w_2^n < \cdots < w_n^n, \qquad \sum_{p=1}^{n} w_p^n = 1.$$

Define the history coefficients by

$$C_1^n = w_1^n, \qquad C_p^n = w_p^n - w_{p-1}^n, \quad 2 \le p \le n.$$

Then,

$$C_p^n > 0, \qquad \sum_{p=1}^{n} C_p^n = w_n^n,$$

and the right-hand side of the numerical scheme, after multiplying Δt, can be written as

$$w_n^n u_i^n - \sum_{p=1}^{n-1} w_p^n \left(u_i^{p+1} - u_i^p\right) = \sum_{p=1}^{n} C_p^n u_i^p.$$

Theorem 1 (Unconditional stability)**.** *Let $\{u_i^n\}$ be the numerical solution obtained from the implicit FDM. Then, for arbitrary $\Delta t > 0$ and $h > 0$,*

$$\|u^{n+1}\|_\infty \le \|u^1\|_\infty, \qquad n \ge 1.$$

Therefore, the proposed numerical scheme is unconditionally stable.

Proof. Let $\mu = \Delta t/h^2$. The numerical scheme can be written as follows:

$$(w_n^n + 2\mu)\, u_i^{n+1} = \mu u_{i-1}^{n+1} + \mu u_{i+1}^{n+1} + \sum_{p=1}^{n} C_p^n u_i^p.$$

Suppose that $|u_m^{n+1}| = \|u^{n+1}\|_\infty$ at an interior grid point x_m. Because $\mu > 0$ and $C_p^n > 0$, we have

$$w_n^n \|u^{n+1}\|_\infty \le \sum_{p=1}^{n} C_p^n \|u^p\|_\infty.$$

Using $\sum_{p=1}^{n} C_p^n = w_n^n$ and mathematical induction gives the following:

$$\|u^{n+1}\|_\infty \le \|u^1\|_\infty.$$

The estimate does not impose any restriction between Δt and h, which completes the proof. □

Assume that the exact solution has sufficient temporal and spatial regularity. The truncation error of the kth normalized fractional term satisfies the following:

$$R_{\Delta t,k}^{n+1} = O\left(\Delta t^{2-\alpha_k}\right).$$

Therefore, the truncation error of the multi-term operator satisfies the following:

$$\sum_{k=1}^{N} \beta_k R_{\Delta t,k}^{n+1} = O\left(\sum_{k=1}^{N} \beta_k \Delta t^{2-\alpha_k}\right) = O\left(\Delta t^{2-\alpha_{\max}}\right).$$

Together with the second-order central difference approximation in space, the total local truncation error satisfies the following:

$$\|R^{n+1}\|_\infty \le C\left(\Delta t^{2-\alpha_{\max}} + h^2\right),$$

where $C > 0$ is independent of Δt and h.

Theorem 2 (Error bound and convergence)**.** *Let $e_i^n = U_i^n - u_i^n$, where $U_i^n = u(x_i, t_n)$ is the exact solution and u_i^n is the numerical solution. Under sufficient regularity assumptions, there exists a constant $C_T > 0$, independent of Δt and h, such that*

$$\|e^n\|_\infty \le C_T\left(\Delta t^{2-\alpha_{\max}} + h^2\right), \qquad 0 \le t_n \le T.$$

Consequently, the proposed scheme is convergent with a temporal order of $2 - \alpha_{\max}$ and second-order spatial accuracy.

Proof. Subtracting the numerical method from the corresponding exact difference equation gives the following:

$$-\mu e_{i-1}^{n+1} + \left(w_n^n + 2\mu\right) e_i^{n+1} - \mu e_{i+1}^{n+1} = \sum_{p=1}^{n} C_p^n e_i^p + \Delta t R_i^{n+1}.$$

Applying the same maximum-norm argument as in the stability proof yields the following:

$$w_n^n \|e^{n+1}\|_\infty \leq \sum_{p=1}^{n} C_p^n \|e^p\|_\infty + \Delta t \|R^{n+1}\|_\infty.$$

The positivity and telescoping properties of the history coefficients, together with mathematical induction, imply

$$\|e^n\|_\infty \leq C_T \max_{1\leq j\leq n} \|R^j\|_\infty.$$

Using the truncation error estimate gives the following:

$$\|e^n\|_\infty \leq C_T \left(\Delta t^{2-\alpha_{\max}} + h^2\right).$$

This completes the proof. □

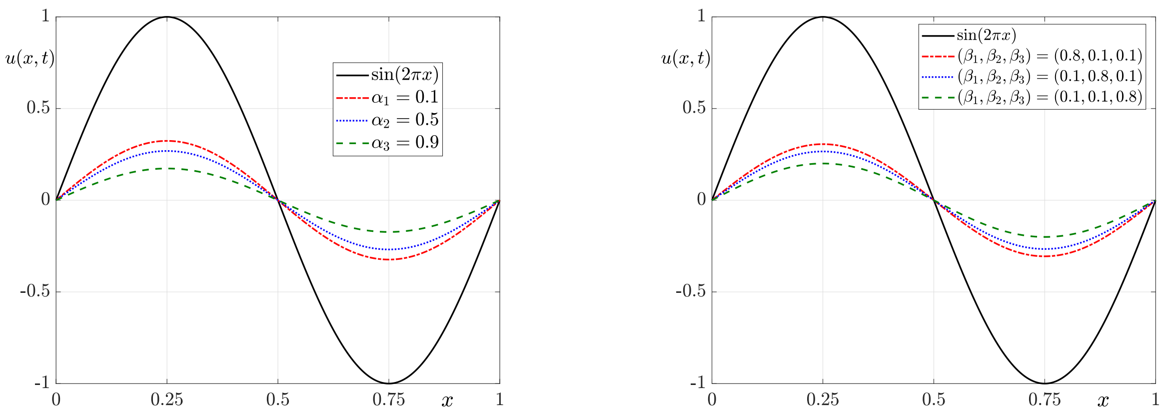
The above proofs only outlined the main ideas. For complete proofs and a more detailed theoretical analysis of a numerical scheme based on a similar idea, readers are referred to the recently published paper [37].

