Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 14, No. 2, 2026, pp. 947-957
DOI:10.22034/cmde.2025.64886.2956

Implicit cubic B-spline quasi-interpolation for solving the generalized distributed-order time-fractional Black-Scholes equation

Roya Montazeri*

Department of Mathematics, Payame Noor University (PNU), P.O.Box 19395-4697, Tehran, Iran.

Abstract

In this study, we implicitly solve the generalized distributed-order time-fractional Black-Scholes equation. We employ finite differences to approximate the time derivatives and cubic B-spline quasi-interpolation for the spatial derivatives. The error analysis of the presented method is investigated. The algorithm of this method is also presented, which shows the simplicity of implementing the method to solve the generalized distributed-order time-fractional Black-Scholes equation. Numerical results demonstrate the method's convergence rate and accuracy.

Keywords. Generalized distributed–order, Fractional derivatives, Black-Scholes equation, B-spline quasi interpolation.

2010 Mathematics Subject Classification. 26A33, 35G25, 65D07.

1. Introduction

The Black-Scholes equation is a fundamental partial differential equation in financial mathematics, used to model the dynamics of financial derivatives, particularly options. Developed by Fischer Black, Myron Scholes, and Robert Merton in 1973, the equation provides a theoretical estimate of the price of European-style options [2, 9].

The Black-Scholes model assumes a frictionless market and uses the concept of risk-neutral pricing. The key insight is that one can hedge an option by continuously buying and selling the underlying asset to eliminate risk, leading to a unique price for the option [2]. The equation is given by

$$\frac{\partial V}{\partial t}+\frac{1}{2}\sigma^2S^2\frac{\partial^2V}{\partial S^2}+rS\frac{\partial V}{\partial S}-rV=0, \tag{1.1}$$

where V is the option price, S is the underlying asset price, t is time, σ is the volatility of the asset, and r is the risk-free interest rate.

The Black-Scholes formula, derived from this equation, has become a cornerstone in the field of financial engineering and is widely used for pricing options and other derivatives [6].

The time fractional Black-Scholes equation is an extension of the classical Black-Scholes model, incorporating fractional calculus to better capture the complexities of financial markets. This model was first introduced by Wyss in 2000, who applied the concept of fractals to financial modeling [13]. The time fractional Black-Scholes equation is particularly useful for modeling markets with memory and hereditary properties, which are not adequately addressed by the traditional Black-Scholes model [7].

The equation is given by

$$\frac{\partial^\alpha V}{\partial t^\alpha}+\frac{1}{2}\sigma^2S^2\frac{\partial^2V}{\partial S^2}+rS\frac{\partial V}{\partial S}-rV=0,$$

where $0<\alpha\leq 1$ represents the order of the fractional derivative.

Received: 06 December 2024; Accepted: 22 April 2025.
∗ Corresponding author. Email: roya.mn@pnu.ac.ir.

This fractional model provides a more accurate representation of market behaviors, capturing both major jumps over short periods and long-range dependencies [12]. It has been the subject of extensive research, leading to various numerical methods for its solution [7].

The distributed-order time-fractional derivative is a generalization of the fractional derivative concept, where the order of differentiation is distributed over a range of values rather than being fixed. This approach allows for a more flexible and accurate modeling of complex systems that exhibit behaviors influenced by multiple time scales and memory effects [5, 8].

Mathematically, the distributed-order time-fractional derivative is defined as

$$D_t^{(\mu)} f(t) = \int_0^1 \mu(\alpha) \frac{\partial^\alpha f(t)}{\partial t^\alpha} \, d\alpha,$$

where $\mu(\alpha)$ is a weight function that determines the contribution of each fractional order α within the interval $[0, 1]$ [11]. This derivative is particularly useful in fields such as viscoelasticity, anomalous diffusion, and control theory, where systems exhibit a combination of local and nonlocal behaviors [3, 5].

The distributed-order time-fractional derivative has been applied to various types of differential equations, enhancing their ability to model real-world phenomena more accurately. For instance, it has been used in the study of ultra-slow diffusion processes and in the development of more robust numerical methods for solving fractional differential equations [8, 11].

In this paper, we examine the nonlinear generalized distributed-order time-fractional Black-Scholes equation, which models the price of the European double barrier option [14].

$$\frac{\partial u(x,t)}{\partial t} + \rho \frac{{}^w\partial^* u(x,t)}{\partial t} - a \frac{\partial^2 u(x,t)}{\partial x^2} + b \frac{\partial u(x,t)}{\partial x} + c = f(x,t), \qquad x \in (x_l, x_r),\ t \in (0, T], \tag{1.2}$$

with the initial condition

$$u(x,0) = \psi(x),\ x_l \le x \le x_r, \tag{1.3}$$

and homogeneous Dirichlet boundary conditions

$$u(x_l,t) = u(x_r,t) = 0,\ 0 < t \le T, \tag{1.4}$$

where $x = Ln(S)$, $V(S, T-t) = u(x,t)$, the parameter $\rho \ge 0$ balances integer-order and distributed-order time derivatives, $a = \frac{\sigma^2}{2}$, $b = a - r$, $c = r$. Here, $r \ge 0$ is the risk-free rate and $\sigma \ge 0$ is the volatility. $f \in C[x_l, x_r] \times C[0, T]$ and $\psi \in C^2[x_l, x_r]$ are considered as known functions. Here, ${}^w\partial^*/\partial t$ denotes the generalized distributed-order differential operator which is defined as follow

$$\frac{{}^w\partial^* u(x,t)}{\partial t} = \int_{\alpha_l}^{\alpha_r} w(\alpha, x, t) \partial_t^\alpha u(x,t) d\alpha, \tag{1.5}$$

where $\partial_t^\alpha u(x,t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{1}{(t-s)^\alpha} \frac{\partial u(x,s)}{\partial s} ds$ and $w(\alpha, x, t)$ is known density function.

