AIMS Mathematics, 9(10): 26671–26687.
DOI: 10.3934/math.20241297
Received: 05 July 2024
Revised: 02 September 2024
Accepted: 11 September 2024
Published: 13 September 2024

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

*Research article*

# Numerical investigation of the dynamics for a normalized time-fractional diffusion equation

**Chaeyoung Lee[1], Yunjae Nam[2], Minjoon Bang[2], Seokjun Ham[3] and Junseok Kim[3,*]**

[1] Department of Mathematics, Kyonggi University, Suwon, 16227, Republic of Korea
[2] Program in Actuarial Science and Financial Engineering, Korea University, Seoul 02841, Republic of Korea
[3] Department of Mathematics, Korea University, Seoul, 02841, Republic of Korea

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

**Abstract:** In this study, we proposed a normalized time-fractional diffusion equation and conducted a numerical investigation of the dynamics of the proposed equation. We discretized the governing equation by using a finite difference method. The proposed normalized time-fractional diffusion equation features a different time scale compared to the conventional time-fractional diffusion equation. This distinct time scale provides an intuitive understanding of the fractional time derivative, which represents a weighted average of the temporal history of the time derivative. Furthermore, the sum of the weight function is one for all values of the fractional parameter and time. The primary advantage of the proposed model over conventional time-fractional equations is the unity property of the sum of the weight function, which allows us to investigate the effects of the fractional order on the evolutionary dynamics of time-fractional equations. To highlight the differences in performance between the conventional and normalized time-fractional diffusion equations, we have conducted several numerical experiments.

**Keywords:** normalized time-fractional diffusion equation; finite difference method; Gamma function
**Mathematics Subject Classification:** 35R11, 80M20, 39A14

## 1. Introduction

In recent decades, time-fractional diffusion equations have received significant attention across various fields. Time-fractional derivatives extend classical models by incorporating memory effects, which better capture real-world processes where the rate of change is not constant [1]. They are particularly useful for modeling phenomena with irregular or non-standard diffusion behaviors, such as in heterogeneous media [2], complex fluids [3], or biological systems [4]. These time-fractional

derivatives allow for a more accurate representation of systems with long-range dependencies, thus providing deeper insights and improved predictions in various scientific and engineering fields. Some specific applications of time-fractional derivatives are as follows. The time-fractional Burgers equation, applying a fractional differential method to the classical Burgers equation [5, 6], was used to model a range of physical processes such as turbulence, shock waves, and traffic flow, incorporating memory effects over time. In biology, the time-fractional method was applied to the reaction-diffusion equation, specifically Fisher's equation, which models species propagation [7, 8]. In the field of finance, European and double-barrier options were evaluated using the time-fractional Black–Scholes equation [9–11]. In particular, for double barrier options, the closer the underlying asset price is to the lower barrier, the more the Black–Scholes model tends to overestimate the option's value. Furthermore, a smaller $\alpha$ exacerbates this price bias. The reaction-diffusion equation, known as the Allen–Cahn (AC) equation, finds applications in many fields such as physics, materials science, and biology. By applying the time-fractional method to the AC equation, these approaches accurately model complex dynamical systems by incorporating memory effects and history dependence associated with time [12, 13].

There have been many numerical methods for the time-fractional diffusion equations [14–16]. For simplicity of exposition, let us consider the following one-dimensional conventional time-fractional diffusion equation:

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} = \frac{\partial^2 u(x,t)}{\partial x^2} \quad \text{for } (x,t) \in \Omega \times (0,\ \infty), \tag{1}$$

$$u(x,0) = u_0(x), \quad x \in \Omega, \tag{2}$$

$$u(0,t) = u(1,t) = 0, \quad t \geq 0, \tag{3}$$

where $u(x,t)$ is the concentration at $x$ and $t$, and $u_0(x)$ is the initial condition,

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{\partial u(x,s)}{\partial s} \frac{ds}{(t-s)^\alpha}, \quad 0 < \alpha < 1, \tag{4}$$

where $\Gamma(z) = \int_0^\infty \tau^{z-1} e^{-\tau} d\tau$ is the gamma function. It is noted that when $\alpha = 1$, Eq (1) reduces to the conventional diffusion equation [17]. Let us define a weight function $w_\alpha^t(s)$ as follows:

$$w_\alpha^t(s) = \frac{1}{\Gamma(1-\alpha)(t-s)^\alpha}. \tag{5}$$

Then, Eq (4) can be rewritten as

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} = \int_0^t w_\alpha^t(s) \frac{\partial u(x,s)}{\partial s} ds, \quad 0 < \alpha < 1. \tag{6}$$

For different values of $\alpha = 0.1, 0.5$, and $0.9$, the weight functions $w_\alpha^t(s)$ at $t = 1$ are illustrated in Figure 1(a). It can be seen that the weight functions $w_\alpha^t(s)$ remain flat for small $\alpha$ and show a sharp transition near time $t$ for large $\alpha$. Figure 1(b) shows the weight functions for different time values of $t = 0.5,\ 1$, and 2 with $\alpha = 0.5$. As $t$ increases, the functions simply translate to the right direction.

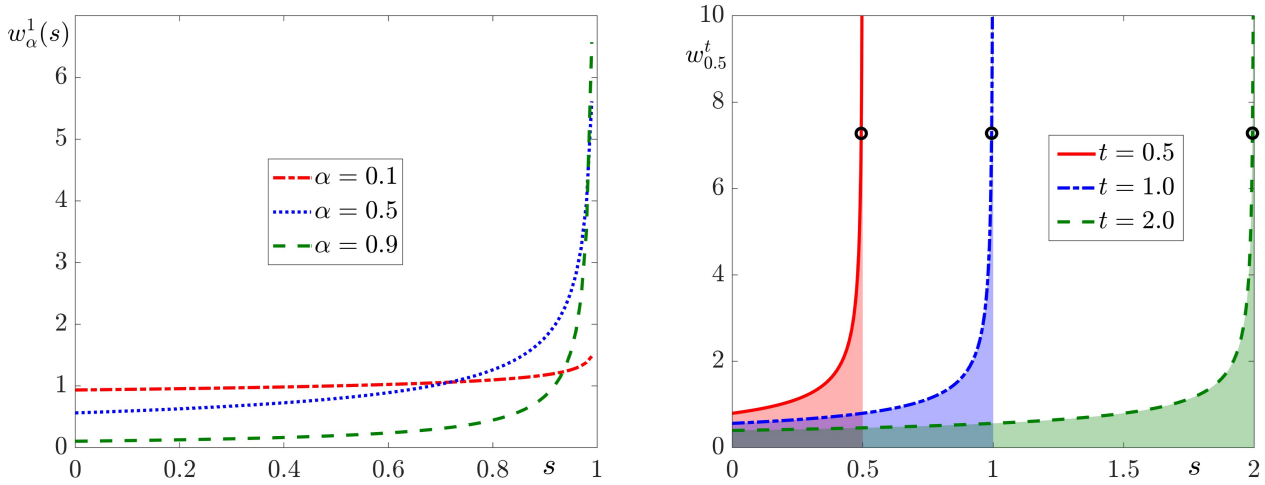



**Figure 1.** (a) Weight functions for different values of $\alpha = 0.1,\ 0.5$, and 0.9 at $t = 1$. (b) Weight functions for different time values of $t = 0.5,\ 1$, and 2 with $\alpha = 0.5$. Here, the circles are points of $(t - 0.001, w_\alpha^t(t - 0.001))$.