3. Computational tests

Now, a series of numerical tests are performed over the computational domain $\Omega \times (0, T)$, where $\Omega = (0, 1)$ and the terminal time T, to analyze the impact of the fractional orders $\boldsymbol{\alpha}$ and the weighting coefficients $\boldsymbol{\beta}$ on the dynamic behavior of the multi-term normalized time-fractional diffusion model. As an initial example, we use the following initial condition:

$$u(x, 0) = \sin(2\pi x), \tag{3.1}$$

which satisfies the zero Dirichlet boundary condition, because $u(0, 0) = \sin(2\pi \cdot 0) = 0$ and $u(1, 0) = \sin(2\pi \cdot 1) = 0$. Figure 4 presents the numerical results of the single-term and multi-term normalized time-fractional diffusion models at time $T = 0.05$. In Figure 4(a), the single-term solutions for $\alpha_1 = 0.1$, $\alpha_2 = 0.5$, and $\alpha_3 = 0.9$ show distinct temporal decay behaviors depending on the fractional order. Smaller α_k values lead to slower temporal evolution, which indicates stronger memory effects and lower diffusion rates, while larger α_k values correspond to faster diffusion due to weaker memory effects. This behavior reflects the intrinsic relationship between the fractional order and the degree of nonlocality in time. In general, the diffusion process evolves more rapidly at early times because the temporal gradients are initially large. As time progresses, these gradients become smaller. Therefore, even if two solutions have the same profile at a given time t, the solution with a smaller fractional order α changes less during the next time interval Δt. This is because a smaller α gives greater weight to the early-time history, where the temporal gradients were larger. Consequently, the stronger memory effect slows the present evolution and results in a lower effective diffusion rate. In contrast, a larger fractional order places less weight on the past history, so the solution evolves more rapidly. Figure 4(b) presents the multi-term solutions for three sets of weighting parameters $\boldsymbol{\beta}$. Variations in $\boldsymbol{\beta}$ change the relative contribution of each fractional order and modify the balance between short-term and long-term memory effects. For example, if a larger weight is assigned to a smaller α_k, then the memory effect becomes stronger and the diffusion process becomes slower. In contrast, if a larger weight is assigned to a larger α_k, then the diffusion process becomes faster. The computational results verify that the proposed multi-term normalized operator provides a flexible methodology to control the diffusion dynamics through the parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ while preserving the integral normalization property.



(a) (b)

Figure 4. (a) Single-term solutions $u(x, t)$ are shown for the fractional orders $\alpha_1 = 0.1$, $\alpha_2 = 0.5$, and $\alpha_3 = 0.9$. Smaller α_k values result in slower decay, which implies stronger memory effects and slower diffusion, whereas larger α_k values yield faster evolution due to weaker memory influence. (b) Multi-term solutions are presented for three sets of weighting parameters $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3) = (0.8, 0.1, 0.1)$, $(0.1, 0.8, 0.1)$, and $(0.1, 0.1, 0.8)$. The common parameter vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3) = (0.1, 0.5, 0.9)$ is applied in all cases. The results indicate that variations in $\boldsymbol{\beta}$ alter the balance between low- and high-order fractional effects, which modifies the effective diffusion rate and memory response. The initial condition is given by Eq (3.1), and the final time is $T = 0.05$.

Next, we choose the following initial condition on $\Omega = (0, \pi)$:

$$u(x, 0) = \sin\left(7\sqrt{\pi x}\right). \tag{3.2}$$

Figure 5 shows the numerical results of the single-term and multi-term normalized time-fractional diffusion models with the initial condition equation (3.2), which contains both low- and high-frequency components. In Figure 5(a), the single-term solutions are presented for the fractional orders $\alpha_1 = 0.1$, $\alpha_2 = 0.5$, and $\alpha_3 = 0.9$. The computational results indicate that smaller α_k values produce slower decay and stronger memory effects, leading to delayed diffusion, whereas larger α_k values yield faster relaxation due to weaker memory influence. In Figure 5(b), the multi-term solutions corresponding to three sets of weighting parameters $\boldsymbol{\beta}$ demonstrate how the interaction between multiple fractional orders affects the diffusion dynamics. If a larger weight is assigned to a smaller α_k, then the long-term memory effect becomes stronger and the diffusion process becomes slower. In contrast, if a larger weight is assigned to a larger α_k, then the diffusion process becomes faster. The coexistence of low- and high-frequency oscillations reveals distinct diffusion behavior: high-frequency modes exhibit stronger damping and amplitude reduction, indicating a higher sensitivity to fractional memory effects, while low-frequency modes maintain smoother propagation due to reduced dissipation. These numerical results confirm that the parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ provide a flexible mechanism to control the frequency-dependent diffusion characteristics and memory response of the system. Consequently, the multi-term normalized fractional operator effectively captures the balance between short-term and long-term memory effects in complex temporal evolutions.