The rest of the paper is organized as follow: In the second section, we present a temporal discrete scheme for Equation (1.2) using the finite difference method. Then we use cubic spline quasi-interpolation for spatial approximation. The error analysis of the presented method is investigated in section 3. In the forth section,the numerical results are shown. Some conclusions are given at the end of the paper in section 5.

2. Numerical scheme

Let $t_n = n\Delta t$, $n = 0, 1, \ldots, N_t$, $\Delta t = \frac{T}{N_t}$, $t_{n+\theta} = (n+\theta)\Delta t = t_n + \theta \Delta t$, and $\alpha_k = \alpha_l + k\Delta\alpha$, $k = 0, 1, \ldots, N_\alpha$, $\Delta\alpha = \frac{\alpha_r - \alpha_l}{N_\alpha}$. Also, assume that $w_k^n = w(\alpha_k, x, t_n)$ and $u^n = u(x, t_n)$. Using the trapezoidal integration method, we have

$${}^w\partial_t^* u^n = \frac{\Delta\alpha}{2} \left(w_0^n \partial_t^{\alpha_0} u^n + 2 \sum_{k=1}^{N_\alpha - 1} w_k^n \partial_t^{\alpha_k} u^n + w_{N_\alpha}^n \partial_t^{\alpha_{N_\alpha}} u^n \right) + O((\Delta\alpha)^2). \tag{2.1}$$

For approximation the fractional derivatives we use following L1 method

$$\partial_t^\alpha u^n = \left(u^{i+1} - u^i\right) J_{i,n}^\alpha + O(\Delta t), \tag{2.2}$$

where $J_{i,n}^\alpha = \int_{t_i}^{t_{i+1}} (t_n - s)^{-\alpha} ds$. By substituting (2.2) in (2.1) we obtain

$${}^w\partial_t^* u^n = \frac{\Delta\alpha}{2\Delta t} \sum_{i=0}^{n-1} (u^{i+1} - u^i) X_i^n + O((\Delta\alpha)^2 + (\Delta\alpha)(\Delta t)), \tag{2.3}$$

where $X_i^n(x) = \nu_{i,0}^n(x) + 2\sum_{k=1}^{N_\alpha - 1} \nu_{i,k}^n(x) + \nu_{i,N_\alpha}^n(x)$ and $\nu_{i,k}^n(x) = \frac{w_k^n(x) J_{i,n}^{\alpha_k}}{\Gamma(1-\alpha_k)}$. By putting $t = t_{n+1/2}$ in (1.2) and using Crank–Nicolson method we get

$$\begin{aligned} u^{n+1} - u^n + \frac{\rho\Delta t}{2}\left({}^w\partial_t^* u^{n+1} + {}^w\partial_t^* u^n\right) - \frac{a\Delta t}{2}\left(u_{xx}^{n+1} + u_{xx}^n\right) \\ + \frac{b\Delta t}{2}\left(u_x^{n+1} + u_x^n\right) + c\Delta t + O((\Delta t)^2) = \Delta t f(x, t_{n+\frac{1}{2}}), \end{aligned} \tag{2.4}$$

Putting (2.3) in (2.4) gives the result

$$Q^n u^{n+1} - \frac{a\Delta t}{2} u_{xx}^{n+1} + \frac{b\Delta t}{2} u_x^{n+1} = R^n + O((\Delta\alpha)^2(\Delta t) + (\Delta\alpha)(\Delta t)^2 + (\Delta t)^2), \tag{2.5}$$

where

$$Q^n = 1 + \frac{\rho\Delta\alpha}{4} X_n^{n+1}, \tag{2.6}$$

$$R^n = Q^n u^n + \frac{a\Delta t}{2} u_{xx}^n - \frac{b\Delta t}{2} u_x^n - \frac{\rho\Delta\alpha}{4} \sum_{i=0}^{n-1} V_i^n (u^{i+1} - u^i) - c\Delta t + \Delta t f(x, t_{n+\frac{1}{2}}), \tag{2.7}$$

$$V_i^n = X_i^{n+1} + X_i^n. \tag{2.8}$$

Therefore we will have the following numerical scheme

$$Q^n U^{n+1} - \frac{a\Delta t}{2} U_{xx}^{n+1} + \frac{b\Delta t}{2} U_x^{n+1} = Rh^n, \tag{2.9}$$

where U^n and Rh^n denoted the approximations of u^n and R^n in (x, t_n), respectively. For spatial approximation, we use cubic spline quasi-interpolation [1]. Let $W = \{x_0, x_1, \ldots, x_{N_x}\}$ be a uniform partition of interval $[x_l, x_r]$, where $x_0 = x_l$, $x_{N_x} = x_r$ and $x_{m+1} - x_m = \Delta x$, $m = 0, 1, \ldots, N_x$. Also, let $\mathbb{S}_4(W)$ be the space of cubic splines on

