AIMS Mathematics, 11(7): 22897–22919.
DOI: 10.3934/math.2026922
Received: 19 May 2026
Revised: 30 June 2026
Accepted: 16 July 2026
Published: 28 July 2026

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

Research article

An unconditionally stable maximum-principle-preserving numerical method for a normalized time-fractional diffusion equation

Xinpei Wu[1], **Ke Zhang**[2,3], **Juho Ma**[1] **and Junseok Kim**[1,*]

[1] Department of Mathematics, Korea University, Seoul 02841, Republic of Korea
[2] School of Mathematics and Statistics, Heze University, Heze 274015, China
[3] College of Mathematics and System Sciences, Xinjiang University, Urumqi 830046, China

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

Abstract: In this paper, we present the mathematical and numerical analysis of a numerical method for a normalized time-fractional diffusion equation. The method is unconditionally stable and preserves the maximum principle. The normalized derivative was defined using a weighting function. The integral of this function is always equal to one for all fractional orders and times. This property makes it possible to compare the diffusion behavior under different fractional orders in a fair way. We considered a recently proposed finite difference scheme with an implicit time discretization for the equation. The resulting algebraic system has a tridiagonal matrix. The Thomas algorithm was applied to solve the system efficiently. We proved that the numerical scheme is unconditionally stable, satisfies the maximum principle, and converges correctly. The convergence order is $O(\Delta t^{2-\alpha})$ in time and $O(h^2)$ in space. Several numerical examples were also presented. The results agree with the theoretical analysis and further validate the influence of different fractional orders on the diffusion behavior.

Keywords: unconditionally stable numerical scheme; normalized time-fractional derivative; convergence analysis
Mathematics Subject Classification: 35R11, 39A14, 80M20

1. Introduction

Fractional diffusion equations describe anomalous diffusion processes with memory effects [1]. Recent studies increasingly interpret spatially and temporally varying fractional orders as physically meaningful quantities that characterize heterogeneous memory effects and anomalous transport mechanisms in complex diffusion systems [2]. Zheng et al. [3] investigated variable-order time-fractional diffusion equations that model anomalous subdiffusion and memory effects without the nonphysical initial-time singularities of constant-order models. Hong et al. [4] investigated an

inverse problem for identifying a spatially variable order in a subdiffusion model with a generalized Caputo time derivative from boundary flux measurements. Although variable-order models provide additional flexibility in modeling heterogeneous memory effects, the contribution of the memory kernel still changes with the fractional order. This makes it difficult to isolate and consistently interpret the pure effect of the fractional order on diffusion dynamics. To resolve this issue, Lee and Kim [5] recently introduced a normalized, variable-order, time-fractional diffusion model in which the total contribution of the memory kernel is independent of the fractional order. Such a normalized formulation provides a more consistent method for comparing diffusion behaviors associated with different fractional orders [6].

Substantial effort has been devoted to the development of efficient and stable computational algorithms for variable-order and distributed-order fractional diffusion problems [7]. Similar ideas on stability and structure-preserving approximations have also been extensively explored in phase-field [8, 9] and diffuse-interface models [10, 11]. Kanwal et al. [12] developed an explicit finite difference method (FDM) for initial-boundary value problems involving linear and semilinear variable-order time-fractional equations with the Caputo derivative. Jia et al. [13] investigated a variable-order fractional wave equation and developed a fast method to reduce computational cost. Although the normalization removes the inconsistency in the total memory contribution across different fractional orders, the resulting operator still retains a strong temporal nonlocality. Moreover, the normalized weighting structure differs from conventional fractional kernels and therefore requires specific numerical analysis. Numerical approximations for normalized fractional diffusion equations require the accumulation of the entire solution history, which introduces significant challenges in stability analysis, convergence, and computational efficiency. In this study, we consider the following normalized time-fractional diffusion equation on $\Omega = (L,\ R)$:

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} = \frac{\partial^2 u(x,t)}{\partial x^2} \quad \text{for } (x,t) \in \Omega \times (0,\ \infty), \tag{1.1}$$

$$u(x,0) = u_0(x), \quad x \in \Omega, \tag{1.2}$$

$$u(L,t) = u(R,t) = 0, \quad t \geq 0, \tag{1.3}$$

where α is the order of the time-fractional derivative, $u(x,t)$ is the concentration, x and t are the spatial and temporal variables, respectively, $\Omega = (L,R)$ is the one-dimensional spatial domain with left boundary L and right boundary R, and $u_0(x)$ denotes the given initial condition. The fractional derivative operator is defined as:

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} = \int_0^t w_\alpha^t(s) \frac{\partial u(x,s)}{\partial s} ds, \quad 0 < \alpha < 1. \tag{1.4}$$

Here, the weighting function is defined as follows:

$$w_\alpha^t(s) = \frac{1-\alpha}{t^{1-\alpha}(t-s)^\alpha}, \tag{1.5}$$

which satisfies

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

The above normalized time-fractional diffusion equation was numerically investigated in [14, 15]. However, the theoretical analysis of the numerical method was not studied. In this study, we present theoretical results for the recently proposed method, including unconditional stability, preservation of the maximum principle, and convergence. Maximum principle preservation is an essential property for maintaining the physical validity of the numerical solution. More generally, the preservation of maximum bounds is also important for guaranteeing physically meaningful solutions, as illustrated by the nonlinear Allen–Cahn equation [16]. We show that the method has temporal accuracy of $O(\Delta t^{2-\alpha})$ and spatial accuracy of $O(h^2)$. Finally, we perform several numerical tests to confirm the theoretical results and to study the effect of different fractional orders on the diffusion behavior.

The rest of this paper is organized as follows. Section 2 describes the recently proposed numerical scheme for the normalized time-fractional diffusion equation. This section also gives proofs of unconditional stability and convergence. Section 3 presents several numerical examples for testing the recently proposed method. Section 4 concludes the paper and discusses possible future work.

2. Numerical scheme and unconditional stability

In this section, we describe a recently proposed FDM for the normalized time-fractional diffusion equation. The recently proposed method uses an implicit time discretization and a tridiagonal matrix system solved by the Thomas algorithm. We also prove the unconditional stability, maximum principle, and convergence of the method.

Consider $\Omega = (L,\ R)$ as the domain for our computation. We construct a uniform grid system $\Omega_h = \{x_i | x_i = L + (i-1)h,\ i = 1, \ldots, N\}$. In this context, $h = (R-L)/(N-1)$ denotes the mesh size and N is a chosen positive integer. Readers can refer to Figure 1 for a schematic overview.

Figure 1. Schematic for the discrete domain.

We denote the solution at the discrete grid point (x_i, t_n) by $u_i^n = u(x_i, t_n)$. The temporal nodes are defined as $t_n = (n-1)\Delta t$, and Δt refers to the uniform time increment. Equation (1.4) at t_{k+1} can be approximated by the following quadrature formula:

$$\begin{aligned}\frac{\partial^\alpha u(x_i, t_{n+1})}{\partial t^\alpha} &= \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} w_\alpha^{t_{n+1}}(s) \frac{\partial u(x_i, s)}{\partial s} ds = \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} w_\alpha^{t_{n+1}}(s) \frac{u_i^{p+1} - u_i^p}{\Delta t} ds + R_{\Delta t} \\ &= \sum_{p=1}^{n} \frac{(n-p+1)^{1-\alpha} - (n-p)^{1-\alpha}}{n^{1-\alpha}} \frac{u_i^{p+1} - u_i^p}{\Delta t} + R_{\Delta t} = \sum_{p=1}^{n} w_p^n \frac{u_i^{p+1} - u_i^p}{\Delta t} + R_{\Delta t},\end{aligned} \tag{2.1}$$

where

$$w_p^n = \frac{(n-p+1)^{1-\alpha} - (n-p)^{1-\alpha}}{n^{1-\alpha}}, \tag{2.2}$$

which satisfies the following criterion for arbitrary n and α:

$$\sum_{p=1}^{n} w_p^n = 1. \tag{2.3}$$

Furthermore, $R_{\Delta t}$ is the truncation error:

$$R_{\Delta t} = \frac{1-\alpha}{t_{n+1}^{1-\alpha}} \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} \left(\frac{\partial u(x_i, s)}{\partial s} - \frac{u_i^{p+1} - u_i^p}{\Delta t} \right) \frac{ds}{(t_{n+1} - s)^\alpha}. \tag{2.4}$$