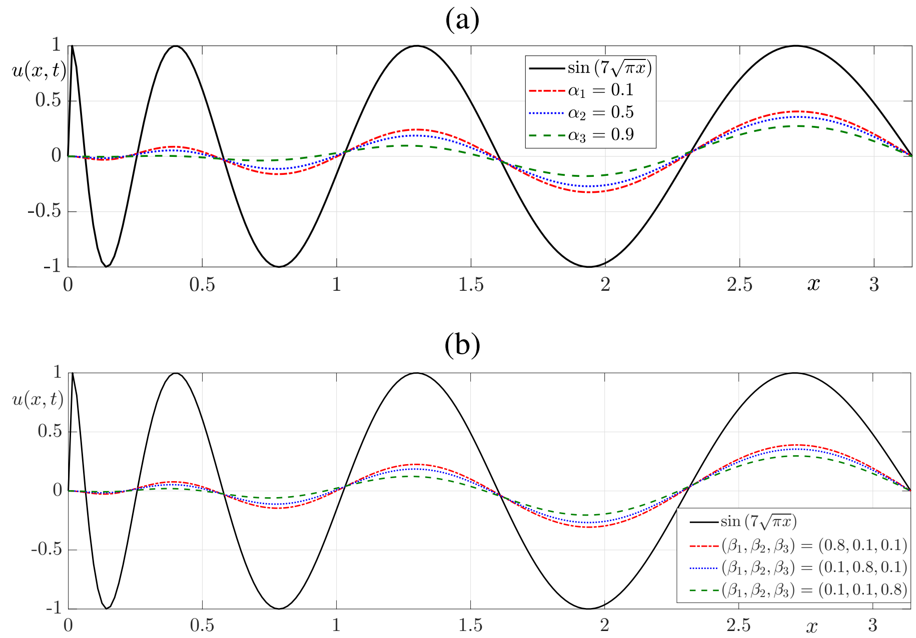


Figure 5. (a) Single-term solutions $u(x,t)$ are shown for the fractional orders $\alpha_1 = 0.1$, $\alpha_2 = 0.5$, and $\alpha_3 = 0.9$ with the initial condition equation (3.2) over the domain $(0, \pi)$. Smaller α_k values cause slower decay and stronger memory effects, whereas larger α_k values lead to faster relaxation due to weaker memory influence. (b) Multi-term solutions are presented for three sets of weighting parameters $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3) = (0.8, 0.1, 0.1)$, $(0.1, 0.8, 0.1)$, and $(0.1, 0.1, 0.8)$, where the common parameter vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3) = (0.1, 0.5, 0.9)$ is used in all cases. The results show that variations in $\boldsymbol{\beta}$ change the influence of low- and high-order fractional components and modify the diffusion behavior and memory response. The final time is $T = 0.1$.

Next, we perform numerical convergence tests to demonstrate the theoretical convergence rate proved in Theorem 2 (i.e., $C_T(\Delta t^{2-\alpha_{\max}} + h^2)$). For this test, an exact solution is used, and the corresponding source term is constructed accordingly. A similar convergence test based on a constructed exact solution can be found in [38]. We consider the following equation:

$$\frac{\partial^{\boldsymbol{\alpha},\boldsymbol{\beta}} u(x,t)}{\partial t^{\boldsymbol{\alpha},\boldsymbol{\beta}}} = \frac{\partial^2 u(x,t)}{\partial x^2} + g(x,t) \quad \text{for } (x,t) \in \Omega \times (0,\ \infty),$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$ and $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3)$ denote the vectors of fractional orders and weighting coefficients, respectively, with $0 < \alpha_k < 1$, $\beta_k > 0$, and $\beta_1 + \beta_2 + \beta_3 = 1$. We choose the following exact solution:

$$u(x,t) = (1+t^2)\sin x,$$

for which the corresponding source term is given by

$$g(x,t) = 2t\left(\frac{\beta_1}{2-\alpha_1} + \frac{\beta_2}{2-\alpha_2} + \frac{\beta_3}{2-\alpha_3}\right)\sin x + (1+t^2)\sin x.$$

The computational domain is $\Omega = (0, \pi)$, with the initial condition $u(x, 0) = \sin x$ and the boundary conditions $u(0, t) = u(\pi, t) = 0$. The final time is set to $T = 0.1$, and $N_t = T/\Delta t$ is the total number of time steps. The numerical error is computed using the discrete L^2-norm, defined as

$$E = \|u(x_i, T) - U_i^{N_t}\|_2 = \sqrt{h\sum_{i=2}^{N_x-1}(u(x_i, T) - U_i^{N_t})^2},$$

where $U_i^{N_t}$ is the computational solution at the final time $t = T$. The temporal and spatial convergence rates are computed as $\log_2(E^{\Delta t}/E^{\Delta t/2})$ and $\log_2(E^h/E^{h/2})$, respectively, where $E^{\Delta t}$ and E^h are the numerical errors computed using the time step Δt and the mesh size h, respectively. For the temporal convergence test, the spatial grid is fixed at $N_x = 4001$, and the time step is refined as $\Delta t = T/2^m$, $m = 10$, 11, and 12. Table 1 shows the numerical errors and temporal convergence rates. The results demonstrate the temporal convergence order of $2 - \alpha_{\max}$.

Table 1. Numerical errors and temporal convergence rates.

α	β	$\Delta t = 0.2/2^{10}$		$\Delta t = 0.2/2^{11}$		$\Delta t = 0.2/2^{12}$
		Error	Order	Error	Order	Error
	$(0.8, 0.1, 0.1)$	5.02×10^{-7}	1.09	2.35×10^{-7}	1.07	1.12×10^{-7}
$(0.1, 0.5, 0.9)$	$(0.1, 0.8, 0.1)$	6.49×10^{-7}	1.17	2.89×10^{-7}	1.13	1.32×10^{-7}
	$(0.1, 0.1, 0.8)$	4.17×10^{-6}	1.10	1.95×10^{-6}	1.10	9.11×10^{-7}
	$(0.8, 0.1, 0.1)$	1.38×10^{-7}	1.29	5.65×10^{-8}	1.15	2.54×10^{-8}
$(0.3, 0.5, 0.7)$	$(0.1, 0.8, 0.1)$	2.52×10^{-7}	1.36	9.80×10^{-8}	1.27	4.06×10^{-8}
	$(0.1, 0.1, 0.8)$	8.32×10^{-6}	1.29	3.40×10^{-7}	1.27	1.41×10^{-7}

For the spatial convergence test, the time step is fixed at $\Delta t = T/2^{11}$, and the spatial grid is refined as $N_x = 2^m + 1$, $m = 5$, 6, and 7. Table 2 lists the numerical errors and spatial convergence rates. The results demonstrate the second-order spatial convergence.