$$W^* = \{x_{-3}, x_{-2}, x_{-1}, x_0, x_1, \ldots, x_{N_x}, x_{N_x+1}, x_{N_x+2}, x_{N_x+3}\},$$

where $x_{-3} = x_{-2} = x_{-1} = x_0$ and $x_{N_x} = x_{N_x+1} = x_{N_x+2} = x_{N_x+3}$. The corresponding set of cubic B-splines $\{B_{-3}, B_{-2} \ldots, B_{N_x-1}\}$, where is a basis for $\mathbb{S}_4(W)$, are defined using the recursive relation [4],

$$B_{j,p}(x) = \frac{x - x_j}{x_{j+p} - x_j} B_{j,p-1}(x) + \frac{x_{j+p+1} - x}{x_{j+p+1} - x_{j+1}} B_{j+1,p-1}(x), \tag{2.10}$$

where $B_{j,0}(x) = \delta[x_j, x_{j+1})$, $j = -1, 0, \ldots, N_x - 1$ and $p = 3$, and under the convention that fractions with zero denominator have value zero. It is notably that a B-spline is right continuous, i.e., the value at a point x is obtained by taking the limit from the right. The univariate cubic spline quasi-interpolants can be defined as operators of the form [1]

$$(\Theta U^n)^{(\iota)}(x) = \sum_{j=1}^{N_x+1} \mu_j(U^n) B_{j-3}^{(\iota)}(x), \tag{2.11}$$

where $\iota = 0, 1, 2$ refer to ιth derivative of functions and

$$\mu_1(U^n) = \frac{1}{18}(18U_1^n - 9U_2^n + 2U_3^n),\ \mu_2(U^n) = \frac{1}{6}(8U_1^n - U_2^n),$$
$$\mu_j(U^n) = \frac{1}{6}(-U_{j-2}^n + 8U_{j-1}^n - U_j^n),\ j = 3, 4, \ldots, N_x - 1,$$
$$\mu_{N_x}(U^n) = \frac{1}{6}(-U_{N_x-2}^n + 8U_{N_x-1}^n),\ \mu_{N_x+1}(U^n) = \frac{1}{18}(2U_{N_x-3}^n - 9U_{N_x-2}^n + 18U_{N_x-1}^n),$$

where $U_m^n = U^n(x_m)$. We know from [4] that $\left\|(\Theta U^n)^{(\iota)} - (U^n)^{(\iota)}\right\|_\infty = O\left((\Delta x)^{(4-\iota)}\right)$ for $\iota = 0, 1, 2$. By substituting (2.11) as approximation in (2.9) we obtain the following system of linear equations

$$A^n \mathbf{U}^{n+1} = \mathbf{R}^n, \tag{2.12}$$

where $\mathbf{U}^n = (U_1^n, U_2^n, \ldots, U_{N_x-1}^n)^T$, $\mathbf{R}^n = (Rh_1^n, Rh_2^n, \ldots, Rh_{N_x-1}^n)^T$, $(A^n)_{m,j} = (\Pi_j^n(x_m))$, $j, m = 1, 2, \ldots, N_x - 1$, and

$$\Pi_1^n(x) = \Lambda_1^n(x) + \frac{8}{6}\Lambda_2^n(x) - \frac{1}{6}\Lambda_3^n(x),$$
$$\Pi_2^n(x) = -\frac{1}{2}\Lambda_1^n(x) - \frac{1}{6}\Lambda_2^n(x) + \frac{8}{6}\Lambda_3^n(x) - \frac{1}{6}\Lambda_4^n(x),$$
$$\Pi_3^n(x) = \frac{1}{9}\Lambda_1^n(x) - \frac{1}{6}\Lambda_3^n(x) + \frac{8}{6}\Lambda_4^n(x) - \frac{1}{6}\Lambda_5^n(x),$$
$$\Pi_j^n(x) = -\frac{1}{6}\Lambda_j^n(x) + \frac{8}{6}\Lambda_{j+1}^n(x) - \frac{1}{6}\Lambda_{j+2}^n(x),\ j = 4, 5, \ldots, N_x - 4,$$
$$\Pi_{N_x-3}^n(x) = -\frac{1}{6}\Lambda_{N_x-3}^n(x) + \frac{8}{6}\Lambda_{N_x-2}^n(x) - \frac{1}{6}\Lambda_{N_x-1}^n(x) + \frac{1}{9}\Lambda_{N_x+1}^n(x),$$
$$\Pi_{N_x-2}^n(x) = -\frac{1}{6}\Lambda_{N_x-2}^n(x) + \frac{8}{6}\Lambda_{N_x-1}^n(x) - \frac{1}{6}\Lambda_{N_x}^n(x) - \frac{1}{2}\Lambda_{N_x+1}^n(x),$$
$$\Pi_{N_x-1}^n(x) = -\frac{1}{6}\Lambda_{N_x-1}^n(x) + \frac{8}{6}\Lambda_{N_x}^n(x) + \Lambda_{N_x+1}^n(x),$$

where Rh_m^n is obtained by substituting U_m^i instead of u_m^i in $R^n(x_m)$, and

$$\Lambda_j^n(x) = Q^n(x)B_{j-3}(x) + \frac{b\Delta t}{2}B'_{j-3}(x) - \frac{a\Delta t}{2}B''_{j-3}(x).$$

