

IMPLICIT CUBIC B-SPLINE SCHEME FOR THE FRACTIONAL BLACK-SCHOLES MODEL WITH CAPUTO-HADAMARD DERIVATIVE

R. MONTAZERI  

Article type: Research Article

(Received: 07 April 2025, Received in revised form 16 September 2025)

(Accepted: 19 October 2025, Published Online: 19 October 2025)

ABSTRACT. In this study, we introduce a novel numerical scheme for solving the Black–Scholes equation endowed with a Caputo–Hadamard fractional time derivative. The temporal derivative is discretized via a finite-difference approach, ensuring both stability and high accuracy in capturing the model’s dynamics. For spatial discretization, we employ a cubic B-spline quasi-interpolation technique, which delivers a robust and efficient framework. A rigorous error analysis is carried out, establishing theoretical bounds that confirm the method’s reliability. Finally, we validate the proposed scheme through a series of numerical experiments, demonstrating its effectiveness in practical financial modeling and its ability to address the challenges posed by fractional differential equations.

Keywords: Caputo–Hadamard, Black–Scholes, Quasi-interpolation, Spline.
2020 MSC: 65M22, 65M70, 65M99

1. Introduction

In this study, we introduce a numerical framework that extends classical differentiation to noninteger orders, enabling the modeling of systems with memory, hereditary characteristics, and nonlocal interactions. Among the various fractional operators, the Caputo–Hadamard derivative stands out for its distinctive logarithmic kernel, which enhances both modeling fidelity and computational performance. By fusing the strengths of Liouville–Caputo and Hadamard formulations, this operator overcomes their individual limitations and broadens its range of practical applications [14].

The Hadamard derivative employs a logarithmic kernel ideally suited to scale-invariant phenomena but struggles with the formulation of physically meaningful initial conditions. In contrast, the Liouville–Caputo derivative naturally accommodates initial data yet sacrifices scale invariance. The Liouville–Caputo–Hadamard (LCH) derivative unites these two approaches: it preserves the Hadamard kernel’s scale invariance while adopting the Liouville–Caputo structure for proper initial-condition treatment. This synergy

✉ roya.mn@pnu.ac.ir, ORCID: 0000-0002-4971-2661

<https://doi.org/10.22103/jmmr.2025.25062.1787>

Publisher: Shahid Bahonar University of Kerman

How to cite: R. Montazeri, *Implicit cubic B-spline scheme for the fractional Black-Scholes model with Caputo-Hadamard derivative*, J. Mahani Math. Res. 2026; 15(1): 485-508.



© the Author(s)

has driven its adoption in financial mathematics, control theory, and signal processing [13].

The LCH fractional derivative delivers clear advantages over conventional operators. Its defining logarithmic kernel

$$\left(\ln \frac{t}{s}\right)^{n-\alpha-1},$$

guarantees scale invariance—absent in standard Liouville–Caputo derivatives that rely on power-law kernels—making it well suited to self-similar systems and multiscale phenomena [21]. Unlike the Riemann–Liouville derivative, which also fails to support natural initial conditions, the LCH operator reconciles these competing demands: it maintains scale invariance while ensuring a zero derivative of constants, in line with classical calculus. This property simplifies the specification and interpretation of initial data, enhancing its utility in physics and engineering applications [22]. Moreover, the logarithmic kernel alleviates computational challenges in numerical approximation of nonlocal interactions across logarithmic scales. As a result, the LCH derivative offers an efficient, unified framework for simulating both local and nonlocal dynamics, significantly expanding its applicability across diverse scientific and engineering problems [12].

The Liouville–Caputo–Hadamard fractional derivative is defined mathematically as follows [13]:

$${}_a^{LCH}D_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \int_a^t \left(\ln \frac{t}{s}\right)^{n-\alpha-1} \frac{d^n}{ds^n} f(s) ds,$$

where $n-1 < \alpha < n$, $\Gamma(\cdot)$ is the Gamma function and a is the lower bound of the domain.

The Liouville–Caputo–Hadamard (LCH) fractional derivative synthesizes two foundational operators by retaining the Hadamard derivative’s inherent scale invariance and embedding the Liouville–Caputo framework’s physically consistent initial-condition treatment. Defined through a logarithmic integration kernel, this operator seamlessly bridges scale-invariant phenomena and practical modeling requirements, establishing itself as a uniquely powerful tool in fractional calculus [13].

A central advantage of the LCH derivative is its handling of initial data: it mirrors classical differential equations by assigning a zero derivative to constant functions, thereby overcoming a major drawback of the standard Hadamard derivative and markedly broadening its practical utility. Additionally, its logarithmic kernel illuminates the behavior of systems with nonlocal interactions and memory effects, enabling more precise modeling of complex phenomena that elude traditional fractional operators [17].

By construction, the LCH derivative embodies nonlocality, capturing long-range dependencies characteristic of fractional dynamical systems. Its logarithmic integration kernel distinguishes it fundamentally from power-law-based

derivatives, delivering a rare combination of scale invariance and clear physical interpretability. This blend of properties makes the LCH derivative especially well suited for applications in financial mathematics, signal processing, and control system design [15].