Because the main objective of this paper is to introduce the novel normalized multi-term time-fractional derivative, we primarily focus on the one-dimensional case. Nevertheless, the proposed formulation can be directly extended to higher dimensions. As a simple example, we consider the following 2D multi-term normalized time-fractional diffusion model:

$$\frac{\partial^{\alpha,\beta} u(x,y,t)}{\partial t^{\alpha,\beta}} = \Delta u(x,y,t), \qquad (x,y,t) \in \Omega \times (0,\infty), \tag{3.3}$$

where $\Omega = (L_x, R_x)\times(L_y, R_y)$ is the two-dimensional computational domain. The multi-term normalized time-fractional derivative is the same as that defined in Eq (1.4), and only the spatial dimension is

Table 2. Numerical errors and spatial convergence rates.

$\boldsymbol{\alpha}$	$\boldsymbol{\beta}$	$\Delta t = 0.2/2^{10}$		$\Delta t = 0.2/2^{11}$		$\Delta t = 0.2/2^{12}$
		Error	Order	Error	Order	Error
(0.1, 0.5, 0.9)	(0.8, 0.1, 0.1)	9.35×10^{-5}	1.99	2.36×10^{-5}	1.96	6.06×10^{-6}
	(0.1, 0.8, 0.1)	9.45×10^{-5}	1.99	2.38×10^{-5}	1.95	6.17×10^{-6}
	(0.1, 0.1, 0.8)	9.73×10^{-5}	1.92	2.58×10^{-5}	1.71	7.90×10^{-6}
(0.3, 0.5, 0.7)	(0.8, 0.1, 0.1)	9.37×10^{-5}	2.00	2.35×10^{-5}	1.99	5.91×10^{-6}
	(0.1, 0.8, 0.1)	9.42×10^{-5}	2.00	2.36×10^{-5}	1.98	5.98×10^{-6}
	(0.1, 0.1, 0.8)	9.50×10^{-5}	1.98	2.40×10^{-5}	1.94	6.25×10^{-6}

extended from one to two dimensions. The computational domain is discretized by $\Omega_h = \{(x_i, y_j) | x_i = L_x + (a-0.5)h, 1 \le a \le N_x,\ y_j = L_y + (b-0.5)h, 1 \le b \le N_y\}$, where $h = (R_x - L_x)/N_x = (R_y - L_y)/N_y$. Let $u_{ij}^n = u(x_i, y_j, t_n)$, where $t_n = (n-1)\Delta t$. For the computational efficiency, the Fourier spectral method is used for the spatial discretization. To formulate the Fourier spectral method, we define the discrete Fourier transform of u_{ij}^n as

$$\hat{u}_{pq}^n = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} u_{ij}^n e^{-i(\mu_p x_i + \lambda_q y_j)},$$

for $-N_x/2 + 1 \le p \le N_x/2$, $-N_y/2 + 1 \le q \le N_y/2$, where $\mu_p = 2\pi p/(R_x - L_x)$, $\lambda_q = 2\pi q/(R_y - L_y)$. The corresponding inverse discrete Fourier transform is given by the following:

$$u_{ij}^n = \frac{1}{N_x N_y} \sum_{p=-N_x/2+1}^{N_x/2} \sum_{q=-N_y/2+1}^{N_y/2} \hat{u}_{pq}^n e^{i(\mu_p x_i + \lambda_q y_j)}. \tag{3.4}$$

Using the discrete Fourier transform to Eq (3.3) leads to the following:

$$w_n^n \frac{\hat{u}_{pq}^{n+1} - \hat{u}_{pq}^n}{\Delta t} = -(\mu_p^2 + \lambda_q^2)\hat{u}_{pq}^{n+1} - \hat{F}_{pq}^n,$$

where $\hat{F}_{pq}^n$ denotes the discrete Fourier transform of F. This can be rewritten as follows:

$$\hat{u}_{pq}^{n+1} = \frac{w_n^n \hat{u}_{pq}^n - \Delta t \hat{F}_{pq}^n}{w_n^n + \Delta t(\mu_p^2 + \lambda_q^2)}.$$

Finally, u_{ij}^{n+1} is obtained from the inverse discrete Fourier transform in Eq (3.4). Further details can be found in [39]. For the numerical experiments, we set $\Omega = (0,1) \times (0,1)$, $N_x = N_y = 256$, $h = 1/N_x$, and $\Delta t = 0.0001$. The initial condition is as follows:

$$u(x, y, 0) = \sin(2\pi x)\sin(2\pi y).$$

Figure 6 displays the temporal evolution of the computational solution for different combinations of the fractional orders $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3) = (0.1, 0.5, 0.9)$ and their corresponding weights $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3) = (0.8, 0.1, 0.1), (0.1, 0.8, 0.1)$, and $(0.1, 0.1, 0.8)$. Similar to the one-dimensional case, the weighting parameters $\boldsymbol{\beta}$ change the relative influence of the fractional orders and control the memory effect and the diffusion rate. A larger weight for a smaller fractional order results in slower diffusion, whereas a larger weight for a larger fractional order leads to faster diffusion.