The values of $B_j^{(\iota)}(x_m)$ for $\iota = 0, 1, 2$ can be obtained from (2.10) as follows

$$\mathbf{B} = \frac{1}{12}\begin{pmatrix} 12 & & & & & & & & \\ 3 & 7 & 2 & & & & & & \\ 2 & 4 & 2 & & & & & & \\ & \ddots & \ddots & \ddots & & & & & \\ & & 2 & 4 & 2 & & & & \\ & & & 2 & 7 & 3 & & & \\ & & & & & & 12 & & \end{pmatrix},$$

$$\mathbf{B}' = \frac{1}{4\Delta x}\begin{pmatrix} -12 & 12 & & & & & & & \\ & -3 & 1 & 2 & & & & & \\ & & -2 & 0 & 2 & & & & \\ & & & \ddots & \ddots & \ddots & & & \\ & & & & -2 & 0 & 2 & & \\ & & & & & -2 & -1 & 3 & \\ & & & & & & & -12 & 12 \end{pmatrix},$$

$$\mathbf{B}'' = \frac{1}{2(\Delta x)^2}\begin{pmatrix} 12 & -18 & 6 & & & & & & \\ & 3 & -5 & 2 & & & & & \\ & & 2 & -4 & 2 & & & & \\ & & & \ddots & \ddots & \ddots & & & \\ & & & & 2 & -4 & 2 & & \\ & & & & & 2 & -5 & 3 & \\ & & & & & & 6 & -18 & 12 \end{pmatrix},$$

where

$$(\mathbf{B})_{m+1,j+4} = B_j(x_m),\ (\mathbf{B}')_{m+1,j+4} = {B'}_j(x_m),\ (\mathbf{B}'')_{m+1,j+4} = {B''}_j(x_m),$$
$$m = 0, 1, \ldots, N_x, \text{ and } j = -3, -2, \ldots, N_x - 1.$$

The algorithm for implementation of the presented method is given in Algorithm 1.

Algorithm 1 Numerical solution of problem (1.2)–(1.4) using presented method.

input f, w, x_l, x_r, α_l, α_r, σ, r, ψ, ρ, N_t, N_x, N_α, T, and compute Δx, $\Delta\alpha$, Δt, a, b, c, ψ', ψ'' and x, t, α as vectors in their domains.
Initialize the first columns of U, U_x and U_{xx} matrices using ψ, ψ' and ψ'' functions,
for n $= 0$ to $N_t - 1$ **do**
  for m $= 1$ to $N_x - 1$ **do**
    $Q(m) = Q^n(x_m)$ from (2.6),
    $V(i, m) = V_i^n(x_m)$ from (2.8),
    $R(m, 1) = R^n(x_m)$ from (2.7),
    for j $= 1$ to $N_x - 1$ **do**
      $A(m, j) = \Pi_j^n(x_m)$ witch is computed using $\mathbf{B}$, $\mathbf{B}'$, $\mathbf{B}''$, $\Lambda_j^n(x_m)$ and definitions of $\Pi_j^n(x_m)$s,
    end for
  end for
  solve the linear system $AU_{temp} = R$,
  Put U_{temp} as the $(n+2)$th column of matrix U,
  Calculate $(n+2)$th columns of U_x and U_{xx} matrices using (2.11).
end for

3. Error analysis

Lemma 3.1. *Let $|w(x,t,\alpha)| < w^*$ the for $j = 1, 2, \ldots, N_x + 1$, $i = 0, 1, \ldots, n-1$ and $n = 1, 2, \ldots, N_t$ we have*

$$\left|\Pi_j^n(x)\right| \le \frac{13}{6} + \frac{13}{18}\rho w^*(\Delta t)^{(1-\alpha_{N_\alpha})} + \frac{15b\Delta t}{4\Delta x} + \frac{171a\Delta t}{12(\Delta x)^2},$$
$$|V_i^n| \le \frac{6w^*(\Delta t)^{(1-\alpha_{N_\alpha})}}{\Delta\alpha}.$$

Proof. According to definition of $\mathbf{B}$, $\mathbf{B}'$, and $\mathbf{B}''$ we obtain

$$\left|\Lambda_j^n\right| \le |Q^n| + \frac{3b\Delta t}{2\Delta x} + \frac{9a\Delta t}{2(\Delta x)^2}. \tag{3.1}$$

Therefore, according to the definition of $\Pi_j^n(x)$s we can conclude that

$$\begin{cases} |\Pi_1^n| \le \frac{5}{2}|Q^n| + \frac{15b\Delta t}{4\Delta x} + \frac{45a\Delta t}{4(\Delta x)^2},\ |\Pi_2^n| \le \frac{13}{6}|Q^n| + \frac{39b\Delta t}{12\Delta x} + \frac{171a\Delta t}{12(\Delta x)^2}, \\ |\Pi_3^n| \le \frac{16}{9}|Q^n| + \frac{48b\Delta t}{18\Delta x} + \frac{144a\Delta t}{18(\Delta x)^2}, \\ |\Pi_j^n| \le \frac{5}{3}|Q^n| + \frac{15b\Delta t}{6\Delta x} + \frac{45a\Delta t}{6(\Delta x)^2},\ j = 4, 5, \ldots, N_x - 4, \\ |\Pi_{Nx-3}^n| \le \frac{16}{9}|Q^n| + \frac{48b\Delta t}{18\Delta x} + \frac{144a\Delta t}{18(\Delta x)^2},\ |\Pi_{Nx-2}^n| \le \frac{13}{6}|Q^n| + \frac{39b\Delta t}{12\Delta x} + \frac{171a\Delta t}{12(\Delta x)^2}, \\ |\Pi_{Nx-1}^n| \le |Q^n| + \frac{3b\Delta t}{2\Delta x} + \frac{9a\Delta t}{2(\Delta x)^2}. \end{cases} \tag{3.2}$$