We note that $\int_0^t w_\alpha^t(s)ds$ approaches infinity as $t$ increases, for any values of $0 < \alpha < 1$. That is,

$$W_\alpha(t) = \int_0^t w_\alpha^t(s)ds = \int_0^t \frac{1}{\Gamma(1-\alpha)(t-s)^\alpha}ds = \frac{t^{1-\alpha}}{\Gamma(2-\alpha)}, \tag{7}$$

which approaches infinity as $t$ increases, for any values of $0 < \alpha < 1$. From Eq (7), we can see there are scaling differences associated with values of $\alpha$ when comparing the effects of $\alpha$ on the dynamics of the time-fractional diffusion equations because $W_\alpha(t)$ depends on both $\alpha$ and time $t$. This is physically sound and can be inferred intuitively from previous studies [18, 19] related to wave propagation and diffusion problems.

To resolve these scaling differences associated with $\alpha$ values, we propose a normalized time-fractional diffusion equation and conduct numerical investigations of the dynamics of the proposed equation. In this study, we propose the following normalized time-fractional diffusion equation:

$$\frac{\partial^\beta u(x,t)}{\partial t^\beta} = \frac{\partial^2 u(x,t)}{\partial x^2} \quad \text{for } (x,t) \in \Omega \times (0,\ \infty), \tag{8}$$

$$u(x,0) = u_0(x), \quad x \in \Omega, \tag{9}$$

$$u(0,t) = u(1,t) = 0, \quad t \geq 0, \tag{10}$$

where

$$\frac{\partial^\beta u(x,t)}{\partial t^\beta} = \frac{1-\beta}{t^{1-\beta}} \int_0^t \frac{\partial u(x,s)}{\partial s} \frac{ds}{(t-s)^\beta}, \quad 0 < \beta < 1, \tag{11}$$

where $(1-\beta)/t^{1-\beta}$ is the normalizing factor, which makes the right-hand side term in Eq (11) $\partial u/\partial x$ when $\partial u/\partial x$ is constant. That is

$$\frac{1-\beta}{t^{1-\beta}} \int_0^t \frac{ds}{(t-s)^\beta} = 1, \quad 0 < \beta < 1. \tag{12}$$

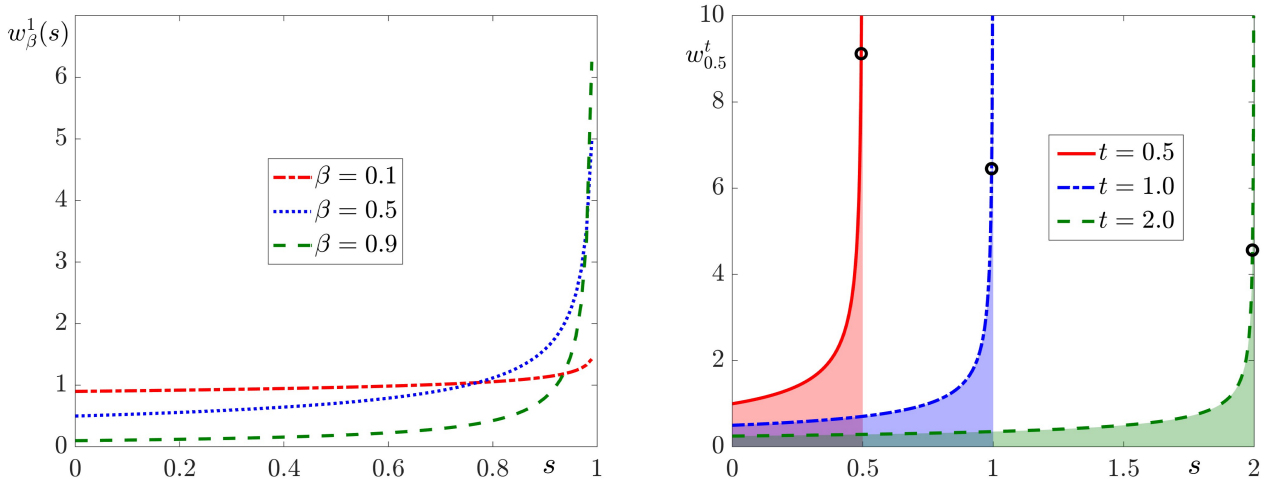
Let us define a weight function $w_\beta^t(s)$ as follows:

$$w_\beta^t(s) = \frac{1-\beta}{t^{1-\beta}(t-s)^\beta}. \tag{13}$$

Then, from Eq (12), we have

$$W_\beta(t) = \int_0^t w_\beta^t(s)ds = 1, \tag{14}$$

which is independent of the fractional order $\beta$ and time $t$, unlike that of the conventional time-fractional derivative, $W_\alpha(t) = t^{1-\alpha}/\Gamma(2-\alpha)$. To the authors' knowledge, this is the first time that the normalized time-fractional diffusion equation is proposed, where the total integration of the weight function is always one for all time-fractional orders and times. For $\beta$ values of 0.1, 0.5, and 0.9, the weight functions $w_\beta^t(s)$ at $t = 1$ are as shown in Figure 2(a). We can observe that the weight functions $w_\beta^t(s)$ are flat when $\beta$ is small and exhibit a sharp transition near time $t$ when $\beta$ is large. Figure 2(b) shows the weight functions for different times $t = 0.5,\ 1$, and 2 with $\beta = 0.5$.




(a) (b)

**Figure 2.** (a) Weight functions for different $\beta$ values with $t = 1$. Here, $\beta = 0.1,\ 0.5$, and 0.9 are used. (b) Weight functions for different time $t$ values with $\beta = 0.5$. Here, $t = 0.5,\ 1$, and 2 are used. Here, the circles are points of $(t - 0.001, w_\beta^t(t - 0.001))$.

Figure 3 shows the temporal evolutions of $W_\alpha(t) = \int_0^t w_\alpha^t(s)ds = t^{1-\alpha}/\Gamma(2-\alpha)$ for $\alpha = 0.1,\ 0.5$, and 0.9. $W_\alpha(t)$ is an increasing function with respect to time $t$ for a fixed fractional order $\alpha$. At early times, $W_\alpha(t)$ increases with respect to the fractional order $\alpha$, whereas at later times, $W_\alpha(t)$ decreases with respect to the fractional order $\alpha$ for a fixed time $t$. However, $W_\beta(t)$ is independent of the fractional order $\beta$ and time $t$.