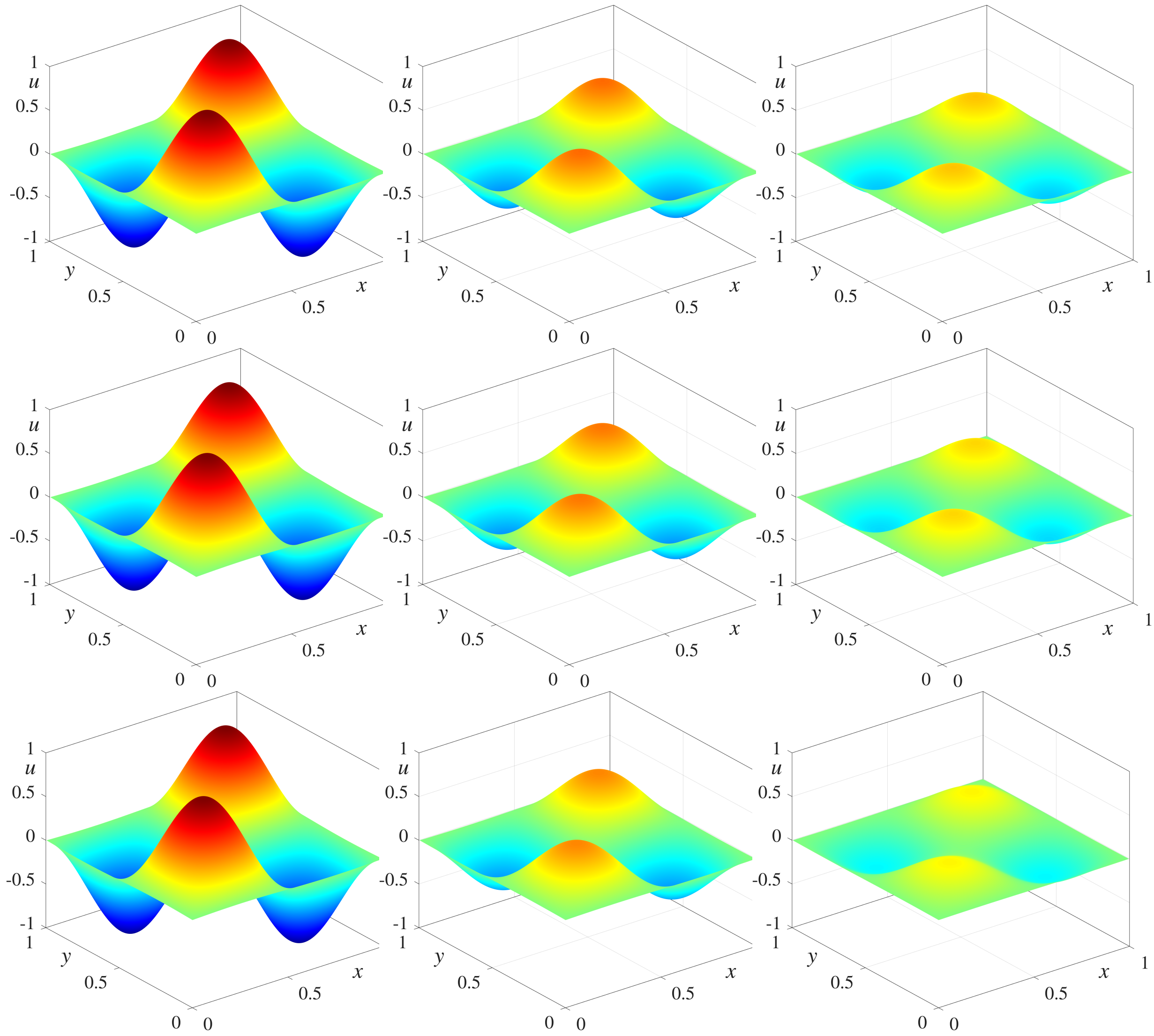


Figure 6. Time evolution of the two-dimensional numerical solutions $u(x, y, t)$ for three weighting parameter vectors $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3)$. The fractional-order vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3) = (0.1, 0.5, 0.9)$ is used in all cases. From left to right, the solutions are shown at $t = 0$, 0.01, and 0.02. From top to bottom, the weighting parameter vectors are $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3) = (0.8, 0.1, 0.1)$, $(0.1, 0.8, 0.1)$, and $(0.1, 0.1, 0.8)$, respectively. A larger weight corresponding to a smaller fractional order results in slower diffusion, whereas a larger weight corresponding to a larger fractional order leads to faster diffusion.

4. Conclusions

We proposed a new multi-term normalized time-fractional diffusion model. The proposed model extends the normalized time-fractional derivative using several fractional orders and weighting coefficients. It keeps the normalization property and gives more choices to describe memory effects than the single-term model. We developed an implicit FDM for the proposed equation. The linear system is tridiagonal and is solved by the Thomas algorithm. We proved that the numerical scheme is unconditionally stable and convergent. The method has a temporal accuracy of $2 - \alpha_{\max}$ and second-order spatial accuracy. The numerical convergence tests agree well with the theoretical results. The numerical results showed that the fractional orders and the weighting coefficients change the diffusion speed and the memory effect. The proposed model can produce many weighting functions that the single-term model cannot produce. Therefore, it can describe a wider range of diffusion behaviors. Additionally, we presented a simple two-dimensional example to validate that the proposed model can be extended to higher-dimensional problems. In future work, we will study fast numerical methods for long-time simulations. Moreover. we will study variable-order models [40], nonlinear problems, inverse problems, and practical applications of the proposed model.