From (3.2), we get

$$|\Pi_j^n| \le \frac{13}{6}\left|1 + \frac{\rho\Delta\alpha}{4}X_n^{n+1}\right| + \frac{15b\Delta t}{4\Delta x} + \frac{171a\Delta t}{12(\Delta x)^2}. \tag{3.3}$$

Since $J_{n,n+1}^{\alpha_k} = \frac{(\Delta t)^{(1-\alpha_k)}}{1-\alpha_k}$ and for $0 < \Delta t, \alpha_k < 1$, $(\Delta t)^{(1-\alpha_{N_\alpha})} > (\Delta t)^{(1-\alpha_k)}$ and $\frac{1}{\Gamma(2-\alpha_k)} < \frac{3}{2}$, we have

$$\begin{aligned} |X_n^{n+1}(x)| &\le \frac{2(\Delta t)^{(1-\alpha_{N_\alpha})}}{3}\left(|w_0^{n+1}(x)| + 2\sum_{k=1}^{N_\alpha-1}|w_k^{n+1}(x)| + |w_{N_\alpha}^{n+1}(x)|\right) \\ &\le \frac{4(\Delta t)^{(1-\alpha_{N_\alpha})}w^*}{3}N_\alpha \le \frac{4(\Delta t)^{(1-\alpha_{N_\alpha})}w^*}{3\Delta\alpha}. \end{aligned} \tag{3.4}$$

From Equations (3.4) and (3.3), the desired result is confirmed. The second inequality in this lemma is confirmed according to definition X_i^n. □

Theorem 3.2. *Let* $E_j^n = u_j^n - U_j^n$, $\|E^n\|_{x,\infty} = \max_{1\le j\le N_x-1}|E_j^n|$,

$$\begin{aligned} \varepsilon_1 &= \left(78(\Delta x)^2 + 26\rho w^*(\Delta t)^{(1-\alpha_{N_\alpha})}(\Delta x)^2 + 135b\Delta t\Delta x + 513a\Delta t\right)(N_x+1), \\ \varepsilon_2 &= (\Delta x)^2\left|O((\Delta\alpha)^2(\Delta t) + (\Delta\alpha)(\Delta t)^2 + (\Delta t)^2 + \Delta t\Delta x)\right|,\ \varepsilon_3 = 54\rho w^*(\Delta x)^2(\Delta t)^{(1-\alpha_{N_\alpha})}, \end{aligned}$$

and $\xi_1 = \frac{\varepsilon_2}{\varepsilon_1}$, $\xi_2 = \frac{\varepsilon_3}{\varepsilon_1}$, *in this case if* $n\xi_1$ *and* ξ_2 *tends to zeros then* $\|E^n\|_{x,\infty} \to 0$.

Proof. By substituting (2.11) in (2.5) and (2.9), we obtain

$$\begin{aligned} &\sum_{j=1}^{N_x+1}\mu_j(u^{n+1})\left(Q^n(x_m)B_{j-3}(x_m) + \frac{b\Delta t}{2}B'_{j-3}(x_m) - \frac{a\Delta t}{2}B''_{j-3}(x_m)\right) \\ &= \sum_{j=1}^{N_x+1}\mu_j(u^n)\left(Q^n(x_m)B_{j-3}(x_m) - \frac{b\Delta t}{2}B'_{j-3}(x_m) + \frac{a\Delta t}{2}B''_{j-3}(x_m)\right) \\ &- \frac{\rho\Delta\alpha}{4}\sum_{i=0}^{n-1}V_i^n(x_m)(u_m^{i+1} - u_m^i) - c\Delta t + \Delta t f(x_m, t_{n+\frac{1}{2}}) \\ &+ O((\Delta\alpha)^2(\Delta t) + (\Delta\alpha)(\Delta t)^2 + (\Delta t)^2 + \Delta t\Delta x), \end{aligned} \tag{3.5}$$

and

$$\sum_{j=1}^{N_x+1}\mu_j(U^{n+1})\left(Q^n(x_m)B_{j-3}(x_m)+\frac{b\Delta t}{2}B'_{j-3}(x_m)-\frac{a\Delta t}{2}B''_{j-3}(x_m)\right)$$
$$=\sum_{j=1}^{N_x+1}\mu_j(U^n)\left(Q^n(x_m)B_{j-3}(x_m)-\frac{b\Delta t}{2}B'_{j-3}(x_m)+\frac{a\Delta t}{2}B''_{j-3}(x_m)\right)$$
$$-\frac{\rho\Delta\alpha}{4}\sum_{i=0}^{n-1}V_i^n(x_m)(U_m^{i+1}-U_m^i)-c\Delta t+\Delta t f(x_m,t_{n+\frac{1}{2}}). \tag{3.6}$$