To investigate the truncation error $R_{\Delta t}$, we consider the difference between the exact derivative and the discrete solution. We acknowledge that a fully rigorous bound for the fractional kernel requires a specific convolution-based approach. In this study, we follow the analytical method of [17,18] and use a standard Taylor expansion to provide a formal theoretical estimate. Assuming that the solution satisfies sufficient temporal regularity, $u(\cdot, s) \in C^3([\Delta t, T])$, we conduct a formal truncation error estimate to validate the expected temporal accuracy of the numerical scheme. The exact derivative at any $s \in (t_p, t_{p+1})$ can be expanded as follows. Note that because $|s - t_{p+1/2}| \le \Delta t/2$, the quadratic remainder term is uniformly bounded by $O(\Delta t^2)$:

$$\frac{\partial u}{\partial s}(s) = \frac{\partial u}{\partial s}(t_{p+\frac{1}{2}}) + \frac{\partial^2 u}{\partial s^2}(t_{p+\frac{1}{2}})(s - t_{p+\frac{1}{2}}) + O(\Delta t^2).$$

Similarly, we have

$$\frac{u^{p+1} - u^p}{\Delta t} = \frac{\partial u}{\partial s}(t_{p+\frac{1}{2}}) + O(\Delta t^2).$$

Subtracting the two equations and noting that

$$s - t_{p+1/2} = \frac{1}{2}(2s - t_p - t_{p+1}),$$

we get

$$\begin{aligned} \frac{\partial u}{\partial s} - \frac{u^{p+1} - u^p}{\Delta t} &= \frac{\partial^2 u}{\partial s^2}(t_{p+\frac{1}{2}})(s - t_{p+\frac{1}{2}}) + O(\Delta t^2) \\ &= \frac{1}{2}\frac{\partial^2 u}{\partial s^2}(t_{p+\frac{1}{2}})(2s - t_p - t_{p+1}) + O(\Delta t^2). \end{aligned} \tag{2.5}$$

Substituting this approximation back into $|R_{\Delta t}|$ gives

$$|R_{\Delta t}| = \left| \frac{1-\alpha}{t_{n+1}^{1-\alpha}} \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} \left(\frac{1}{2}\frac{\partial^2 u}{\partial s^2}(t_{p+\frac{1}{2}})(2s - t_p - t_{p+1}) + O(\Delta t^2) \right) \frac{ds}{(t_{n+1} - s)^\alpha} \right|. \tag{2.6}$$

Since $u(\cdot, s) \in C^3([\Delta t, T])$, $|\partial^2 u/\partial s^2|$ is bounded on $[\Delta t, T]$. Let

$$M_u = \max_{s \in [\Delta t, T]} |\partial^2 u/\partial s^2|.$$

Factoring out this constant and noting that the $O(\Delta t^2)$ term contributes to a higher-order error after integration [17], we obtain

$$|R_{\Delta t}| \leq \frac{(1-\alpha)M_u}{2t_{n+1}^{1-\alpha}} \left| \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} \frac{2s - t_p - t_{p+1}}{(t_{n+1} - s)^{\alpha}} ds \right| + O(\Delta t^2)$$
$$\leq C_1 \left| \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} \frac{2s - t_p - t_{p+1}}{(t_{n+1} - s)^{\alpha}} ds \right| + O(\Delta t^2). \tag{2.7}$$

Let $s = t_p + \eta \Delta t$ for $\eta \in [0, 1]$, and then $ds = \Delta t d\eta$ and the term $2s - t_p - t_{p+1}$ simplifies to $(2\eta - 1)\Delta t$. The denominator becomes $(t_{n+1} - s)^{\alpha} = (n + 1 - p - \eta)^{\alpha} \Delta t^{\alpha}$. Substituting these into the summation gives

$$\left| \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} \frac{2s - t_p - t_{p+1}}{(t_{n+1} - s)^{\alpha}} ds \right| = \Delta t^{2-\alpha} \left| \sum_{p=1}^{n} \int_{0}^{1} \frac{2\eta - 1}{(n + 1 - p - \eta)^{\alpha}} d\eta \right|. \tag{2.8}$$

The remaining summation of integrals converges to a finite constant for $0 < \alpha < 1$. Therefore, we have

$$|R_{\Delta t}| \leq C_2 \Delta t^{2-\alpha} + O(\Delta t^2) \leq C \Delta t^{2-\alpha}, \tag{2.9}$$

where C is a positive constant and does not depend on Δt. Thus, the theoretical temporal convergence rate is $O(\Delta t^{2-\alpha})$. We remark that this error estimate is derived under sufficient smoothness assumptions and should be understood as a formal analysis. In reality, time-fractional diffusion equations typically exhibit a weak initial singularity where the time derivative blows up near $t = 0$. Consequently, the theoretical convergence rate of $O(\Delta t^{2-\alpha})$ may not fully hold during the first few time steps on a uniform temporal mesh. In practice, the computational method is applied for $t \geq \Delta t$, where the solution is less affected by the initial weak singularity.

Thus, the substitution of Eq (2.1) into Eq (1.1) leads to the following discrete formulation:

$$\sum_{p=1}^{n} w_p^n \frac{u_i^{p+1} - u_i^p}{\Delta t} = \frac{u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1}}{h^2}. \tag{2.10}$$

Here, we apply zero Dirichlet conditions at both ends of the domain. This setup requires $u_0^{n+1} = 0$ and $u_N^{n+1} = 0$. It is worth noting that if the zero Neumann boundary condition is applied instead of the Dirichlet condition, the computational approach for the normalized time-fractional diffusion model strictly preserves the total mass conservation at every time level.

$$w_n^n \frac{u_i^{n+1} - u_i^n}{\Delta t} + \sum_{p=1}^{n-1} w_p^n \frac{u_i^{p+1} - u_i^p}{\Delta t} = \frac{u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1}}{h^2}, \tag{2.11}$$

which can be rearranged as follows:

$$-\frac{\Delta t}{h^2} u_{i-1}^{n+1} + \left(w_n^n + \frac{2\Delta t}{h^2} \right) u_i^{n+1} - \frac{\Delta t}{h^2} u_{i+1}^{n+1} = w_n^n u_i^n - F_i. \tag{2.12}$$

In this formulation, F_i serves as the source term and is defined as

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

An implicit temporal discretization is used to guarantee numerical stability. Equation (2.12) constitutes a tridiagonal linear system under homogeneous Dirichlet boundary conditions. We use the Thomas algorithm [19, 20] to solve these systems efficiently. This allows us to find the updated solution vector $\mathbf{u}^{n+1}$ by solving the following matrix equation:

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

Here, A is a tridiagonal matrix that enforces the zero Dirichlet conditions at the boundaries $i = 1$ and $i = N$. The specific steps are outlined below.

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

$$\mathbf{u}^{n+1} = \begin{pmatrix} u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_{N-1}^{n+1} \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} w_n^n u_2^n - F_2 \\ w_n^n u_3^n - F_3 \\ \vdots \\ w_n^n u_{N-1}^n - F_{N-1} \end{pmatrix}.$$

For clarity, the physical and computational parameters used in the numerical scheme and the subsequent analysis are summarized in Table A.1 of Appendix A.

To proceed with the stability analysis, we must rewrite the right-hand side (RHS) of Eq (2.12) as a linear combination of past states u_i^p. We define a new set of coefficients C_p^n as follows: $C_p^n = w_p^n - w_{p-1}^n$ for $p = 2, 3, \ldots, n$, and $C_1^n = w_1^n$.