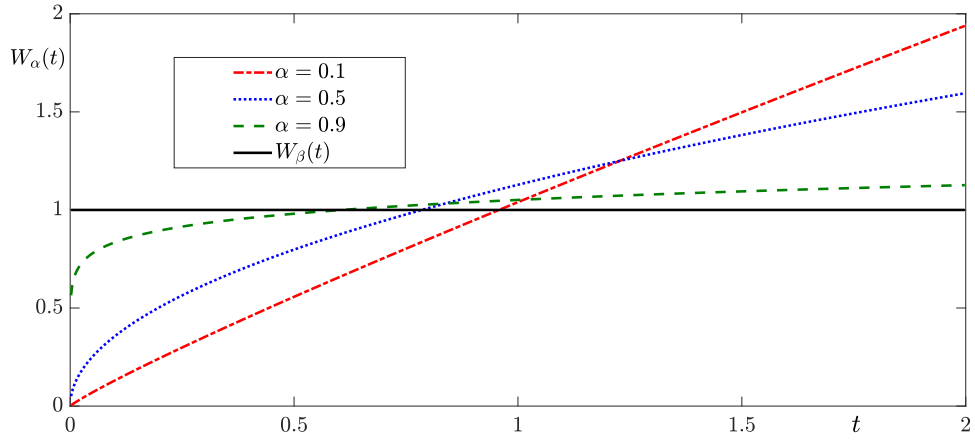



**Figure 3.** Temporal evolutions of $W_\alpha(t) = \int_0^t w_\alpha^t(s)ds$ for $\alpha = 0.1,\ 0.5$, and 0.9. Here, $W_\beta(t) = 1$.

The contents of this paper are as follows. In Section 2, numerical solution algorithms for the conventional and normalized time-fractional diffusion equations are presented. In Section 3, numerical experiments are provided. Finally, Section 4 presents conclusion and potential progress for future study.

## 2. Numerical solutions

### *2.1. Conventional time-fractional diffusion equation*

Let $\Omega = (L_x,\ R_x)$ be the computational domain, which is discretized as follows: $\Omega_h = \{x_i | x_i = L_x + (i-1)h,\ i = 1, \ldots, N_x\}$, where $h = (R_x - L_x)/(N_x - 1)$ for some positive integer $N_x$, see Figure 4.

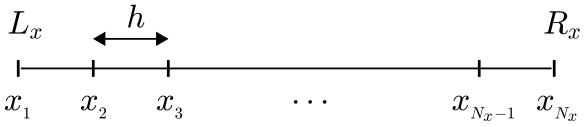



**Figure 4.** Discrete domain.

Let $u_i^n = u(x_i,\ t_n)$ and $t_n = (n-1)\Delta t$, where $\Delta t$ is the time step. Equation (4) can be approximated by the following numerical quadrature formula:

$$\begin{aligned}\frac{\partial^\alpha u(x_i, t_{n+1})}{\partial t^\alpha} &= \frac{1}{\Gamma(1-\alpha)} \sum_{p=1}^{n} \int_{t_p}^{t_{p+1}} \frac{\partial u(x_i, s)}{\partial s} \frac{ds}{(t_{n+1}-s)^\alpha} \\ &\approx \sum_{p=1}^{n} \frac{1}{\Gamma(1-\alpha)} \int_{t_p}^{t_{p+1}} \frac{ds}{(t_{n+1}-s)^\alpha} \frac{u_i^{p+1} - u_i^p}{\Delta t} \qquad (15) \\ &= \sum_{p=1}^{n} \frac{(n+1-p)^{1-\alpha} - (n-p)^{1-\alpha}}{(\Delta t)^{\alpha-1}\Gamma(2-\alpha)} \frac{u_i^{p+1} - u_i^p}{\Delta t}, \qquad (16)\end{aligned}$$

where we have used the identity $(1-\alpha)\Gamma(1-\alpha) = \Gamma(2-\alpha)$ and approximated $\partial u(x_i, s)/\partial s$ over the interval $[t_p,\ t_{p+1}]$ using the finite difference $(u_i^{p+1} - u_i^p)/\Delta t$ in Eq (15). Therefore, we have the following

finite difference discretization of Eq (1) using Eq (16):

$$\sum_{p=1}^{n} w_p^n \frac{u_i^{p+1} - u_i^p}{\Delta t} = \frac{u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1}}{h^2}, \tag{17}$$

where

$$w_p^n = \frac{(n+1-p)^{1-\alpha} - (n-p)^{1-\alpha}}{(\Delta t)^{\alpha-1}\Gamma(2-\alpha)}. \tag{18}$$

Here, we use the zero Dirichlet boundary condition: $u_0^{n+1} = 0$ and $u_{N_x}^{n+1} = 0$. Then, Eq (17) can be rewritten as follows:

$$w_n^n \frac{u_i^{n+1} - u_i^n}{\Delta t} + \sum_{p=1}^{n-1} w_p^n \frac{u_i^{p+1} - u_i^p}{\Delta t} = \frac{u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1}}{h^2}, \tag{19}$$

which can be rearranged as follows:

$$-\frac{1}{h^2}u_{i-1}^{n+1} + \left(\frac{w_n^n}{\Delta t} + \frac{2}{h^2}\right)u_i^{n+1} - \frac{1}{h^2}u_{i+1}^{n+1} = \frac{w_n^n}{\Delta t}u_i^n - \sum_{p=1}^{n-1} w_p^n \frac{u_i^{p+1} - u_i^p}{\Delta t}. \tag{20}$$

Note that an implicit temporal discretization is used for stability. Although fully explicit schemes are generally sufficient in terms of stability and efficiency for second-order partial differential equations, as indicated in [20, 21], an implicit scheme is necessary due to the presence of the source term, $-\sum_{p=1}^{n-1} w_p^n(u_i^{p+1} - u_i^p)/\Delta t$, in Eq (20).

Equation (20) is a tridiagonal system with a zero Dirichlet boundary condition and we can use the Thomas algorithm [22] to efficiently solve this system. Thus, the solution vector $\mathbf{u}^{n+1}$ can be found by solving the tridiagonal system using the Thomas algorithm:

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

where $A$ is a tridiagonal matrix with zero Dirichlet at $i = 1$ and $i = N_x$. The detailed procedure is provided below.

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

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

where $F = \sum_{p=1}^{n-1} w_p^n(u_i^{p+1} - u_i^p)/\Delta t$.