Use of AI tools declaration

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

Acknowledgments

The first author (Y. Choi) was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2022R1I1A3072824). The corresponding author (J. Kim) was supported by the Korea University Grant. The authors sincerely thank the reviewers for their helpful comments and suggestions.

Conflict of interest

The authors declare that they have no competing interests.

References

1. A. A. Alikhanov, A new difference scheme for the time fractional diffusion equation, *J. Comput. Phys.*, **280** (2015), 424–438. https://doi.org/10.1016/j.jcp.2014.09.031
2. L. Hu, Z. Li, X. Yang, A strong positivity property and a related inverse source problem for multi-term time-fractional diffusion equations, *Acta Math. Sci.*, **44** (2024), 2019–2040. https://doi.org/10.1007/s10473-024-0523-2
3. B. Derbissaly, M. Sadybekov, Inverse source problem for multi-term time-fractional diffusion equation with nonlocal boundary conditions, *AIMS Math.*, **9** (2024), 9969–9988. https://doi.org/10.3934/math.2024488
4. H. K. Dwivedi, Rajeev, A fast difference scheme for the multi-term time fractional advection-diffusion equation with a non-linear source term, *Chin. J. Phys.*, **89** (2024), 86–103. https://doi.org/10.1016/j.cjph.2024.02.051
5. F. Mostajeran, S. M. Hosseini, Solving multi-term time-fractional diffusion equations using a high-order fractional finite difference scheme accompanied by neural network techniques, *Numer. Methods Partial Differ. Equations*, **41** (2025), e23156. https://doi.org/10.1002/num.23156

6. Y. Liu, M. Yamamoto, Uniqueness of orders and parameters in multi-term time-fractional diffusion equations by short-time behavior, *Inverse Probl.*, **39** (2023), 024003. https://doi.org/10.1088/1361-6420/acab7a
7. J. Liu, Y. Liu, X. Yu, X. Ye, An efficient numerical method based on QSC for multi-term variable-order time fractional mobile-immobile diffusion equation with Neumann boundary condition, *Electron. Res. Arch.*, **33** (2025), 642–666. https://doi.org/10.3934/era.2025030
8. C. Hou, H. Chen, Stability and pointwise-in-time convergence analysis of a finite difference scheme for a 2D nonlinear multi-term subdiffusion equation, *Electron. Res. Arch.*, **33** (2025), 1476–1489. https://doi.org/10.3934/era.2025069
9. W. Zhang, F. Xie, J. Ni, On the existence of solutions for multi-term p-Laplacian fractional differential equations with anti-periodic boundary conditions, *Electron. Res. Arch.*, **34** (2026), 3593–3610. https://doi.org/10.3934/era.2026161
10. X. X. Xiong, Y. S. Li, Simultaneous inversion for the space-dependent diffusion coefficient and the fractional orders in a multi-term time-fractional diffusion equation, *J. Sci. Comput.*, **106** (2026), 70. https://doi.org/10.1007/s10915-026-03200-3
11. Y. H. Youssri, A. G. Atta, An enhanced shifted Jacobi collocation framework for the time-fractional FitzHugh-Nagumo equation, *Int. J. Mod. Phys. C*, **2026** (2026), 2750107. https://doi.org/10.1142/S0129183127501075
12. Y. H. Youssri, M. M. Shaikh, I. M. Batiha, N. Anakira, I. Irianto, T. Sasa, An enhanced ultraspherical collocation framework with Chebyshev nodes for the time-fractional FitzHugh-Nagumo equation, *AIMS Math.*, **11** (2026), 13710–13743. http://doi.org/10.3934/math.2026565
13. M. M. Alsuyuti, E. H. Doha, S. S. Ezz-Eldien, Numerical simulation for classes of one- and two-dimensional multi-term time-fractional diffusion and diffusion-wave equation based on shifted Jacobi Galerkin scheme, *Math. Methods Appl. Sci.*, **48** (2025), 8217–8244. https://doi.org/10.1002/mma.9659
14. R. Hafez, Y. Youssri, E. Doha, M. Hammad, Exponential/fractional Jacobi spectral Galerkin and collocation schemes for the multi-dimensional time-fractional diffusion equations on the semi-infinite domain, *Hacet. J. Math. Stat.*, **55** (2026), 942–961. https://doi.org/10.15672/hujms.1399274
15. L. L. Sun, M. L. Chang, Galerkin spectral method for a multi-term time-fractional diffusion equation and an application to inverse source problem, *Netw. Heterog. Media*, **18** (2023), 212–243. https://doi.org/10.3934/nhm.2023008
16. S. G. Kamel, A. G. Atta, Y. H. Youssri, An explicit spectral Tau method with Delannoy polynomials for the time-fractional diffusion equation, *Open J. Math. Anal.*, **9** (2025), 134–144. https://doi.org/10.30538/psrp-oma2025.0169
17. Y. H. Youssri, L. A. Alnaser, A. G. Atta, A spectral collocation approach for time-fractional Korteweg-de Vries-Burgers equation via first-kind Chebyshev polynomials, *Contemp. Math.*, **6** (2025), 1501–1519. https://doi.org/10.37256/cm.6220255948
18. B. Fan, Efficient numerical method for multi-term time-fractional diffusion equations with Caputo-Fabrizio derivatives, *AIMS Math.* **9** (2024), 7293–7320. https://doi.org/10.3934/math.2024354

19. S. Santra, Analysis of a higher-order scheme for multi-term time-fractional integro-partial differential equations with multi-term weakly singular kernels, *Numer. Algorithms*, **99** (2025), 1649–1695. https://doi.org/10.1007/s11075-024-01927-4