Lemma 1. *The source term $w_n^n u_i^n - F_i$ on the right-hand side of Eq (2.12) is algebraically equivalent to $\sum_{p=1}^n C_p^n u_i^p$.*

Proof. By substituting the definition of $F_i = \sum_{p=1}^{n-1} w_p^n (u_i^{p+1} - u_i^p)$, we expand the RHS expression as follows:

$$w_n^n u_i^n - F_i = w_n^n u_i^n - \sum_{p=1}^{n-1} w_p^n u_i^{p+1} + \sum_{p=1}^{n-1} w_p^n u_i^p.$$

We shift the index in the first summation by letting $k = p + 1$:

$$\sum_{p=1}^{n-1} w_p^n u_i^{p+1} = \sum_{k=2}^{n} w_{k-1}^n u_i^k = \sum_{p=2}^{n} w_{p-1}^n u_i^p.$$

Substituting this shifted index back into the expanded RHS yields:

$$w_n^n u_i^n - F_i = w_n^n u_i^n - \sum_{p=2}^{n} w_{p-1}^n u_i^p + \sum_{p=1}^{n-1} w_p^n u_i^p.$$

We extract the $p = n$ term from the first part and the $p = 1$ term from the second summation to align the indices:

$$w_n^n u_i^n - F_i = w_n^n u_i^n - w_{n-1}^n u_i^n - \sum_{p=2}^{n-1} w_{p-1}^n u_i^p + w_1^n u_i^1 + \sum_{p=2}^{n-1} w_p^n u_i^p.$$

Grouping the coefficients for each corresponding u_i^p term provides:

$$w_n^n u_i^n - F_i = (w_n^n - w_{n-1}^n) u_i^n + \sum_{p=2}^{n-1} (w_p^n - w_{p-1}^n) u_i^p + w_1^n u_i^1.$$

By applying the definition of the newly introduced coefficients C_p^n, we obtain the final exact expression:

$$w_n^n u_i^n - F_i = C_n^n u_i^n + \sum_{p=2}^{n-1} C_p^n u_i^p + C_1^n u_i^1 = \sum_{p=1}^{n} C_p^n u_i^p.$$

This completes the proof. □

To prove that the scheme satisfies the maximum principle and is unconditionally stable, we must examine the properties of these new coefficients C_p^n.

Lemma 2. *The sum of the coefficients is conserved.*

The sum of all coefficients C_p^n forms a telescoping sum:

$$\sum_{p=1}^{n} C_p^n = (w_n^n - w_{n-1}^n) + (w_{n-1}^n - w_{n-2}^n) + \cdots + (w_2^n - w_1^n) + w_1^n = w_n^n.$$

This indicates that the sum of the coefficients of all historical terms on the RHS exactly balances the time-dependent portion of the coefficient of u_i^{n+1} on the left-hand side.

Lemma 3. *The weight term w_p^n is monotonically increasing with respect to p.*

Let us define a function $g(x) = (x+1)^{1-\alpha} - x^{1-\alpha}$ for $x \geq 0$. Because $0 < \alpha < 1$, we take the derivative of $g(x)$:

$$g'(x) = (1 - \alpha)\left[\frac{1}{(x+1)^{\alpha}} - \frac{1}{x^{\alpha}}\right].$$

Because $x + 1 > x$, we have $(x + 1)^{\alpha} > x^{\alpha}$. This leads to $g'(x) < 0$. Therefore, $g(x)$ is a strictly monotonically decreasing function. The weight is defined as $w_p^n = \frac{g(n-p)}{n^{1-\alpha}}$. As the integration time step p increases, $(n-p)$ decreases. Because $g(x)$ is a decreasing function, $g(n-p)$ increases. Thus, we have

$$0 < w_1^n < w_2^n < \cdots < w_{n-1}^n < w_n^n.$$

Lemma 4. *Coefficients C_p^n are positive.*

From Lemma 3, we immediately obtain:

$$C_p^n = w_p^n - w_{p-1}^n > 0 \quad \text{for all} \ \ p \geq 2,$$

and $C_1^n = w_1^n > 0$. Therefore, all historical states u_i^p on the right side of the equation have positive coefficients.

For the coefficient matrix A in the tridiagonal system, the main diagonal element is $w_n^n + 2\Delta t/h^2$, and the non-diagonal elements are $-\Delta t/h^2$. Because $w_n^n > 0$ and $\Delta t > 0$, we obtain:

$$w_n^n + \frac{2\Delta t}{h^2} > \left|-\frac{\Delta t}{h^2}\right| + \left|-\frac{\Delta t}{h^2}\right| = \frac{2\Delta t}{h^2}.$$

This proves that the coefficient matrix A is strictly diagonally dominant. This lemma guarantees that the implicit difference scheme has a unique solution at each time step.

Theorem 1. *The implicit finite difference method is unconditionally stable.*

Proof. We adopt a proof procedure similar to that presented in [17, 21]. Let $\mu = \Delta t/h^2$ and define the infinity norm of the numerical solution at time level n as $\|u^n\|_\infty = \max_i |u_i^n|$. By applying the result of Lemma 1, the implicit difference scheme can be rewritten as:

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

If the maximum absolute value at the $(n+1)$-th time level occurs at the boundary nodes, then $\|u^{n+1}\|_\infty = 0$, and the stability condition $\|u^{n+1}\|_\infty \leq \|u^1\|_\infty$ holds trivially. Suppose that the maximum absolute value occurs at an internal spatial node m, which implies $\|u^{n+1}\|_\infty = |u_m^{n+1}|$. Evaluating the equality at node m for $m = 2, 3, \cdots, N-1$ and taking the absolute value of both sides gives:

$$(w_n^n + 2\mu)|u_m^{n+1}| = \left|\mu u_{m-1}^{n+1} + \mu u_{m+1}^{n+1} + \sum_{p=1}^{n} C_p^n u_m^p\right|.$$

From Lemma 4, we note that $\mu > 0$, $w_n^n > 0$, and $C_p^n > 0$. Applying the triangle inequality, we have:

$$(w_n^n + 2\mu)|u_m^{n+1}| \leq \mu|u_{m-1}^{n+1}| + \mu|u_{m+1}^{n+1}| + \sum_{p=1}^{n} C_p^n |u_m^p|.$$

By the definition of the infinity norm, we have $|u_{m-1}^{n+1}| \leq \|u^{n+1}\|_\infty$, $|u_{m+1}^{n+1}| \leq \|u^{n+1}\|_\infty$, and $|u_m^p| \leq \|u^p\|_\infty$. Substituting these into the inequality yields:

$$(w_n^n + 2\mu)\|u^{n+1}\|_\infty \leq 2\mu\|u^{n+1}\|_\infty + \sum_{p=1}^{n} C_p^n \|u^p\|_\infty.$$

Subtracting $2\mu\|u^{n+1}\|_\infty$ from both sides, we arrive at:

$$w_n^n \|u^{n+1}\|_\infty \leq \sum_{p=1}^{n} C_p^n \|u^p\|_\infty. \tag{2.13}$$

We now use mathematical induction to prove that $\|u^{n+1}\|_\infty \leq \|u^1\|_\infty$ for all $n \geq 1$. For $n = 1$, inequality (2.13) becomes:

$$w_1^1\|u^2\|_\infty \leq C_1^1\|u^1\|_\infty.$$

From the definition of the coefficients, we have $C_1^1 = w_1^1$. Because $w_1^1 > 0$, we divide both sides by w_1^1:

$$\|u^2\|_\infty \leq \|u^1\|_\infty.$$

Thus, the base case is proven. Assume that the condition $\|u^k\|_\infty \leq \|u^1\|_\infty$ holds for all $1 \leq k \leq n$. We then proceed to prove the inequality for the $(n+1)$-th level. Substituting the inductive hypothesis into Eq (2.13), we obtain:

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

According to Lemma 2, the sum of the history coefficients satisfies $\sum_{p=1}^{n} C_p^n = w_n^n$. Substituting this Lemma into the inequality gives:

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

Since $w_n^n > 0$, we conclude:

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

This demonstrates that the numerical solution at any time step is uniformly bounded by the initial state. Therefore, the implicit difference scheme is unconditionally stable. □

Theorem 2. *Suppose that the exact solution $u(x,t)$ of the normalized time-fractional diffusion model satisfies sufficient regularity, $u(\cdot,t) \in C^3([\Delta t, T])$ in time, and $u(x,\cdot) \in C^4(\Omega)$ in space. Let $U_i^n = u(x_i,t_n)$ be the exact solution and u_i^n be the numerical solution computed using the scheme. Let $e_i^n = U_i^n - u_i^n$. Then, there exists a positive constant C^* such that the global error $\|e^n\|_\infty = \max_{1\leq i\leq N-1}|U_i^n - u_i^n|$ satisfies:*

$$\|e^n\|_\infty \leq C^*(\Delta t^{2-\alpha} + h^2), \quad \textit{for all } n \geq 1.$$

Proof. We adopt a proof procedure similar to that presented in [17]. By subtracting the numerical scheme from the exact solution equation and using the algebraic rearrangement established in Lemma 1, we obtain the global error evolution equation:

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

where R_i^{n+1} is the overall local truncation error at node i and time t_{n+1}. According to Eq (2.10), the temporal truncation error is bounded by $C_1\Delta t^{2-\alpha}$. Combined with the error from the standard spatial central difference approximation, which is bounded by C_2h^2, the overall local truncation error satisfies

$\|R^{n+1}\|_\infty \le C(\Delta t^{2-\alpha} + h^2)$, where $C = \max(C_1, C_2)$. For simplicity, we denote $\rho = C(\Delta t^{2-\alpha} + h^2)$. Thus, for any spatial node i, we have $|R_i^{n+1}| \le \rho$.