*2.2. Normalized time-fractional diffusion equation*

Equation (11) can be approximated by the following numerical quadrature formula:

$$\begin{aligned}\frac{\partial^\beta u(x_i,t_{n+1})}{\partial t^\beta} &= \frac{1-\beta}{t_{n+1}^{1-\beta}}\sum_{p=1}^{n}\int_{t_p}^{t_{p+1}}\frac{\partial u(x_i,s)}{\partial s}\frac{ds}{(t_{n+1}-s)^\beta}\\ &\approx \sum_{p=1}^{n}\frac{1-\beta}{t_{n+1}^{1-\beta}}\int_{t_p}^{t_{p+1}}\frac{ds}{(t_{n+1}-s)^\beta}\frac{u_i^{p+1}-u_i^p}{\Delta t}\\ &= \sum_{p=1}^{n}\frac{(n+1-p)^{1-\beta}-(n-p)^{1-\beta}}{n^{1-\beta}}\frac{u_i^{p+1}-u_i^p}{\Delta t}. \end{aligned} \tag{21}$$

Therefore, we have the following finite difference discretization of Eq (8) using Eq (21):

$$\sum_{p=1}^{n} w_p^n\frac{u_i^{p+1}-u_i^p}{\Delta t} = \frac{u_{i-1}^{n+1}-2u_i^{n+1}+u_{i+1}^{n+1}}{h^2}, \tag{22}$$

where

$$w_p^n = \frac{(n+1-p)^{1-\beta}-(n-p)^{1-\beta}}{n^{1-\beta}}. \tag{23}$$

We note that the weight parameter $w_p^n$ satisfies the following condition for any value of $n$:

$$\sum_{p=1}^{n} w_p^n = 1. \tag{24}$$

Then, Eq (22) can be rewritten as follows:

$$-\frac{1}{h^2}u_{i-1}^{n+1}+\left(\frac{w_n^n}{\Delta t}+\frac{2}{h^2}\right)u_i^{n+1}-\frac{1}{h^2}u_{i+1}^{n+1} = \frac{w_n^n}{\Delta t}u_i^n-\sum_{p=1}^{n-1}w_p^n\frac{u_i^{p+1}-u_i^p}{\Delta t}. \tag{25}$$

Equation (25) is a tridiagonal system with a zero Dirichlet boundary condition and we can use the Thomas algorithm to efficiently solve this system. Thus, the solution vector $\mathbf{u}^{n+1}$ can be found by solving the tridiagonal system using the Thomas algorithm:

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

where $A$ is a tridiagonal matrix with zero Dirichlet at $i = 1$ and $i = N_x$.

## 3. Numerical experiments

In this section, we present several numerical experiments in a finite domain $\Omega \times (0, T)$, where $\Omega = (0, 1)$ and $T$ is a final time, to investigate the effects of $\alpha$ and $\beta$ on the evolution dynamics of the conventional and normalized time-fractional diffusion equations. As the first numerical test, we consider the following low-frequency initial condition:

$$u(x, 0) = \sin(2\pi x), \text{ for } x \in \Omega. \tag{26}$$

Figure 5(a) and (b) show $u(x,T)$ for different values of $\alpha = 0.1, 0.5$, and 0.9; and $\beta = 0.1, 0.5$, and 0.9, respectively. Here, $T = 0.001$ is used. As shown in Figure 5(a), for the conventional time-fractional diffusion equation, we observe that the temporal evolution is faster when the value of $\alpha$ is smaller. This phenomenon is attributed to the different weight function associated with different $\alpha$ values, as shown in Figure 1(a). When $\alpha$ is smaller, the total sum of the weight function at early times is smaller as shown in Figure 3, which results in an effectively larger diffusion process and faster temporal evolution. This can be understood as follows: if we divide both sides of Eq (1) by the smaller weight value, we obtain an effectively larger diffusion coefficient, which leads to an increased diffusion process. However, in the case of the normalized time-fractional diffusion equation, there is little variation with respect to different values of $\beta$ compared to the conventional time-fractional diffusion equation, as seen in Figure 5(b).

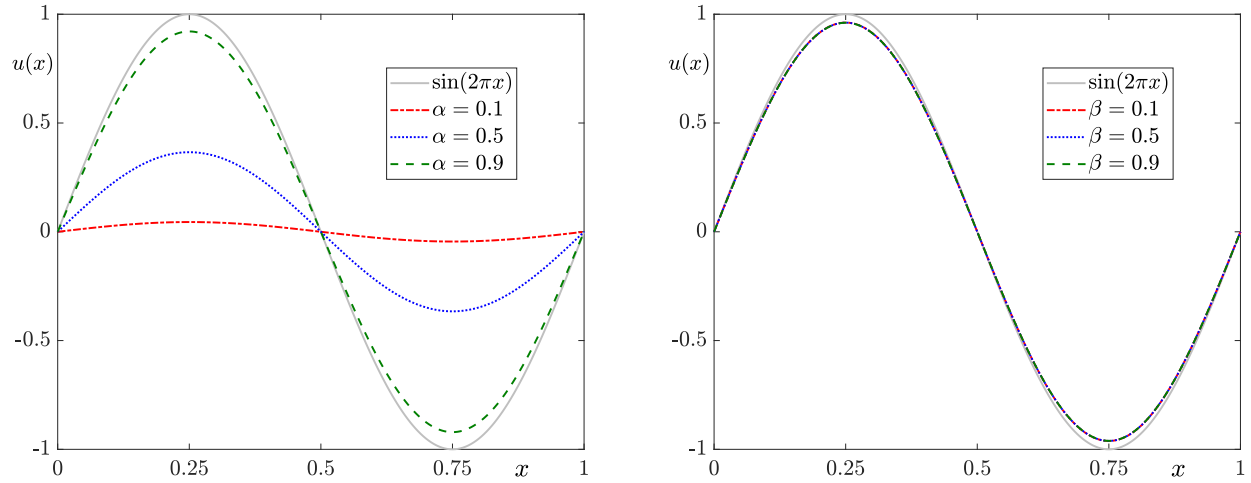
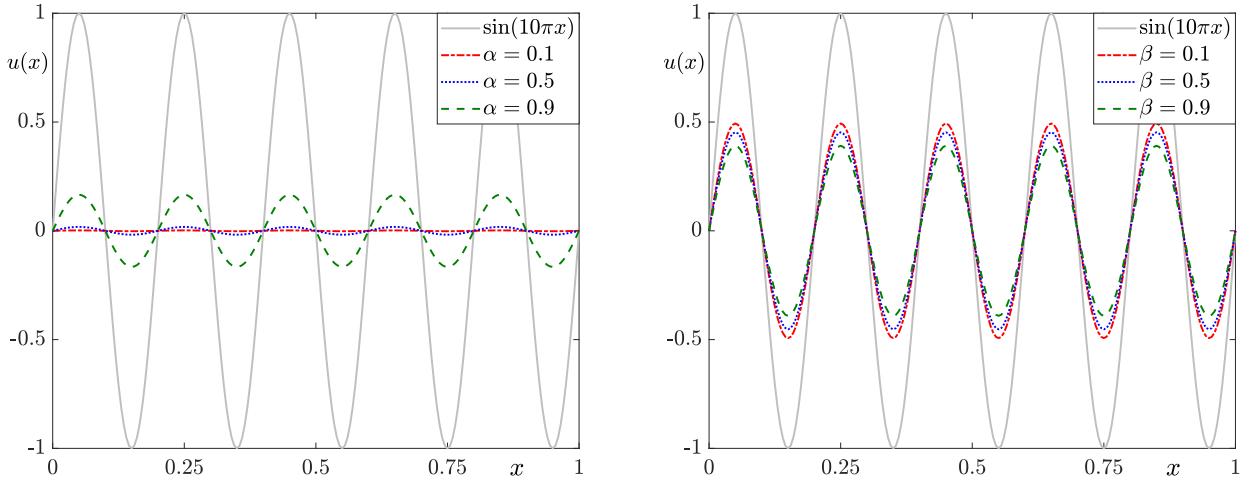



**Figure 5.** (a) and (b) are the numerical solutions for different values of $\alpha$ and $\beta$, respectively. Here, $T = 0.001$ is used.