The difference of (3.5) from (3.6) leads to the following result

$$\sum_{j=1}^{N_x+1}\Pi_j^n(x_m)E_j^{n+1}=\sum_{j=1}^{N_x+1}\Pi'^n_j(x_m)E_j^n-\frac{\rho\Delta\alpha}{4}\sum_{i=0}^{n-1}V_i^n(x_m)(E_m^{i+1}-E_m^i)$$
$$+O((\Delta\alpha)^2(\Delta t)+(\Delta\alpha)(\Delta t)^2+(\Delta t)^2+\Delta t\Delta x), \tag{3.7}$$

where Π'^n_j is similar to Π_j^n, using $\Lambda'^n_j=Q^n(x_m)B_{j-3}(x_m)-\frac{b\Delta t}{2}B'_{j-3}(x_m)+\frac{a\Delta t}{2}B''_{j-3}(x_m)$ instead of Λ_j^n is obtained. From Lemma 3.1, we know that

$$\left|\sum_{j=1}^{N_x+1}\Pi_j^n(x_m)E_j^{n+1}\right|\le\sum_{j=1}^{N_x+1}\left|\Pi_j^n(x_m)\right|\left|E_j^{n+1}\right|$$
$$\le(N_x+1)\max_{1\le j\le N_x+1}\left|\Pi_j^n(x_m)\right|\left\|E^{n+1}\right\|_{x,\infty}$$
$$\le\left(\frac{13}{6}+\frac{13}{18}\rho w^*(\Delta t)^{(1-\alpha_{N_\alpha})}+\frac{15b\Delta t}{4\Delta x}+\frac{171a\Delta t}{12(\Delta x)^2}\right)(N_x+1)\left\|E^{n+1}\right\|_{x,\infty}. \tag{3.8}$$

Now, we assume that

$$\left(\frac{13}{6}+\frac{13}{18}\rho w^*(\Delta t)^{(1-\alpha_{N_\alpha})}+\frac{15b\Delta t}{4\Delta x}+\frac{171a\Delta t}{12(\Delta x)^2}\right)(N_x+1)\left\|E^{n+1}\right\|_{x,\infty}$$
$$\le\left|\sum_{j=1}^{N_x+1}\Pi'^n_j(x_m)E_j^n-\frac{\rho\Delta\alpha}{4}\sum_{i=0}^{n-1}V_i^n(x_m)(E_m^{i+1}-E_m^i)\right.$$
$$\left.+O((\Delta\alpha)^2(\Delta t)+(\Delta\alpha)(\Delta t)^2+(\Delta t)^2+\Delta t\Delta x)\right|. \tag{3.9}$$

From (3.9) and carrying out a process similar to the proof of Lemma 3.1, we have

$$\left(\frac{13}{6}+\frac{13}{18}\rho w^*(\Delta t)^{(1-\alpha_{N_\alpha})}+\frac{15b\Delta t}{4\Delta x}+\frac{171a\Delta t}{12(\Delta x)^2}\right)(N_x+1)\left\|E^{n+1}\right\|_{x,\infty}$$
$$\le\left(\frac{13}{6}+\frac{13}{18}\rho w^*(\Delta t)^{(1-\alpha_{N_\alpha})}+\frac{15b\Delta t}{4\Delta x}+\frac{171a\Delta t}{12(\Delta x)^2}\right)(N_x+1)\|E^n\|_{x,\infty}$$
$$+\frac{\rho\Delta\alpha}{4}\left(\frac{6w^*(\Delta t)^{(1-\alpha_{N_\alpha})}}{\Delta\alpha}\right)\sum_{i=0}^{n-1}\left|E_m^{i+1}-E_m^i\right|$$
$$+\left|O((\Delta\alpha)^2(\Delta t)+(\Delta\alpha)(\Delta t)^2+(\Delta t)^2+\Delta t\Delta x)\right|. \tag{3.10}$$

Hence,

$$\left\|E^{n+1}\right\|_{x,\infty} \leq \left\|E^{n}\right\|_{x,\infty} + \xi_2 \sum_{i=0}^{n-1} \left(\left\|E^{i+1}\right\|_{x,\infty} + \left\|E^{i}\right\|_{x,\infty}\right) + \xi_1. \tag{3.11}$$

Since $\left\|E^{0}\right\|_{x,\infty} = 0$, we have

$$\left\|E^{1}\right\|_{x,\infty} \leq \xi_1, \quad \left\|E^{2}\right\|_{x,\infty} \leq \xi_1 \left(\xi_2 + 2\right), \quad \left\|E^{3}\right\|_{x,\infty} \leq \xi_1 \left(\left(\xi_2\right)^2 + 5\xi_2 + 3\right),$$
$$\left\|E^{4}\right\|_{x,\infty} \leq \xi_1 \left(\left(\xi_2\right)^3 + 8\left(\xi_2\right)^2 + 14\xi_2 + 4\right),$$
$$\left\|E^{5}\right\|_{x,\infty} \leq \xi_1 \left(\left(\xi_2\right)^4 + 11\left(\xi_2\right)^3 + 34\left(\xi_2\right)^2 + 30\xi_2 + 5\right), \ldots.$$

In other words,

$$\left\|E^{n+1}\right\|_{x,\infty} \leq \xi_1 \left(\sum_{k=1}^{n} a_k \left(\xi_2\right)^k + n\right). \tag{3.12}$$

Then, relation (3.12) concludes what we wanted to show.