We assume that the maximum absolute error at the $(n+1)$-th time level occurs at spatial node m, which implies $|e_m^{n+1}| = \|e^{n+1}\|_\infty$. Evaluating the global error evolution equation at node m, we get:

$$(w_n^n + 2\mu)\, e_m^{n+1} = \mu e_{m-1}^{n+1} + \mu e_{m+1}^{n+1} + \sum_{p=1}^{n} C_p^n e_m^p + \Delta t R_m^{n+1}.$$

Taking the absolute value of both sides and applying the triangle inequality, we obtain:

$$(w_n^n + 2\mu)\, |e_m^{n+1}| \le \mu |e_{m-1}^{n+1}| + \mu |e_{m+1}^{n+1}| + \sum_{p=1}^{n} C_p^n |e_m^p| + \Delta t |R_m^{n+1}|.$$

Here, the absolute value signs on the coefficients are omitted because $\mu > 0$, $w_n^n > 0$, and $C_p^n > 0$, as demonstrated in Lemma 4.

By definition of the infinity norm, the errors at the neighboring nodes and the historical time levels satisfy $|e_{m-1}^{n+1}| \le \|e^{n+1}\|_\infty$, $|e_{m+1}^{n+1}| \le \|e^{n+1}\|_\infty$, and $|e_m^p| \le \|e^p\|_\infty$. Substituting these maximum conditions and $|R_m^{n+1}| \le \rho$ into the inequality yields:

$$(w_n^n + 2\mu)\, \|e^{n+1}\|_\infty \le 2\mu \|e^{n+1}\|_\infty + \sum_{p=1}^{n} C_p^n \|e^p\|_\infty + \Delta t \rho.$$

Subtracting $2\mu \|e^{n+1}\|_\infty$ from both sides, we arrive at:

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

We define the sequence $b_j = (j+1)^{1-\alpha} - j^{1-\alpha}$. We rewrite the normalized weight from Eq (2.2) as $w_p^n = b_{n-p}/n^{1-\alpha}$. Consequently, the leading coefficient corresponds to $w_n^n = b_0/n^{1-\alpha}$, and the history coefficients become

$$C_p^n = w_p^n - w_{p-1}^n = (b_{n-p} - b_{n-p+1})/n^{1-\alpha}.$$

We substitute these equivalent expressions into the inequality. We then multiply both sides by the positive scaling factor $n^{1-\alpha}$ to eliminate the denominators, which provides:

$$b_0 \|e^{n+1}\|_\infty \le \sum_{p=1}^{n} \left(b_{n-p} - b_{n-p+1}\right) \|e^p\|_\infty + n^{1-\alpha} \Delta t \rho.$$

We simplify the inequality by changing the summation index to $j = n + 1 - p$, and letting $c_j = b_{j-1} - b_j$. Since $b_0 = 1$, we have:

$$\|e^{n+1}\|_\infty \le \sum_{j=1}^{n} c_j \|e^{n+1-j}\|_\infty + n^{1-\alpha} \Delta t \rho.$$

Notice that the time at the $(n+1)$-th level is $t_{n+1} = n\Delta t \le T$. Then, we have

$$n^{1-\alpha} \Delta t \rho = (n\Delta t)^{1-\alpha} \Delta t^\alpha \rho \le T^{1-\alpha} \Delta t^\alpha \rho.$$

Let $\tilde{\rho} = T^{1-\alpha}\Delta t^{\alpha}\rho$. The inequality becomes:

$$\|e^{n+1}\|_{\infty} \leq \sum_{j=1}^{n} c_j \|e^{n+1-j}\|_{\infty} + \tilde{\rho}.$$

To bound the right-hand side, we now use mathematical induction to prove $\|e^{n+1}\|_{\infty} \leq b_n^{-1}\tilde{\rho}$. For $n = 1$, since $e^1 = 0$, we have:

$$\|e^2\|_{\infty} \leq \tilde{\rho} \leq b_1^{-1}\tilde{\rho},$$

noting that $b_1 < 1$. The base case is proven. Now we assume that the condition $\|e^{k+1}\|_{\infty} \leq b_k^{-1}\tilde{\rho}$ is satisfied for all $k < n$. We then proceed to prove the result for the $n + 1$ level:

$$\|e^{n+1}\|_{\infty} \leq \sum_{j=1}^{n} c_j b_{n-j}^{-1}\tilde{\rho} + \tilde{\rho}.$$

Since b_k is a strictly decreasing sequence, we have $b_{n-j}^{-1} \leq b_n^{-1}$. Using the telescoping property

$$\sum_{j=1}^{n} c_j = 1 - b_n,$$

we obtain:

$$\|e^{n+1}\|_{\infty} \leq b_n^{-1}\tilde{\rho}\sum_{j=1}^{n} c_j + \tilde{\rho} = b_n^{-1}\tilde{\rho}(1 - b_n) + \tilde{\rho} = b_n^{-1}\tilde{\rho}.$$

According to the mean value theorem, $b_n = (n + 1)^{1-\alpha} - n^{1-\alpha} \geq (1 - \alpha)(n + 1)^{-\alpha}$, which yields $b_n^{-1} \leq (n + 1)^{\alpha}/(1 - \alpha)$. We substitute this bound and the definition of $\tilde{\rho} = T^{1-\alpha}\Delta t^{\alpha}\rho$ into the inequality. This substitution allows the Δt^{α} factor to recombine with $(n+1)^{\alpha}$ to form the physical time $((n+1)\Delta t)^{\alpha}$:

$$\|e^{n+1}\|_{\infty} \leq \frac{(n + 1)^{\alpha}}{1 - \alpha} T^{1-\alpha}\Delta t^{\alpha}\rho = \frac{T^{1-\alpha}}{1 - \alpha}((n + 1)\Delta t)^{\alpha}\rho.$$

Notice that the integration time is bounded by $(n + 1)\Delta t \leq T$. We apply this bound to eliminate the temporal step dependency from the spatial error term:

$$\|e^{n+1}\|_{\infty} \leq \frac{T^{1-\alpha}}{1 - \alpha} T^{\alpha}\rho = \frac{T}{1 - \alpha}\rho.$$

Finally, we substitute $\rho = C(\Delta t^{2-\alpha} + h^2)$ back into the inequality. The intermediate Δt^{α} factor has completely vanished, which leaves the spatial error as plain h^2:

$$\|e^{n+1}\|_{\infty} \leq \frac{CT}{1 - \alpha}(\Delta t^{2-\alpha} + h^2).$$

Let $C^* = \frac{CT}{1-\alpha}$. We conclude:

$$\|e^{n+1}\|_{\infty} \leq C^*(\Delta t^{2-\alpha} + h^2).$$

This proves that the computational scheme is unconditionally convergent with a rate of $O(\Delta t^{2-\alpha} + h^2)$. □

3. Computational tests

We conduct several computational tests to validate the performance of the numerical scheme. We investigate the convergence, unconditional stability, and maximum-principle-preserving property of the method through various computational tests. We also study the effect of different fractional orders on the diffusion behavior and compare the numerical results with the theoretical analysis.

3.1. Convergence test

3.1.1. Smooth initial condition

To evaluate the temporal convergence, we carry out numerical tests within spatial domain $\Omega = (0, 1)$ using a fixed grid resolution of $N = 201$. The final computation time is $T = 0.5$, and the fractional orders are chosen as $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. We initialize the system with

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

The discrete L_2-norm is introduced as

$$\|u^k\|_2 = \sqrt{\sum_{i=1}^{N} (u_i^k)^2 / N},$$

where k denotes the time-step index corresponding to the final time T. Thus, the relative error ϵ is computed by

$$\epsilon = \frac{\|u^k - u_{\text{ref}}^k\|_2}{\|u_{\text{ref}}^k\|_2}, \tag{3.2}$$

where u_{ref}^k represents the reference solution generated with a fine time step of $\Delta t = T/2^{13}$. The temporal accuracy is assessed by systematically halving the time step size from $\Delta t = T/2^5$ down to $\Delta t = T/2^8$. The corresponding temporal convergence order is calculated as follows:

$$\text{Order} = \log_2 \left(\frac{\epsilon_j}{\epsilon_{j+1}} \right), \tag{3.3}$$

where ϵ_j and ϵ_{j+1} denote the relative errors evaluated at two successive time steps $\Delta t_j = T/2^j$ and $\Delta t_{j+1} = T/2^{j+1}$ for $j \in \{5, 6, 7\}$. The computed discretization errors and their corresponding convergence rates are detailed in Table 1.