In the second numerical experiment, let us consider the following high-frequency initial condition:

$$u(x,0) = \sin(10\pi x), \text{ for } x \in \Omega. \tag{27}$$

Figures 6(a) and (b) show $u(x,T)$ for different values of $\alpha = 0.1, 0.5$, and 0.9; and $\beta = 0.1, 0.5$, and 0.9, respectively. We observe that the temporal evolutions are faster than those of the low-frequency initial condition for both the conventional and normalized time-fractional diffusion equations. As illustrated in Figure 6(a), the temporal evolution for the conventional time-fractional diffusion equation accelerates more than that of the low-frequency initial condition as the value of $\alpha$ decreases. A smaller $\alpha$ results in a smaller total sum of the weight function, which leads to an effectively larger diffusion process and consequently faster temporal evolution. However, for the normalized time-fractional diffusion equation, there is a little variation concerning different $\beta$ values, unlike the conventional time-fractional diffusion equation, as shown in Figure 6(b).




(a) (b)

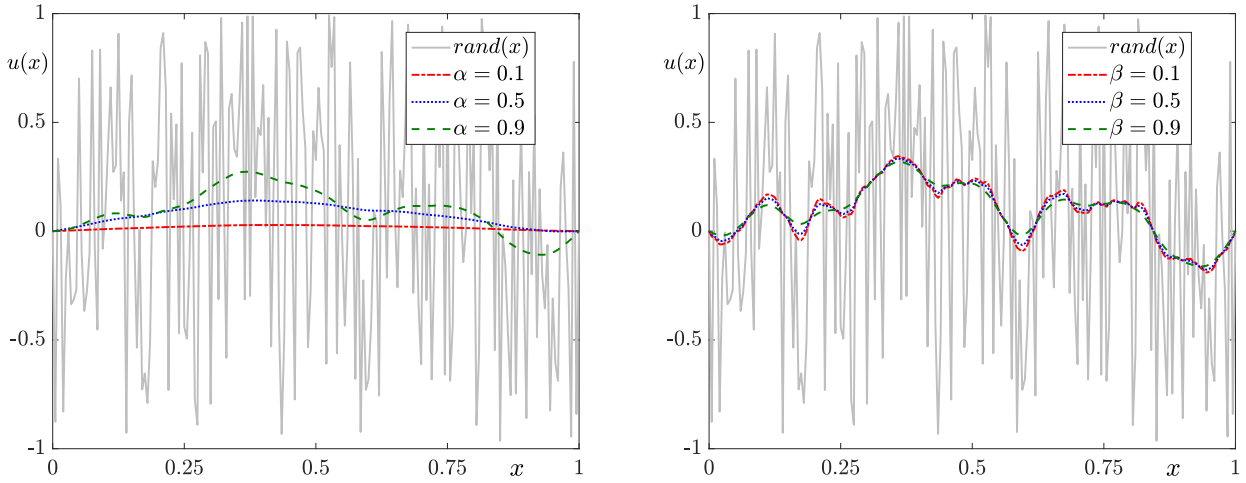
**Figure 6.** (a) and (b) are the numerical solutions for different values of $\alpha$ and $\beta$, respectively. Here, $T = 0.001$ is used.

For the third numerical test, let us consider the following random initial condition:

$$u(x, 0) = rand(x), \text{ for } x \in \Omega, \tag{28}$$

$$u(0, t) = u(1, t) = 0, \tag{29}$$

where $rand(x)$ is a random number between $-1$ and 1. Figures 7(a) and (b) show $u(x, T)$ for different values of $\alpha = 0.1, 0.5$, and 0.9; and $\beta = 0.1, 0.5$, and 0.9, respectively. The random initial condition can be considered as a combination of multiple frequency modes. As expected from the previous computational results, we observe large variation with respect to $\alpha$ values and small variation with respect to $\beta$ values.




(a) (b)

**Figure 7.** (a) and (b) are the numerical solutions for different values of $\alpha$ and $\beta$, respectively. Here, $T = 0.001$ is used.

In the fourth numerical test, we consider the following initial condition:

$$u(x,0) = \sin\left(7\sqrt{\pi x}\right), \text{ for } x \in (0,\pi). \tag{30}$$

This initial condition contains multiple modes with gradually changing profiles. Figure 8(a) shows the numerical solution $u(x,T)$ for a relatively short time, $T = 0.0005$, and Figure 8(b) and 8(c) display $u(x,T)$ for a relatively long time, $T = 0.05$. Figures 8(a) and 8(b) present results for different values of $\alpha = 0.1, 0.5$, and 0.9, and Fig. 8(c) shows results for different values of $\beta = 0.1, 0.5$, and 0.9. Because the initial condition is a combination of multiple frequency modes, we observe significant evolutions at high frequencies and minor changes at low frequencies.