20. S. Sabir, A. Ahmad, A. Kanaujiya, J. Mohapatra, A robust numerical scheme and analysis for a class of multi-term time-fractional advection-diffusion equation with variable coefficients, *J. Appl. Math. Comput.*, **71** (2025), 1943–1976. https://doi.org/10.1007/s12190-024-02312-z

21. A. Mehri, M. S. Abdo, H. Bouhadjera, A. Dawood, K. Aldwoah, R. Egami, Finite element analysis of a multi-term nonlinear time-fractional convection-diffusion equation with Caputo-Fabrizio derivative, *Bound. Value Probl.*, **2025** (2025), 161. https://doi.org/10.1186/s13661-025-02112-9

22. X. Zhang, Y. Chen, L. Wei, S. Kumar, Numerical simulation based on interpolation technique for multi-term time-fractional convection-diffusion equations, *Fractal Fract.*, **8** (2024), 687. https://doi.org/10.3390/fractalfract8120687

23. J. Kim, A normalized time-fractional Lotka–Volterra model, *Korean J. Math.*, **33** (2025), 87–97. https://doi.org/10.11568/kjm.2025.33.2.87

24. M. Jornet, J. J. Nieto, Power-series solution of the L-fractional logistic equation, *Appl. Math. Lett.*, **154** (2024), 109085. https://doi.org/10.1016/j.aml.2024.109085

25. C. Lee, Y. Nam, M. Bang, S. Ham, J. Kim, Numerical investigation of the dynamics for a normalized time-fractional diffusion equation, *AIMS Math.*, **9** (2024), 26671–26687. https://doi.org/10.3934/math.20241297

26. T. Akbar, S. U. Arifeen, I. Ali, S. A. Idris, S. Islam, Computational study of variable order time-fractional differential equations arising in modeling of transport processes and viscoelastic oscillator, *Ain Shams Eng. J.*, **16** (2025), 103532. https://doi.org/10.1016/j.asej.2025.103532

27. S. U. Arifeen, S. Haq, I. Ali, S. F. Aldosary, Galerkin approximation for multi-term time-fractional differential equations, *Ain Shams Eng. J.*, **15** (2024), 102806. https://doi.org/10.1016/j.asej.2024.102806

28. H. Chen, Y. Jiang, J. Wang, Grünwald-Letnikov scheme for a multi-term time fractional reaction-subdiffusion equation, *Commun. Nonlinear Sci. Numer. Simul.*, **132** (2024), 107930. https://doi.org/10.1016/j.cnsns.2024.107930

29. L. L. Sun, Y. S. Li, Y. Zhang, Simultaneous inversion of the potential term and the fractional orders in a multi-term time-fractional diffusion equation, *Inverse Probl.*, **37** (2021), 055007. https://doi.org/10.1088/1361-6420/abf162

30. M. P. Alam, A. Khan, D. Baleanu, A high-order unconditionally stable numerical method for a class of multi-term time-fractional diffusion equation arising in the solute transport models, *Int. J. Comput. Math.*, **100** (2023), 105–132. https://doi.org/10.1080/00207160.2022.2082248

31. R. Choudhary, S. Singh, D. Kumar, A higher order unconditionally stable numerical technique for multi-term time-fractional diffusion and advection-diffusion equations, *Comput. Appl. Math.*, **43** (2024), 303. https://doi.org/10.1007/s40314-024-02688-5

32. R. Du, Z. Sun, Temporal second-order difference methods for solving multi-term time-fractional mixed diffusion and wave equations, *Numer. Algorithms*, **88** (2021), 191–226. https://doi.org/10.1007/s11075-020-01037-x

33. A. A. Alikhanov, P. Yadav, V. K. Singh, M. S. Asl, A high-order compact difference scheme for the multi-term time-fractional Sobolev-type convection-diffusion equation, *Comput. Appl. Math.*, **44** (2025), 115. https://doi.org/10.1007/s40314-024-03077-8

34. H. Jafari, B. F. Malidareh, V. R. Hosseini, Collocation discrete least squares meshless method for solving nonlinear multi-term time fractional differential equations, *Eng. Anal. Bound. Elem.*, **158** (2024), 107–120. https://doi.org/10.1016/j.enganabound.2023.10.014

35. J. Lin, S. Reutskiy, Y. Zhang, Y. Sun, J. Lu, The novel analytical-numerical method for multi-dimensional multi-term time-fractional equations with general boundary conditions, *Mathematics*, **11** (2023), 929. https://doi.org/10.3390/math11040929

36. S. Kwak, Practical implementation of boundary conditions in the Thomas algorithm, *J. Korean Soc. Ind. Appl. Math.*, **29** (2025), 171–183. https://doi.org/10.12941/jksiam.2025.29.171

37. X. Wu, K. Zhang, J. Ma, J. Kim, An unconditionally stable maximum-principle-preserving numerical method for a normalized time-fractional diffusion equation, *AIMS Math.*, **11** (2026), 22897–22919. https://doi.org/10.3934/math.2026922

38. J. Ren, Z. Sun, Efficient and stable numerical methods for multi-term time fractional sub-diffusion equations, *East Asian J. Appl. Math.*, **4** (2014), 242–266. https://doi.org/10.4208/eajam.181113.280514a

39. K. Zhang, S. Lee, X. Wu, M. Nan, J. Kim, A Fourier spectral method for normalized time-fractional diffusion equations in two and three dimensions, *Electron. Res. Arch.*, **34** (2026), 1939–1956. https://doi.org/10.3934/era.2026087

40. C. Lee, J. Kim, A normalized variable-order time-fractional diffusion equation, *Phys. A*, **680** (2025), 131052. https://doi.org/10.1016/j.physa.2025.131052

Appendix

The MATLAB code for the single-term and multi-term normalized time-fractional diffusion equations is given in Listing 1.