Table 1. Temporal discretization errors and convergence rates.

	$\Delta t = T/2^5$		$\Delta t = T/2^6$		$\Delta t = T/2^7$		$\Delta t = T/2^8$
α	ϵ	Order	ϵ	Order	ϵ	Order	ϵ
0.1	8.0122×10^{-5}	1.89	2.1544×10^{-5}	1.90	5.7821×10^{-6}	1.90	1.5495×10^{-6}
0.3	4.9864×10^{-4}	1.72	1.5131×10^{-4}	1.72	4.5922×10^{-5}	1.72	1.3940×10^{-5}
0.5	1.9121×10^{-3}	1.53	6.6369×10^{-4}	1.52	2.3118×10^{-4}	1.52	8.0613×10^{-5}
0.7	7.3912×10^{-3}	1.33	2.9382×10^{-3}	1.32	1.1760×10^{-3}	1.32	4.7109×10^{-4}
0.9	4.5415×10^{-2}	1.15	2.0477×10^{-2}	1.13	9.3492×10^{-3}	1.13	4.2750×10^{-3}

To check the spatial discretization errors, we fix the time step size at $\Delta t = T/2^8$ to guarantee sufficient temporal accuracy. For this example, we set the fractional order to $\alpha = 0.7$. We compute the reference solution u^k_{ref} on a fine grid with $2^{11} + 1$ points. To check spatial convergence, we test different grid sizes $\mathcal{N} = 2^\tau + 1$ with $\tau \in \{3, 4, 5, 6, 7\}$. As shown in Figure 2, the numerical error drops as we refine the mesh. This perfectly matches what we proved in Theorem 2.

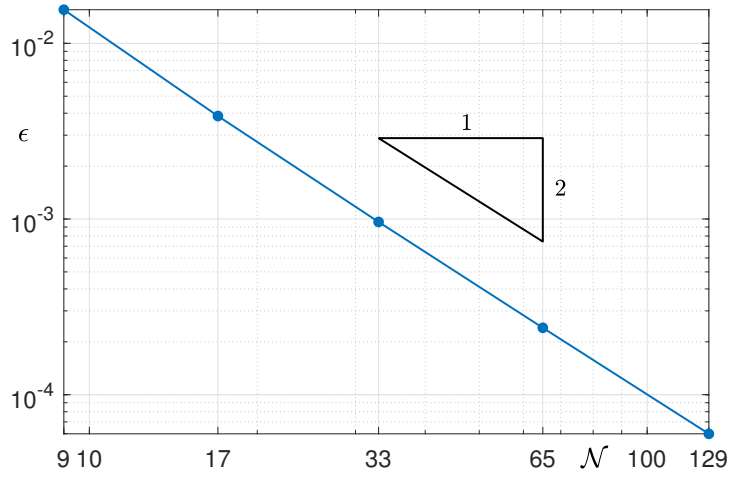

Figure 2. Log-log plot of the spatial error versus grid spacing.

3.1.2. Non-smooth initial condition

To further evaluate the robustness of the numerical scheme and its dependence on spatial regularity, we consider the fractional problem with a non-smooth initial condition. Specifically, we replace the smooth sine function with a standard hat function:

$$u(x, 0) = 1 - 2|x - 0.5|, \quad x \in [0, 1].$$

While this function is continuous, its first derivative is discontinuous at $x = 0.5$. The numerical errors in the L^2-norm and their corresponding convergence rates are tabulated in Table 2 and Figure 3. As expected, the empirical convergence rates exhibit a degradation compared to the smooth case presented in Section 3.1.1.

Table 2. Temporal discretization errors and convergence rates using a non-smooth initial condition.

	$\Delta t = T/2^5$		$\Delta t = T/2^6$		$\Delta t = T/2^7$		$\Delta t = T/2^8$
α	ϵ	Order	ϵ	Order	ϵ	Order	ϵ
0.1	2.8367×10^{-5}	1.84	7.9367×10^{-6}	1.84	2.2107×10^{-6}	1.85	6.1291×10^{-7}
0.3	1.9195×10^{-4}	1.68	5.9995×10^{-5}	1.68	1.8688×10^{-5}	1.69	5.7989×10^{-6}
0.5	7.6946×10^{-4}	1.49	2.7338×10^{-4}	1.50	9.6879×10^{-5}	1.50	3.4214×10^{-5}
0.7	2.6819×10^{-3}	1.30	1.0919×10^{-3}	1.30	4.4305×10^{-4}	1.31	1.7891×10^{-4}
0.9	8.6058×10^{-3}	1.10	4.0266×10^{-3}	1.10	1.8731×10^{-3}	1.12	8.6451×10^{-4}

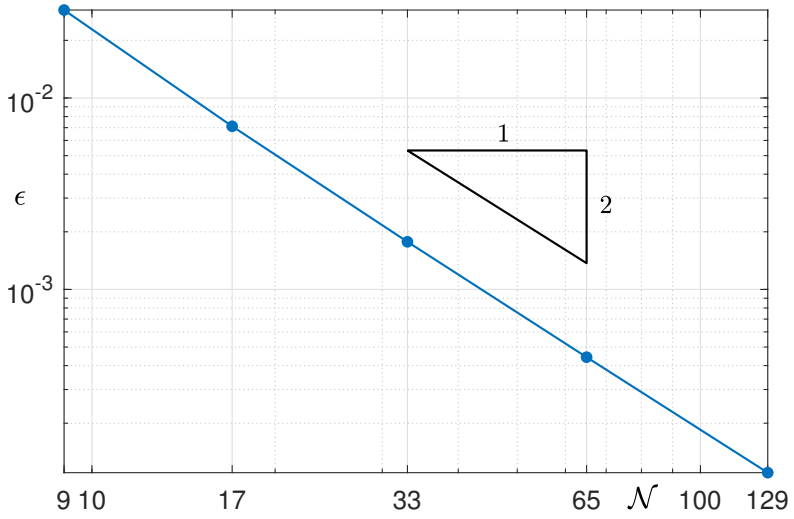

Figure 3. Log-log plot of the spatial error versus grid spacing using a non-smooth initial condition.

3.2. Unconditional stability test

In this test, we solve the time-fractional diffusion equation on the interval $x \in [0, 100]$. The simulation is performed up to a final time of $T = 50$ with a spatial step size of $h = 1/20$. The initial condition is set as

$$u(x, 0) = \begin{cases} 1, & |x - 20| \le 5 \text{ or } |x - 50| \le 5 \text{ or } |x - 80| \le 5, \\ 0, & \text{otherwise.} \end{cases}$$

To examine the impact of the time step, we test three values: $\Delta t = 0.1, 1$, and 10. Figure 4 shows the numerical solutions at $T = 50$ for different fractional orders α. For $\alpha = 0.1$ and $\alpha = 0.5$, the computational results under different time-step sizes almost completely overlap. Even with a notably large time step of $\Delta t = 10$, the numerical solution accurately matches the high-precision baseline calculated with $\Delta t = 0.1$. However, when $\alpha = 0.9$, the result computed with the large time step $\Delta t = 10$ exhibits minor distortion. The numerical profile overestimates the amplitude at the peaks and underestimates it at the valleys. As demonstrated in Theorem 2 and Table 1, at $\alpha = 0.9$, the scheme degenerates to $O(\Delta t^{1.1})$ accuracy. A coarse time step under this near first-order precision fails to capture the rapid initial smoothing of the discontinuous initial values.