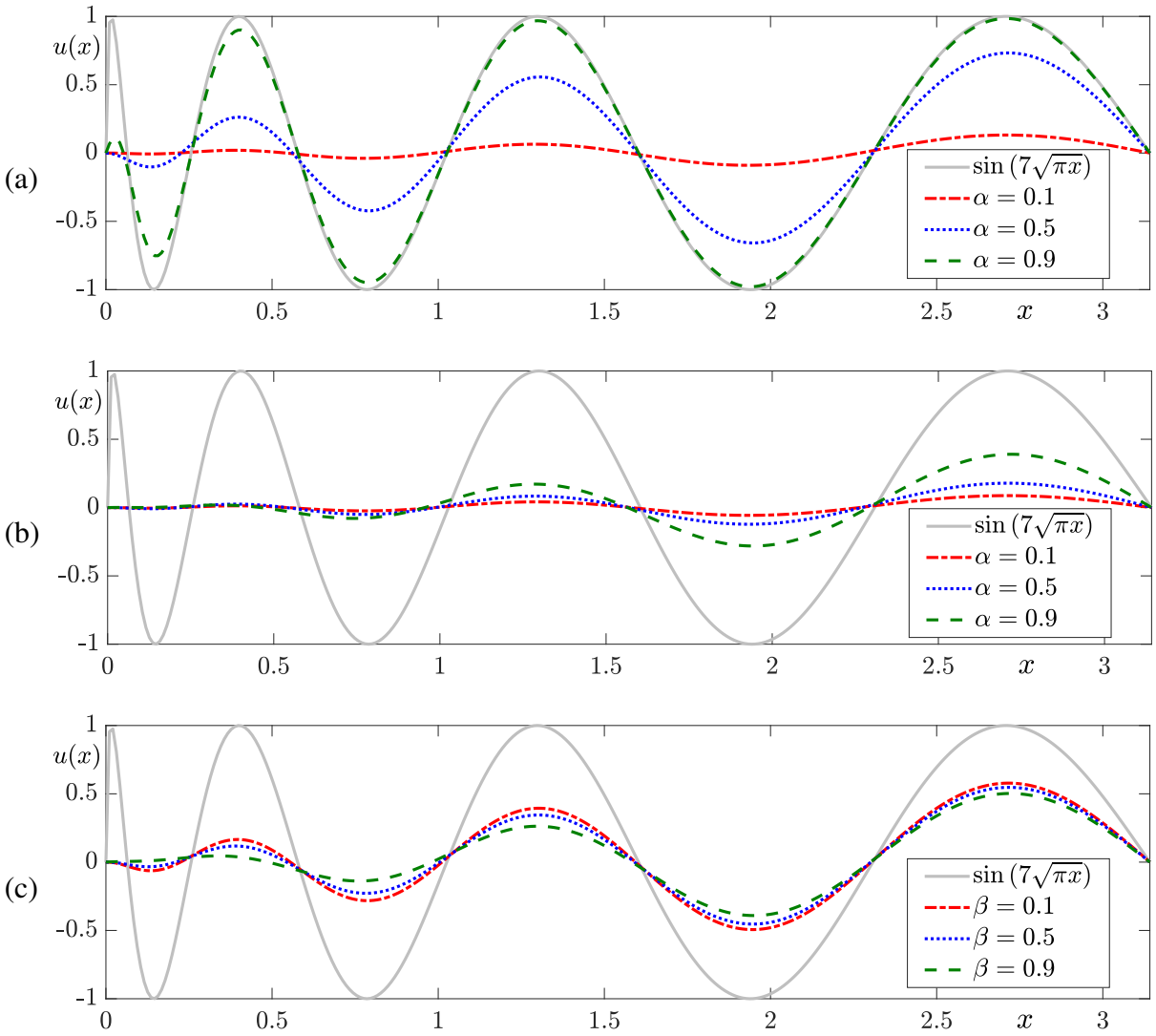



**Figure 8.** (a) and (b) are the numerical solutions for different times $T = 0.0005$ and $T = 0.05$, respectively, with $\alpha = 0.1, 0.5$, and 0.9. (c) is the numerical solution for $T = 0.05$ with $\beta = 0.1, 0.5$, and 0.9.

In the final numerical test, we examine the evolution process of the conventional time-fractional method and the normalized time-fractional method for different fractional orders $\alpha = 0.1, 0.9$ and

$\beta = 0.1, 0.9$. We consider the following initial condition:

$$u(x, 0) = \sin(0.2\pi x), \text{ for } x \in (0, 5). \tag{31}$$

In this test, we used a final time of $T = 5$. Figures 9(a) and (b) represent the evolution of the conventional time-fractional diffusion equation for the different $\alpha = 0.1$ and 0.9, respectively. When $\alpha$ is small, the temporal evolution is fast at early times but suddenly slows down at later times. This phenomenon, as shown in Figure 3, can be attributed to the small weight function values at early times and the large weight function values at later times. When $\alpha$ is large, the opposite behavior can be observed. Next, Figures 9(c) and (d) illustrate the evolution of the normalized time-fractional diffusion equation for the different values of $\beta = 0.1$ and 0.9, respectively. In this case, the temporal evolution depends on the $\beta$ values but it is relatively independent of time.

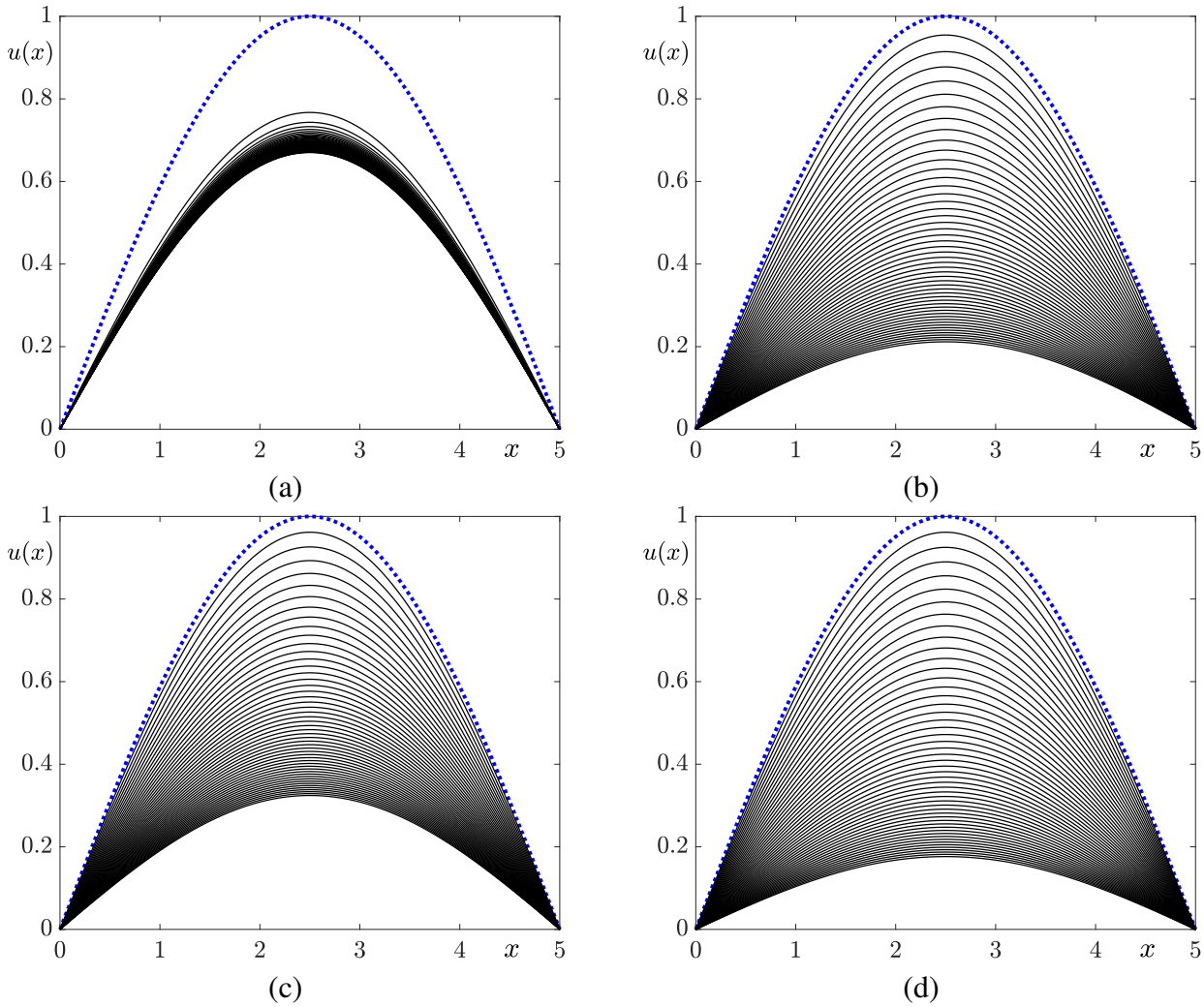



**Figure 9.** (a) and (b) are the evolution process of the conventional time-fractional diffusion equation when $\alpha = 0.1$ and 0.9, respectively. (c) and (d) are the evolution process of the normalized time-fractional diffusion equation when $\beta = 0.1$ and 0.9, respectively.