□

4. Numerical results

In this section, we study the numerical results of the implementation of the presented method on an example. The versatility and the accuracy of the proposed method is measured using the L_2^n and L_∞^n error norms for the test problems. The error norms are defined as

$$L_2(t) = \sqrt{\Delta x \sum_{m=0}^{N_x} \left|u_m(t) - U_m(t)\right|^2}, \qquad L_\infty(t) = \max_{0 \leq m \leq N_x} \left|u_m(t) - U_m(t)\right|.$$

Example 4.1. In this example, we consider $\rho = 1$, $x_l = 0$, $x_r = \pi$, $\alpha_l = 0.2$, $\alpha_r = 0.8$, $r = 0.05$, $\sigma = 0.5$, $\psi(x) = \sin(x)$ and $w(x,t,\alpha) = 0.01\exp(\alpha)xt$. The exact solution is assumed to be $u(x,t) = (t^2 - t + 1)\sin(x)$ and $f(x,t)$ can be obtained from exact solution.

The norm of errors and the order of convergence for different values of N_x, N_t, and N_α are reported in Table 1. Also, the numerical solution and its related absolute errors of Example 4.1 are shown in Figure 1.

Table 1. Error norms in Example 4.1 with $N_x = N_t = N_\alpha = 50$.

t	$L_2(t)$	$L_\infty(t)$
1	$1.887022e-05$	$1.757039e-05$
2	$3.312101e-05$	$3.130544e-05$
3	$6.725444e-05$	$6.533021e-05$
4	$1.527224e-04$	$1.427221e-04$
5	$3.260535e-04$	$2.866728e-04$
6	$6.116438e-04$	$5.175115e-04$
7	$1.028845e-03$	$8.517188e-04$
8	$1.594779e-03$	$1.303750e-03$
9	$2.325069e-03$	$1.885706e-03$
10	$3.234265e-03$	$2.611468e-03$

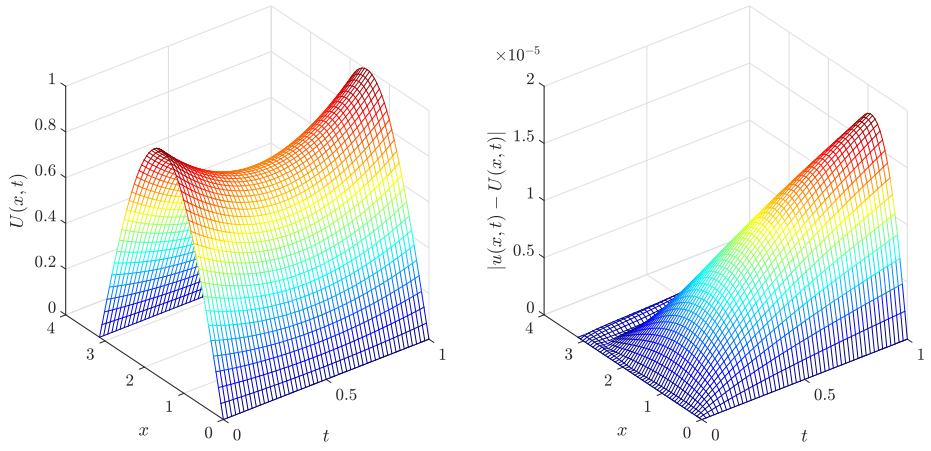

FIGURE 1. Numerical solution (left) and the absolute errors for all mesh points in Example 4.1 with $N_t = 50$, $N_x = 50$ and $N_\alpha = 50$.

Example 4.2. In this example, we consider $\rho = 1$, $x_l = 0$, $x_r = 1$, $\alpha_l = 0.1$, $\alpha_r = 0.6$, $r = 0.02$, $\sigma = 0.1$, $\psi(x) = x - x^2$ and $w(x,t,\alpha) = 0.1\Gamma(1-\alpha)(x+1)(t+1)$. The exact solution is assumed to be $u(x,t) = (1+t^2)(x-x^2)$ and $f(x,t)$ can be obtained from exact solution.

The norm of errors and the order of convergence for different values of N_x, N_t and N_α are reported in Table 2. Also, the numerical solution and its related absolute errors of Example 4.2 are shown in Figure 2.

TABLE 2. Error norms in Example 4.2 with $N_x = N_t = N_\alpha = 50$.

t	$L_2(t)$	$L_\infty(t)$
1	$6.572882e-07$	$1.104885e-06$
2	$1.272913e-06$	$2.496485e-06$
3	$4.732949e-06$	$7.050678e-06$
4	$1.304282e-05$	$1.840373e-05$
5	$2.784658e-05$	$3.902088e-05$
6	$5.112576e-05$	$7.152497e-05$
7	$8.510130e-05$	$1.189414e-04$
8	$1.322167e-04$	$1.847418e-04$
9	$1.951248e-04$	$2.724979e-04$
10	$2.766686e-04$	$3.860744e-04$

5. CONCLUSION

In this work, we proposed a novel numerical scheme to solve the generalized distributed-order time-fractional Black-Scholes equation implicitly. Our approach combines finite differences for approximating time derivatives and cubic B-spline quasi-interpolation for spatial derivatives. Additionally, we investigated the error analysis of the presented method, demonstrating its effectiveness and accuracy through numerical results.