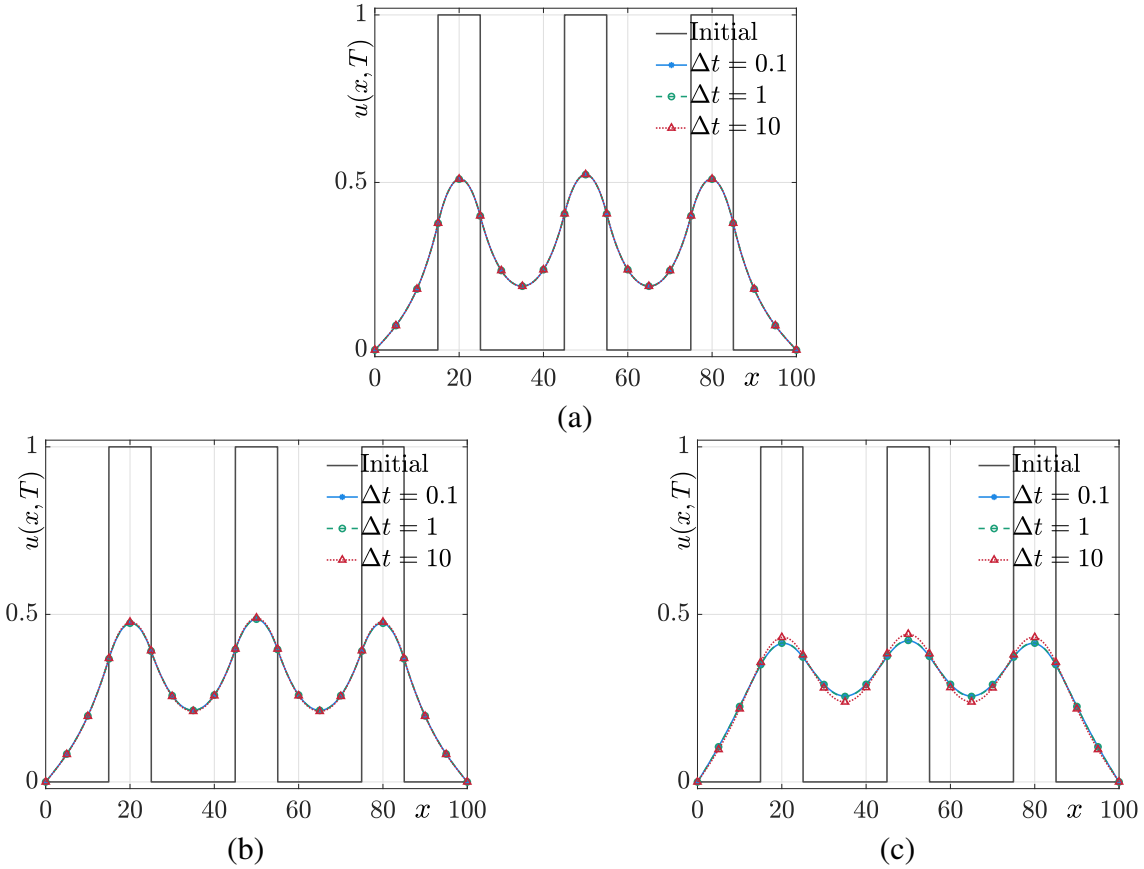

Figure 4. Solution profiles for the time-fractional diffusion equation using various time-step sizes: (a) $\alpha = 0.1$, (b) $\alpha = 0.5$, and (c) $\alpha = 0.9$.

3.3. Maximum-principle-preserving test

In this numerical experiment, we validate the maximum-principle-preserving property by using the step initial condition with random numbers. The spatial discretization settings are identical to those in Section 3.2. The initial condition is set as

$$u(x,0) = \min\Big(v(x) + 0.3\xi(x),\ 1\Big),$$

where

$$v(x) = \begin{cases} 0.7, & \text{if } \left|x - \frac{100}{3}\right| \le \frac{20}{3} \text{ or } \left|x - \frac{200}{3}\right| \le \frac{20}{3}, \\ 0, & \text{otherwise,} \end{cases}$$

and $\xi(x)$ is the random function uniformly distributed in the interval $[0, 1]$.

Figure 5 illustrates the numerical solutions of the time-fractional diffusion model using three different fractional orders $\alpha = 0.1$, $\alpha = 0.5$, and $\alpha = 0.9$. The simulations use a discontinuous initial condition with uniform random noise. To evaluate the robustness of the numerical scheme, three distinct time-step sizes are adopted: $\Delta t = 0.1$, 1, and 10. The agreement among the solutions proves that the implicit scheme remains strictly stable and does not diverge even when the simulation is executed with significantly large time steps. Furthermore, the initial condition fluctuates strictly

within the interval [0, 1]. As the diffusion process evolves, the numerical scheme completely smooths out the high-frequency random noise. The resulting profiles remain strictly within the interval [0, 1] without any non-physical overshoots or undershoots. This result confirms that our numerical method preserves the maximum principle. Note that the development of maximum-principle-preserving schemes has also recently attracted significant attention in other complex nonlinear applications, such as phase-field modeling [22]. We also see slight shifts in the wave shapes and peak heights for different α values. These results are reasonable because each fractional order has different memory effects and diffusion properties. Figure 5d and Table 3 show the max and min values of u under different α values using $\Delta t = 10$.

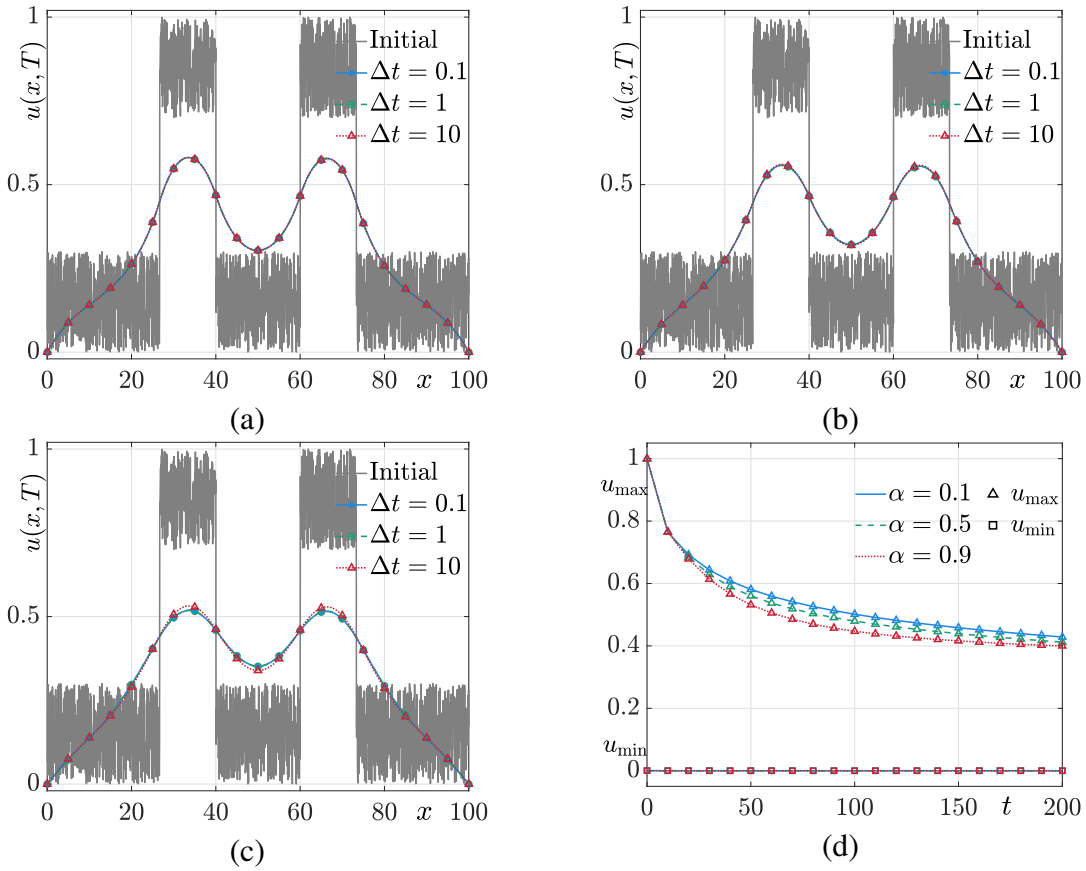

Figure 5. Solution profiles for the time-fractional diffusion equation starting with a random initial condition and using different time-step sizes for (a) $\alpha = 0.1$, (b) $\alpha = 0.5$, (c) $\alpha = 0.9$, and (d) max and min values of u.

Table 3. Maximum values of $u(x,t)$ at different times and fractional orders α. The minimum value remains 0.

		t								
	α	0	25	50	75	100	125	150	175	200
	0.1	0.9997	0.6434	0.5810	0.5262	0.5015	0.4734	0.4584	0.4394	0.4284
max u	0.5	0.9997	0.6293	0.5602	0.5036	0.4796	0.4532	0.4394	0.4220	0.4120
	0.9	0.9997	0.6135	0.5317	0.4697	0.4470	0.4258	0.4162	0.4051	0.4000
min u	-	0	0	0	0	0	0	0	0	0

We use the same initial condition and spatial discretization as those in Section 3.2. In this example, we choose the time step as $\Delta t = 1$ and the final time as $T = 1000$. We also use $u_{\max}$ to represent the maximum value of u.