## 4. Conclusions

In summary, the evolution dynamics of the conventional and normalized time-fractional diffusion equations are different. The total integration of the weight function for the conventional time-fractional diffusion equation is not normalized, continues to increase to infinity as time progresses; it depends on the values of $\alpha$ and time $t$. Due to these dependencies, it is difficult to study the effects of $\alpha$ on the evolutionary dynamics. As an alternative, we proposed a normalized time-fractional diffusion equation that ensures that the total integration of the weight function remains one, regardless of the values of $\beta$ and time $t$. The proposed weight functions provide an intuitive understanding of the fractional time derivative, which represents a weighted average of the temporal history of the time derivative. Numerical experiments indicate that the normalized time-fractional diffusion equation shows slower dynamics for simple initial conditions as the fractional order parameter $\beta$ decreases. The diffusion equation is a fundamental model for various phenomena such as heat conduction, fluid flow, chemical dispersion, cellular diffusion processes, and the evolution of option prices and market risks in finance. As future work, we will apply the proposed model to the Black–Scholes equation [23, 24], the AC equation [25–27], and the nonlocal Cahn–Hilliard equation [28], and investigate the development of a fast algorithm to reduce memory requirements [29].

## Author contributions

Chaeyoung Lee: Investigation, Writing–original draft, Validation, Writing–Review & Editing, Software; Yunjae Nam: Formal analysis, Visualization, Writing–original draft, Writing–Review & Editing; Minjoon Bang: Writing–original draft, Writing–Review & Editing, Software, Visualization; Seokjun Ham: Validation, Writing–original draft, Writing–Review & Editing, Visualization; Junseok Kim: Conceptualization, Investigation, Formal analysis, Supervision, Writing–original draft, Writing–Review & Editing. All authors have read and agreed to the published version of the manuscript.

## Acknowledgments

The first author (C. Lee) was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1C1C2003896). The corresponding author (J.S. Kim) was supported by the Brain Korea 21 FOUR from the Ministry of Education of Korea. The authors are grateful to the reviewers for their insightful and constructive feedback, which has significantly enhanced the quality of this paper.

## Conflict of interest

The authors declare that there is no conflict of interest.

## Appendix

The following listings 1 and 2 are Python codes for the conventional and normalized time-fractional diffusion equations, respectively.

Listing 1. Python code for a conventional time-fractional heat equation

```
# Import
import numpy as np
import matplotlib.pyplot as plt
from math import gamma

# Thomas algorithm
def thomas(alpha,beta,gamma,f):
    n=len(f)
    beta=np.copy(beta)
    f=np.copy(f)
    for i in range(1,n):
        mult=alpha[i]/beta[i-1]
        beta[i]-=mult*gamma[i-1]
        f[i]-=mult*f[i-1]
    x=np.zeros(n)
    x[-1]=f[-1]/beta[-1]
    for i in range(n-2,-1,-1):
        x[i]=(f[i]-gamma[i]*x[i+1])/beta[i]
    return x

# Alpha
alpha=0.1

# Parameters
L=np.pi; Nx=401; h=L/Nx; x=np.linspace(0,L,Nx);
T=0.05; dt=1.0000e-04; Nt=round(T/dt); dt=T/Nt;
u=np.zeros((Nx,Nt+1)); u[:,0]=np.sin(7*(np.pi*x)**0.5);
deno=dt**(alpha-1)*gamma(2-alpha)
a=(-1/h**2)*np.ones(Nx); c=a;

# Main iteration
for n in range(Nt):
    w=np.zeros(n+1)
    for p in range(n+1):
        w[p]=((n+1-p)**(1-alpha)-(n-p)**(1-alpha))/deno
    F=np.zeros(Nx)
    if n>0:
        for p in range(n):
            F+=w[p]*(u[:,p+1]-u[:,p])/dt
    d=np.zeros(Nx)
    for i in range(Nx):
        d[i]=w[n]/dt+2.0/h**2
```

```
        f=w[n]/dt*u[:,n]-F
        u[1:Nx-1,n+1]=thomas(a[1:Nx-1],d[1:Nx-1],c[1:Nx-1],f[1:Nx-1])

    # Plot
    plt.figure(figsize=(10,6))
    plt.plot(x,u[:,0],'-',color='0.5',linewidth=2,label='Initial')
    plt.plot(x,u[:,-1],'r--',linewidth=2,label='Final')
    plt.axis([0, np.pi, -1, 1])
    plt.legend()
    plt.show()
```

Listing 2. Python code for a normalized time-fractional heat equation

```
# Import
import numpy as np
import matplotlib.pyplot as plt

# Thomas algorithm
def thomas(alpha,beta,gamma,f):
    n=len(f)
    beta=np.copy(beta)
    f=np.copy(f)
    for i in range(1,n):
        mult=alpha[i]/beta[i-1]
        beta[i]-=mult*gamma[i-1]
        f[i]-=mult*f[i-1]
    x=np.zeros(n)
    x[-1]=f[-1]/beta[-1]
    for i in range(n-2,-1,-1):
        x[i]=(f[i]-gamma[i]*x[i+1])/beta[i]
    return x

# Beta
beta=0.1

# Parameters
L=np.pi; Nx=401; h=L/Nx;
x=np.linspace(0,L,Nx); T=0.05; dt=1.0000e-04;
Nt=round(T/dt); dt=T/Nt; u=np.zeros((Nx,Nt+1));
u[:,0]=np.sin(7*(np.pi*x)**0.5);
a=(-1/h**2)*np.ones(Nx); c=a;

# Main iteration
```

```
    for n in range(1,Nt+1):
        deno=n**(1-beta)
        w=np.zeros(n)
        for p in range(n):
            w[p]=((n+1-p)**(1-beta)-(n-p)**(1-beta))/deno
        F=np.zeros(Nx)
        if n>1:
            for p in range(n-1):
                F+=w[p]*(u[:,p+1]-u[:,p])/dt
        d=np.zeros(Nx)
        for i in range(Nx):
            d[i]=w[n-1]/dt+2.0/h**2
        f=w[n-1]/dt*u[:,n-1]-F
        u[1:Nx-1,n]=thomas(a[1:Nx-1],d[1:Nx-1],c[1:Nx-1],f[1:Nx-1])

    # Plot
    plt.figure(figsize=(10,6))
    plt.plot(x,u[:,0],'-',color='0.5',linewidth=2,label='Initial')
    plt.plot(x,u[:,-1],'r--',linewidth=2,label='Final')
    plt.axis([0, np.pi, -1, 1])
    plt.legend()
    plt.show()
```