Listing 1. MATLAB implementation of the single-term and multi-term normalized time-fractional diffusion equations

```
clear; Leftx=0; Rightx=1; nx=201; h=(Rightx-Leftx)/(nx-1);
x=linspace(Leftx,Rightx,nx); T=0.05; nt=401; dt=T/nt;
uu=zeros(nx,nt+1); uu(:,1)=sin(2*pi*x);
a=(-1/h^2)*ones(nx,1); c=a;
figure(1); clf; box on; hold on;
set(gcf, 'position', [100 100 700 550]); set(gca, 'fontsize', 17);
set(gca, 'TickLabelInterpreter', 'latex');
xticks([0 0.25 0.5 0.75 1.0]); yticks([-1 -0.5 0 0.5 1])
plot(x, uu(:, 1), 'k', 'LineWidth', 1.5); alphas = [0.1 0.5 0.9];
legendLabels = cell(1, length(alphas));
```

```
legendLabels{1}=sprintf('$\\sin(2\\pi x)$');
line_styles = {'r-.', 'b:', '--'}; dark_green = [0 0.5 0];

for ii = 1:length(alphas)
    alphaval = alphas(ii);
    for nn = 1:nt
        Denominator = nn^(1-alphaval);
        for pp = 1:nn
            w(pp) = ((nn+1-pp)^(1-alphaval)-(nn-pp)^(1-alphaval))/Denominator;
        end
        F = zeros(nx, 1);
        if nn > 1
            for pp = 1:nn-1
                F = F+w(pp)*(uu(:,pp+1)-uu(:,pp))/dt;
            end
        end
        for i = 1:nx
            d(i) = w(nn)/dt+2.0/h^2;
        end
        f = w(nn)/dt*uu(:,nn)-F;
        uu(2:nx-1, nn+1) = thomas(a(2:nx-1), d(2:nx-1), c(2:nx-1), f(2:nx-1));
    end
    if ii == 3
    plot(x,uu(:,end),line_styles{ii},'Color',dark_green,'LineWidth',1.5);
    else
    plot(x, uu(:, end), line_styles{ii}, 'LineWidth', 1.5);
    end
    legendLabels{ii+1}=sprintf('$\\alpha_{%d} = %.1f$',ii,alphaval);
end
lgd=legend(legendLabels,'Interpreter','latex','fontsize',17);
set(lgd, 'Position', [0.62 0.625 0.18 0.19])
text('Interpreter','latex','String','$$x$$','Position', ...
     [0.868 -1.095],'FontSize',20)
text('Interpreter','latex','String','$$u(x,t)$$','Position', ...
     [-0.115 0.77],'FontSize',17)
set(gcf,'PaperPositionMode','auto'); grid on

figure(2); clf; hold on; box on;
set(gcf, 'position', [100 100 700 550]); set(gca, 'fontsize', 17);
set(gca, 'TickLabelInterpreter', 'latex');
xticks([0 0.25 0.5 0.75 1.0]); yticks([-1 -0.5 0 0.5 1])
plot(x, uu(:, 1), 'k', 'LineWidth', 1.5); alphas = [0.1 0.5 0.9];
legendLabels = cell(1, length(alphas));
legendLabels{1}=sprintf('$\\sin(2\\pi x)$');
line_styles = {'r-.', 'b:', '--'}; dark_green = [0 0.5 0];
beta_set=[0.8 0.1 0.1; 0.1 0.8 0.1; 0.1 0.1 0.8];

for k = 1:size(beta_set,1)
 beta=beta_set(k,:);
 for nn = 1:nt
  Denominator = nn.^(1-alphas);
  for pp = 1:nn
  w(pp)=beta*(((nn+1-pp).^(1-alphas)-(nn-pp).^(1-alphas))./Denominator)';
  end
```

```
    F = zeros(nx, 1);
    if nn > 1
      for pp = 1:nn-1
          F = F+w(pp)*(uu(:,pp+1)-uu(:,pp))/dt;
      end
    end
    for i = 1:nx
     d(i) = w(nn)/dt+2.0/h^2;
    end
    f = w(nn)/dt*uu(:,nn)-F;
     uu(2:nx-1, nn+1) = thomas(a(2:nx-1), d(2:nx-1), c(2:nx-1), f(2:nx-1));
    end
    if k == 3
    plot(x, uu(:, end), line_styles{k}, 'Color', dark_green, 'LineWidth', 1.5);
    else
    plot(x, uu(:, end), line_styles{k}, 'LineWidth', 1.5);
    end
    legendLabels{k+1}=sprintf('$(\\beta_1,\\beta_2,\\beta_3)=(%.1f, %.1f, %.1f)$'
    ,beta_set(k,:));
  end
  lgd=legend(legendLabels, 'Interpreter', 'latex', 'fontsize', 15);
  set(lgd, 'Position', [0.6 0.72 0.18 0.19])
  text('Interpreter','latex','String','$$x$$','Position', ...
       [0.868 -1.095],'FontSize',20)
  text('Interpreter','latex','String','$$u(x,t)$$','Position', ...
       [-0.115 0.77],'FontSize',17)
  set(gcf,'PaperPositionMode','auto'); grid on

  function xx = thomas(A, B, C, ff)
      n = length(ff);
      for i = 2:n
          MM = A(i)/B(i-1);
          B(i) = B(i)-MM*C(i-1);
          ff(i) = ff(i)-MM*ff(i-1);
      end
      xx(n) = ff(n)/B(n);
      for i = n-1:-1:1
          xx(i) = (ff(i)-C(i)*xx(i+1))/B(i);
      end
      xx = xx';
  end
```

AIMS Press