Figure 6 presents the long-time evolution of the maximum value $u_{\max}$ for the time-fractional diffusion equation. All curves begin from $u_{\max} = 1$. The decay speed strongly depends on the fractional order α. For example, the case with $\alpha = 0.9$ shows the fastest decay and behaves similarly to the classical diffusion model. In contrast, the cases with $\alpha = 0.5$ and $\alpha = 0.1$ show much slower decay. This phenomenon reflects the memory effect in anomalous sub-diffusion. A smaller value of α gives a stronger memory effect. As a result, the dissipation process becomes slower and the relaxation curve shows a heavy-tail behavior.

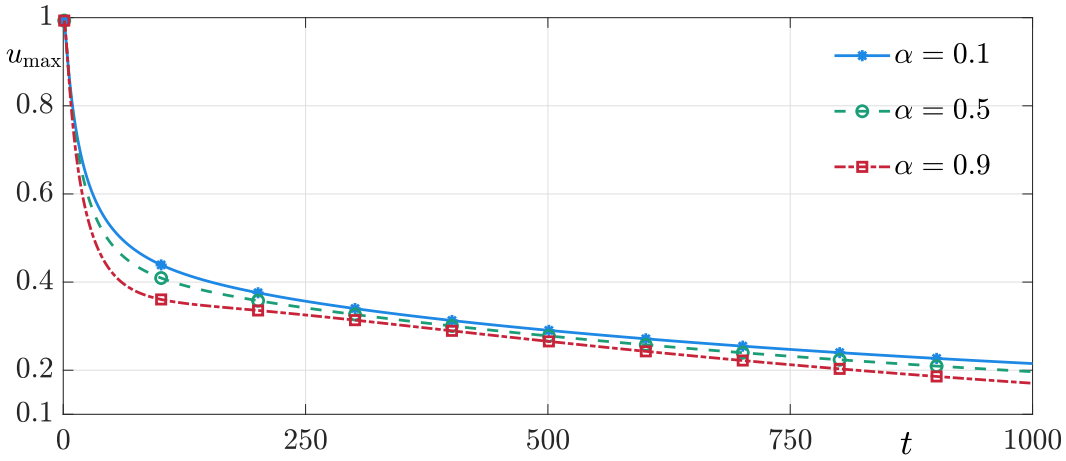

Figure 6. Snapshots of the time-fractional diffusion equation using different time-step sizes with (a) $\alpha = 0.1$, (b) $\alpha = 0.5$, and (c) $\alpha = 0.9$.

3.4. Comparison with the explicit method

To demonstrate the necessity and superiority of the unconditionally stable implicit approach, we construct a fully explicit finite difference scheme for direct comparison. We evaluate the spatial derivative at the known time step n rather than the future time step $n+1$. This evaluation allows the discrete formulation to be rearranged explicitly as follows:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{w_n^n}\left(\frac{u_{i-1}^n - 2u_i^n + u_{i+1}^n}{h^2} - \frac{F_i}{\Delta t}\right). \tag{3.4}$$

In this experiment, we apply the same initial condition and experimental settings described in Section 3.2. We reduce the number of spatial nodes N from 2001 to 201 and fix $\alpha = 0.9$. For the explicit scheme, we use $\Delta t_{\text{explicit}} = 0.01$ to guarantee stability. In contrast, for the unconditionally stable implicit scheme, we set $\Delta t_{\text{implicit}} = 0.1$.

Figure 7 demonstrates that the two schemes yield almost identical results despite their different time-step sizes. However, the explicit scheme required 4.595 seconds to compute, whereas the implicit scheme required only 0.053 second. Notice that the implicit scheme could be substantially faster due to the unconditional stability which allows unrestricted choice of Δt.

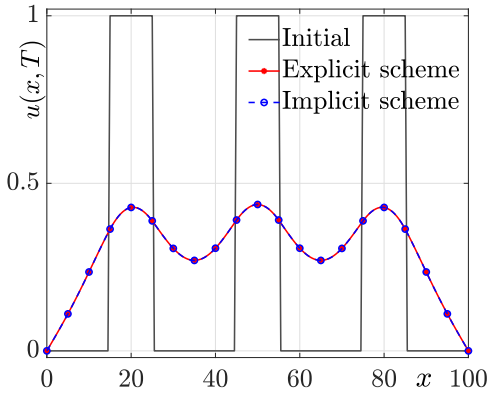

Figure 7. Comparison of the numerical solutions computed by the explicit and implicit finite difference schemes for the normalized time-fractional diffusion equation at final time T.

4. Conclusions

In this paper, we presented the mathematical and numerical analysis of an FDM for a normalized time-fractional diffusion equation. The derivative uses a weighting function whose integral is always equal to one for all fractional orders and times. This property makes it possible to compare diffusion processes under different fractional orders in a fair way. We considered an FDM with an implicit time scheme for the normalized equation. The numerical method produces a tridiagonal linear system. We solve this system by the Thomas algorithm. Theoretical analysis shows that the scheme is unconditionally stable, preserves the maximum principle, and converges correctly. The convergence order is $O(\Delta t^{2-\alpha})$ in time and $O(h^2)$ in space. Numerical results agree well with the theoretical analysis. The numerical tests validate the stability and convergence of the method. They also show the effect of different fractional orders on the diffusion process. The normalized model provides a useful approach for studying time-fractional diffusion problems with consistent weights. In future work, we will extend this approach to higher-dimensional models, nonlinear problems, and variable-order equations. Additionally, to completely resolve the impact of the initial weak singularity near $t = 0$, we plan to incorporate non-uniform graded time meshes [23] or initial correction techniques to maintain the optimal convergence order from the first time step.

Author contributions

Xinpei Wu: writing-original draft, methodology, software, formal analysis; Ke Zhang: validation, investigation, formal analysis, writing-review and editing; Juho Ma: visualization, investigation, methodology, writing-review and editing; Junseok Kim: conceptualization, supervision, project administration, funding acquisition, writing-review and editing. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

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

Acknowledgments

This work was supported by the National Research Foundation (NRF), Korea, under project BK21 FOUR. The authors would like to thank the anonymous reviewers for their valuable comments and suggestions, which helped improve the quality of this manuscript.

Conflict of interest

The authors declare they have no conflicts of interest.

References

1. F. Effenberger, S. Aerdker, L. Merten, H. Fichtner, Superdiffusion of energetic particles at shocks: a fractional diffusion and Lévy flight model of spatial transport, *Astron. Astrophys.*, **686** (2024), A219. https://doi.org/10.1051/0004-6361/202449334
2. R. Shah, M. Amjad, M. Younis, M. Öztürk, A. Büyükkaya, Analytical and numerical study of nonlinear variable-order time fractional reaction-diffusion coupled equations arising in biological and chemical processes, *Fractal Fract.*, **10** (2026), 151. https://doi.org/10.3390/fractalfract10030151
3. X. Zheng, J. Cheng, H. Wang, Uniqueness of determining the variable fractional order in variable-order time-fractional diffusion equations, *Inverse Probl.*, **35** (2019), 125002. https://doi.org/10.1088/1361-6420/ab3aa3
4. J. Hong, B. Jin, Y. Kian, Identification of a spatially-dependent variable order in one-dimensional subdiffusion, *SIAM J. Math. Anal.*, **57** (2025), 1315–1341. https://doi.org/10.1137/24M1673371
5. 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
6. J. Kim, A normalized Caputo-Fabrizio fractional diffusion equation, *AIMS Math.*, **10** (2025), 6195–6208. https://doi.org/10.3934/math.2025282
7. S. Bera, M. Sen, S. Nath, Efficient numerical simulation of variable-order fractional diffusion processes with a memory kernel, *J. Comput. Sci.*, **92** (2025), 102705. https://doi.org/10.1016/j.jocs.2025.102705

8. J. Cheng, Q. Xia, J. Kim, Y. Li, Decoupled, efficient and structure-preserving numerical scheme for the non-isothermal phase field sintering model, *Comput. Math. Appl.*, **196** (2025), 49–63. https://doi.org/10.1016/j.camwa.2025.07.007