## References


1. J. J. Liu, M. Yamamoto, A backward problem for the time-fractional diffusion equation, *Appl. Anal.,* **89** (2010), 1769–1788. https://doi.org/10.1080/00036810903479731
2. L. Feng, I. Turner, P. Perré, K. Burrage, The use of a time-fractional transport model for performing computational homogenisation of 2D heterogeneous media exhibiting memory effects, *J. Comput. Phys.,* **480** (2023), 112020. https://doi.org/10.1016/j.jcp.2023.112020
3. M. Biglari, A. R. Soheili, Efficient simulation of two-dimensional time-fractional Navier–Stokes equations using RBF-FD approach, *Eng. Anal. Bound. Elem.,* **160** (2024), 134–159. https://doi.org/10.1016/j.enganabound.2023.12.021
4. F. A. Rihan, Q. M. Al-Mdallal, H. J. AlSakaji, A. Hashish, A fractional-order epidemic model with time-delay and nonlinear incidence rate, *Chaos Soliton. Fract.,* **126** (2019), 97–105. https://doi.org/10.1016/j.chaos.2019.05.039
5. M. Inc, The approximate and exact solutions of the space-and time-fractional Burgers equations with initial conditions by variational iteration method, *J. Math. Anal. Appl.,* **345** (2008), 476–484. https://doi.org/10.1016/j.jmaa.2008.04.007
6. J. G. Liu, J. Zhang, A new approximate method to the time fractional damped Burger equation, *AIMS Math.,* **8** (2023), 13317–13324. https://doi.org/10.3934/math.2023674

7. A. M. Zidan, A. Khan, R. Shah, M. K. Alaoui, W. Weera, Evaluation of time-fractional Fisher's equations with the help of analytical methods, *AIMS Math.,* **7** (2022), 18746–66. https://doi.org/10.3934/math.20221031
8. X. Qin, X. Yang, P. Lyu, A class of explicit implicit alternating difference schemes for generalized time fractional Fisher equation, *AIMS Math.,* **6** (2021), 11449–11466. https://doi.org/10.3934/math.2021663
9. W. Chen, X. Xu, S. P. Zhu, Analytically pricing double barrier options based on a time-fractional Black–Scholes equation, *Comput. Math. Appl.,* **69** (2015), 1407–1419. https://doi.org/10.1016/j.camwa.2015.03.025
10. A. Golbabai, O. Nikan, T. Nikazad, Numerical analysis of time fractional Black–Scholes European option pricing model arising in financial market, *Comput. Appl. Math.,* **38** (2019), 1–24. https://doi.org/10.1007/s40314-019-0957-7
11. H. Zhang, F. Liu, I. Turner, Q. Yang, Numerical solution of the time fractional Black–Scholes model governing European options, *Comput. Math. Appl.,* **71** (2016), 1772–1783. https://doi.org/10.1016/j.camwa.2016.02.007
12. Q. Du, J. Yang, Z. Zhou, Time-fractional Allen–Cahn equations: analysis and numerical methods, *J. Sci. Comput.,* **85** (2020), 42. https://doi.org/10.1007/s10915-020-01351-5
13. H. Liu, A. Cheng, H. Wang, J. Zhao, Time-fractional Allen–Cahn and Cahn–Hilliard phase-field models and their numerical investigation, *Comput. Math. Appl.,* **76** (2018), 1876–1892. https://doi.org/10.1016/j.jocs.2023.102114
14. 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
15. W. M. Abd-Elhameed, H. M. Ahmed, Spectral solutions for the time-fractional heat differential equation through a novel unified sequence of Chebyshev polynomials, *AIMS Math.,* **9** (2024), 2137–2166. https://doi.org/10.3934/math.2024107
16. Y. E. Aghdam, H. Mesgarani, Z. Asadi, V. T. Nguyen, Investigation and analysis of the numerical approach to solve the multi-term time-fractional advection-diffusion model, *AIMS Math.,* **8** (2023), 29474. https://doi.org/10.3934/math.20231509
17. J. Kim, S. Kwak, H. G. Lee, Y. Hwang, S. Ham, A maximum principle of the Fourier spectral method for diffusion equations, *Electron. Res. Arch.,* **31** (2023), 5396–5405. https://doi.org/10.3934/era.2023273
18. J. M. Carcione, Theory and modeling of constant-Q P-and S-waves using fractional time derivatives, *Geophysics,* **74** (2009), T1–T11. https://doi.org/10.1190/1.3008548
19. J. M. Carcione, F. Cavallini, F. Mainardi, A. Hanyga, Time-domain modeling of constant-Q seismic waves using fractional derivatives, *Pure Appl. Geophys.,* **159** (2002), 1719–1736. https://doi.org/10.1007/s00024-002-8705-z
20. S. Ham, J. Kim, Stability analysis for a maximum principle preserving explicit scheme of the Allen–Cahn equation, *Math. Comput. Simul.,* **207** (2023), 453–465. https://doi.org/10.1016/j.matcom.2023.01.016

21. J. Wang, Z. Han, W. Jiang, J. Kim, A fast, efficient, and explicit phase-field model for 3D mesh denoising, *Appl. Math. Comput.,* **458** (2023), 128239. https://doi.org/10.1016/j.amc.2023.128239

22. J. W. Thomas, Numerical partial differential equations: finite difference methods in *Springer Science & Business Media* (2013).

23. M. Sarboland, A. Aminataei, On the numerical solution of time fractional Black-Scholes equation, *Int. J. Comput. Math.,* **99** (2022), 1736–1753. https://doi.org/10.1080/00207160.2021.2011248

24. J. Huang, Z. Cen, J. Zhao, An adaptive moving mesh method for a time-fractional Black—Scholes equation, *Adv. Differ. Equ.,* **2019** (2019), 1–14. https://doi.org/10.1186/s13662-019-2453-1

25. B. Xia, R. Yu, X. Song, X. Zhang, J. Kim, An efficient data assimilation algorithm using the Allen–Cahn equation, *Eng. Anal. Bound. Elem.,* **155** (2023), 511–517. https://doi.org/10.1016/j.enganabound.2023.06.029

26. Y. Hwang, I. Kim, S. Kwak, S. Ham, S. Kim, J. Kim, Unconditionally stable monte carlo simulation for solving the multi-dimensional Allen–Cahn equation, *Electron. Res. Arch.,* **31** (2023), 5104–5123. https://doi.org/10.3934/era.2023261

27. Y. Hwang, S. Ham, C. Lee, G. Lee, S. Kang, J. Kim, A simple and efficient numerical method for the Allen–Cahn equation on effective symmetric triangular meshes, *Electron. Res. Arch.,* **31** (2023), 4557–4578. https://doi.org/10.3934/era.2023233

28. C. Lee, S. Kim, S. Kwak, Y. Hwang, S. Ham, S. Kang, J. Kim, Semi-automatic fingerprint image restoration algorithm using a partial differential equation, *AIMS Math.,* **8** (2023), 27528-27541. https://doi.org/10.3934/math.20231408

29. Z. W. Fang, H. W. Sun, H. Wang, A fast method for variable-order Caputo fractional derivative with applications to time-fractional diffusion equations, *Comput. Math. Appl.,* **80** (2020), 1443–1458. https://doi.org/10.1016/j.camwa.2020.07.009


AIMS Press