While our numerical scheme presents a significant advancement in solving the generalized distributed-order time-fractional Black-Scholes equation, several avenues for further research and unresolved challenges remain. One key area for future work is exploring alternative types of derivatives. For instance, the local generalized derivative,

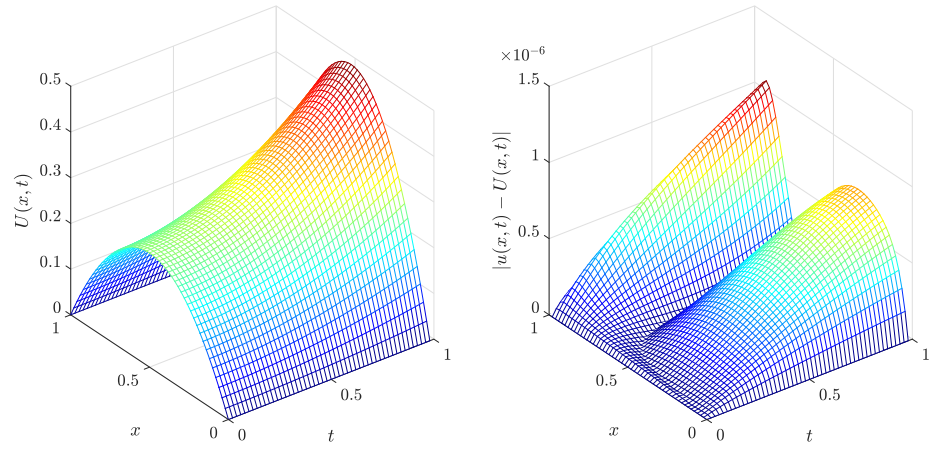

FIGURE 2. Numerical solution (left) and the absolute errors for all mesh points in Example 4.2 with $N_t = 50$, $N_x = 50$ and $N_\alpha = 50$.

as discussed in the paper [10], offers a promising avenue for improving the flexibility and robustness of numerical methods for fractional differential equations. Incorporating such derivatives could enhance the modeling of complex systems with heterogeneous memory properties and anomalous diffusion processes. Moreover, further investigation is warranted into the stability and convergence properties of the proposed scheme under various boundaries and initial conditions. Extending the method to higher-dimensional problems and more complex financial derivatives would provide a more comprehensive toolset for practical applications. Additionally, the development of adaptive mesh refinement techniques could enhance the computational efficiency and accuracy of the numerical solutions, particularly for problems exhibiting sharp gradients or localized phenomena. Another area of potential research is the integration of machine learning techniques to optimize the parameters and performance of the numerical scheme. By leveraging data-driven approaches, it may be possible to achieve more accurate and efficient solutions, particularly in cases where analytical solutions are intractable or unknown. In summary, our work lays the groundwork for future advancements in the numerical analysis of generalized distributed-order time-fractional differential equations. By addressing the unresolved challenges and exploring new types of derivatives, such as those highlighted in [10] researchers can further refine and expand the applicability of these methods to a broader range of complex, real-world problems.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgements

We are very grateful to anonymous referees for their careful reading and valuable comments which led to the improvement of this paper.

References

[1] H. Aminikhah and J. Alavi, *Applying cubic B-Spline quasi-interpolation to solve 1D wave equations in polar coordinates*, International Scholarly Research Notices, *2013* (2013), Article ID 710529.
[2] F. Black and M. Scholes, *The Pricing of Options and Corporate Liabilities*, Journal of Political Economy, *81*(3) (1973), 637–654.
[3] F. Broucke and L. Oparnica, *Distributed-order time-fractional wave equations*, Z. Angew. Math. Phys., *74*(1) (2023), 1.

[4] C. De Boor, *A practical guide to splines*, Revised edition, Springer-Verlag New York, Inc., (2001).
[5] W. Ding, S. Patnaik, S. Sidhardh, and F. Semperlotti, *Applications of Distributed-Order Fractional Operators: A Review*, Entropy, *23* (2021), 110.
[6] J. C. Hull, *Options, Futures and Other Derivatives*, Prentice Hall, (2005).
[7] M. N. Koleva and L. G. Vulkov, *Numerical solution of time-fractional Black-Scholes equation*, Comp. Appl. Math. *36* (2017), 1699–1715.
[8] Z. Li, K. Fujishiro, and G. Li, *An inverse problem for distributed order time-fractional diffusion equations*, arXiv:1707.02556.
[9] R. Merton, *Theory of Rational Option Pricing*, Bell Journal of Economics and Management Science, *4*(1) (1973), 141–183.
[10] J. E. Nápoles, P. M. Guzmán, L. M. Lugo, and A. Kashuri, *The local generalized derivative and Mittag Leffler function*, Sigma J Eng & Nat Sci, *38*(2) (2020), 1007–1017.
[11] F. Ndaïrou and D. F. M. Torres, *Distributed-Order Non-Local Optimal Control*, arXiv: 2010.11648.
[12] M. She, L. Li, R. Tang, et al, *A novel numerical scheme for a time fractional Black-Scholes equation*, J. Appl. Math. Comput., *66* (2021), 853–870.
[13] W. Wyss, *The fractional Black-Scholes equation*, Fract. Cal. Appl. Anal., *3*(1) (2000), 51–61.
[14] M. Zhang, J. Jia, and X. Zheng, *Numerical approximation and fast implementation to a generalized distributed-order time-fractional option pricing model*, Chaos, Solitons and Fractals, *170* (2023), 113353.