9. Y. Li, Z. Lv, Q. Xia, On the unconditionally stable phase field model of ternary components system considering mass conservation and liquid-solid phase transition, *J. Comput. Phys.*, **543** (2025), 114404. https://doi.org/10.1016/j.jcp.2025.114404

10. F. Zhu, K. Sun, G. Zhang, J. Yang, Diffuse-interface modeling and energy-stable numerical framework for the heat transfer-coupled two-phase fluids in contact with solids, *J. Comput. Phys.*, **524** (2025), 113699. https://doi.org/10.1016/j.jcp.2024.113699

11. J. Yang, J. Kim, On a two-phase incompressible diffuse interface fluid model with curvature-dependent mobility, *J. Comput. Phys.*, **525** (2025), 113764. https://doi.org/10.1016/j.jcp.2025.113764

12. A. Kanwal, S. Boulaaras, R. Shafqat, B. Taufeeq, M. ur Rahman, Explicit scheme for solving variable-order time-fractional initial boundary value problems, *Sci. Rep.*, **14** (2024), 5396. https://doi.org/10.1038/s41598-024-55943-4

13. J. Jia, C. Jiang, Y. Li, M. Liu, W. Qiu, Variable-order fractional wave equation: analysis, numerical approximation, and fast algorithm, *Commun. Nonlinear Sci. Numer. Simul.*, **158** (2026), 109786. https://doi.org/10.1016/j.cnsns.2026.109786

14. 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

15. 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

16. J. Kim, Maximum principle preserving unconditionally stable method for the Allen–Cahn equation with a high-order potential, *Electron. Res. Arch.*, **33** (2025), 433–446. https://doi.org/10.3934/era.2025021

17. Y. Lin, C. Xu, Finite difference/spectral approximations for the time-fractional diffusion equation, *J. Comput. Phys.*, **225** (2007), 1533–1552. https://doi.org/10.1016/j.jcp.2007.02.001

18. W. Zhao, D. Guo, J. Li, Q. Zhao, Cubic spline rule to compute hypersingular integral on a circle, *Electron. Res. Arch.*, **34** (2026), 3008–3023. https://doi.org/10.3934/era.2026136

19. J. W. Thomas, *Numerical partial differential equations: finite difference methods*, Springer, 1995. https://doi.org/10.1007/978-1-4899-7278-1

20. 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

21. Z. Z. Sun, G. Gao, *Fractional differential equations: finite difference methods*, Boston De Gruyter, 2020. https://doi.org/10.1515/9783110616064

22. S. Su, J. Yang, Maximum principle-preserving, unconditionally energy-stable, and convergent method with second-order accuracy for the phase-field model of image inpainting, *Comput. Math. Appl.*, **183** (2025), 32–45. https://doi.org/10.1016/j.camwa.2025.01.032

23. M. Stynes, E. O'Riordan, J. L. Gracia, Error analysis of a finite difference method on graded meshes for a time-fractional diffusion equation, *SIAM J. Numer. Anal.*, **55** (2017), 1057–1079. https://doi.org/10.1137/16M1082329

Appendix

Appendix A. List of algorithmic parameters

For better readability, a comprehensive list and detailed descriptions of all computational parameters and intermediate variables used in this scheme are compiled in Table A.1.

Table A.1. Parameters of the finite difference method.

Parameter	Description
α	Order of the time-fractional derivative ($0 < \alpha < 1$).
L, R	Left boundary and right boundary of the one-dimensional spatial domain $\Omega = (L, R)$.
N	Total number of spatial grid points.
h	Spatial mesh size, calculated as $h = (R - L)/(N - 1)$.
Δt	Time-step size.
T	Final computation time.
x_i	Discrete spatial nodes defined as $x_i = L + (i - 1)h$.
t_n	Temporal nodes defined as $t_n = (n - 1)\Delta t$.
u_i^n	Numerical solution at the discrete grid point (x_i, t_n).
$u_0(x)$	Given initial condition.
w_p^n	Discrete normalized weights.
μ	Grid ratio parameter defined as $\mu = \Delta t/h^2$.
F_i	Source term at node i, defined as $F_i = \sum_{p=1}^{n-1} w_p^n (u_i^{p+1} - u_i^p)$.
C_p^n	Rearranged historical state coefficients, defined as $C_p^n = w_p^n - w_{p-1}^n$ for $p \geq 2$, and $C_1^n = w_1^n$.

Appendix B. MATLAB code for the numerical model

The following listing presents the MATLAB code for Figure 4a.

Listing 1. MATLAB code for a normalized time-fractional diffusion model.

```
% Spatial domain and grid parameters
L=100; Naa=2001; h1=L/Naa; % spatial length; number of nodes; step size
xa=linspace(0,L,Naa);      % spatial grid points
ns=(Naa-1)/20; T=50;       % interval for plot markers; total time
% Initial condition setup (u0)
u0 = zeros(Naa,1);
% Set square wave initial values within three specific intervals
u0(abs(xa-L/5)≤L/20) = 1;
u0(abs(xa-L/2)≤L/20) = 1;
u0(abs(xa-4*L/5)≤L/20) = 1;
% Off-diagonal coefficients
```

```
a1=(-1/h1^2)*ones(Naa,1); c3=a1; % lower diagonal; upper diagonal
% Plot initialization and initial state plotting
figure(1); clf; box on; grid on; hold on;
lwd=1.5; fs=25; axis([0 L -0.02 1.02])
set(gcf, 'position', [100 100 700 550])
xticks([0 L/5 2*L/5 3*L/5 4*L/5 L])
yticks([-0.2 0 0.5 1 1.2]); clr=[0.3 0.3 0.3];
plot(xa, u0, 'color',clr, 'LineWidth', lwd);
% Time step test array and fractional derivative order
dt_st = 1000*[0.0001, 0.001, 0.01]; % Test three different time steps
alp = 0.1;                          % Fractional derivative order (alpha)
% Legend and plot style setup
legendLabels = cell(1, length(dt_st)+1);
legendLabels{1}='Initial';
line_styles = {'*-', 'o--', '^:'};
colors=[0.1176, 0.5333, 0.8980;
0.1059, 0.6196, 0.4667;
0.7843, 0.1373, 0.2235; ];
% Iterate through different time steps for solution validation
for bb = 1:length(dt_st)
dt = dt_st(bb);         % Current time step
Nt = round(T/dt);       % Total number of time steps
uud=zeros(Naa,Nt+1);    % Initialize the solution matrix
uud(:,1)=u0;            % Assign initial condition
% Time advancement loop
for n = 1:Nt
deno = n^(1-alp);
% Calculate discrete weights for the fractional derivative
for p = 1:n
w(p) = ((n+1-p)^(1-alp)-(n-p)^(1-alp))/deno;
end
Ff = zeros(Naa, 1); % Initialize the memory term
% Accumulate contributions from historical time steps
if n > 1
for p = 1:n-1
Ff = Ff+w(p)*(uud(:,p+1)-uud(:,p))/dt;
end
end
% Construct main diagonal elements
for i = 1:Naa
d4(i) = w(n)/dt+2.0/h1^2;
end
% Construct the right-hand-side term of the linear equation system
```

```
    f = w(n)/dt*uud(:,n)-Ff;
    % Use the Thomas algorithm to solve the tridiagonal linear system
    uud(2:Naa-1,n+1)=thomas(a1(2:Naa-1),d4(2:Naa-1),c3(2:Naa-1),f(2:Naa-1));
    end
    plot(xa, uud(:, end), line_styles{bb}, 'MarkerIndices', ...
    1:ns:length(xa), 'Color', colors(bb, :), 'LineWidth', lwd);
    legendLabels{bb+1} = sprintf('\\Delta t = %.1f', dt);
    end
    % Finalize chart elements
    legend(legendLabels, 'Location', 'northwest', 'fontsize', fs, 'Box', 'off');
    xlabel('x'); ylabel('u(x, T)');
    set(gca, 'fontsize', fs);
    %% Function
    % Thomas Algorithm
    function xu = thomas(ap, bt, gm, fc)
    n0 = length(fc);
    % Elimination process (forward elimination)
    for i = 2:n0
    mult = ap(i) / bt(i - 1);
    bt(i) = bt(i) - mult * gm(i - 1);
    fc(i) = fc(i) - mult * fc(i - 1);
    end
    % Back substitution process
    xu(n0) = fc(n0) / bt(n0);
    for i = n0-1:-1:1
    xu(i) = (fc(i) - gm(i) * xu(i + 1)) / bt(i);
    end
    xu = xu'; % Transpose to a column vector and return
    end
```

AIMS Press