In this work, we assume that $0 < \alpha < 1$, $t \in [0, T]$, and we use the Liouville-Caputo-type Hadamard fractional derivative of order α of the function $f(t)$ as follow

$$(1) \quad {}_a^{LCH}D_t^\alpha f(t) = \frac{1}{\Gamma(1-\alpha)} \int_a^t \left(\ln \frac{t}{s}\right)^{-\alpha} \frac{df(s)}{ds} ds.$$

In modern financial mathematics, the Black–Scholes equation stands as a foundational pillar, providing the essential framework for option pricing and derivative valuation. Recent advances have enriched this classical model by incorporating fractional derivatives, markedly enhancing its ability to capture the complex dynamics of contemporary markets—particularly their nonlocal dependencies, memory effects, and anomalous diffusion behaviors that traditional formulations fail to address.

Originally introduced by Black, Scholes, and Merton in their seminal 1973 paper [11], the Black–Scholes equation is a partial differential equation governing the valuation of European options. This landmark model rests on three key assumptions: a frictionless trading environment, constant volatility of the underlying asset, and a known, constant risk-free interest rate. It is expressed as

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

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

The Black-Scholes model postulates that the underlying asset's price dynamics follow geometric Brownian motion. The partial differential equation is solved subject to boundary conditions determined by the option type (call or put), yielding a theoretical option price that has become ubiquitous in financial practice [9].

Despite its transformative impact, the model's reliance on several simplifying assumptions constrains its practical effectiveness in real market conditions. Key limitations include [23]:

- A core limitation arises from the model's assumption of constant volatility - an oversimplification that contradicts empirical market behavior, where volatility demonstrates pronounced stochastic characteristics and time-varying dynamics.
- The model's continuous-path assumption represents another significant constraint, as it fails to accommodate sudden jumps or price discontinuities that are frequently observed in real market behavior. This limitation stems from the geometric Brownian motion framework, which

by construction excludes the leptokurtic features and tail events characteristic of empirical asset returns.

- The model's idealized market microstructure assumption - postulating perfectly frictionless trading with zero transaction costs, no tax implications, and unlimited liquidity - deviates substantially from observable market conditions. This simplification neglects critical frictions that materially impact hedging costs and arbitrage strategies in practice.

These well-documented limitations have spurred significant research into generalized Black-Scholes frameworks, particularly through the incorporation of fractional calculus.

The integration of fractional derivatives into option pricing models addresses several fundamental shortcomings by:

- Naturally incorporating memory effects and non-local dependencies
- Capturing heavy-tailed return distributions and volatility clustering patterns [23]
- Providing flexible modeling through operators like the Liouville-Caputo and Riemann-Liouville derivatives

This fractional approach better represents the complex dynamics observed in real financial markets, where asset prices exhibit persistent correlations and scale-invariant behavior that classical Brownian motion fails to capture.

The fractional Black-Scholes equation modifies the classical PDE by replacing the standard time derivative with a fractional derivative of order $0 < \alpha \leq 1$. A typical form of the fractional Black-Scholes equation is (see [1], [20], [2]):

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

Here, the fractional derivative introduces a non-local temporal component, allowing the model to account for historical market behavior [10].

Fractional derivatives introduce essential memory effects into pricing models, accounting for the persistent influence of historical market conditions on current price dynamics. This formulation provides three key advantages over classical approaches:

- Improved tail risk modeling: The heavy-tailed distributions generated by fractional operators better capture the probability of extreme market events, including financial crises and flash crashes.
- Adaptive volatility: Unlike classical models, fractional formulations naturally accommodate time-varying volatility structures observed in empirical markets.
- Market irregularity: These models prove particularly valuable in high-uncertainty environments or when pricing exotic instruments with complex risk profiles [6].

From a computational perspective, the non-local nature of fractional derivatives typically requires numerical solution methods. Finite difference schemes

(explicit/implicit) and spectral methods have shown particular effectiveness, as closed-form solutions remain analytically intractable in most cases.

In this work, our objective is to investigate the approximate solution of the fractional Black-Scholes model derived by [19] (also, see [5] and [18])

$$(2) \quad {}_0^{LCH}D_t^\alpha V(S, t) + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V(S, t)}{\partial S^2} + rS \frac{\partial V(S, t)}{\partial S} - rV(S, t) = 0, \\ 0 < \alpha < 1, \quad (S, t) \in (0, +\infty) \times (0, T),$$

with the following terminal and boundary conditions

$$(3) \quad V(S, T) = \eta(S),$$

and

$$(4) \quad V(0, t) = \gamma(t), \quad V(\infty, t) = \zeta(t),$$

respectively.

In the traditional Black–Scholes setting, asset prices evolve according to a memoryless, first-order time derivative. Empirical observations, however, exhibit long-range dependence, volatility clustering, heavy tails in return distributions, and trading activity that grows logarithmically. By substituting the standard time derivative with a Caputo–Hadamard fractional operator of order α (where $0 < \alpha < 1$), one embeds logarithmic memory and non-Markovian behavior directly into the pricing framework. This extension not only deepens the theoretical underpinnings of option valuation but also brings model dynamics into closer agreement with real-world market features. As a result, the fractional Black–Scholes equation accommodates a far richer array of price movements than its classical counterpart.

The fractional approach delivers several concrete benefits for practitioners:

- It captures subdiffusive price dynamics typical in low-volatility or illiquid markets.
- It models persistent volatility memory, improving long-term risk assessments.
- It accommodates logarithmic scaling of trading time, making it suitable for markets where activity accelerates or decelerates on a log-scale.
- It naturally incorporates path-dependent effects—crucial for pricing Asian options and explains observed price stickiness.
- It better reproduces the shape of the implied volatility surface under anomalous conditions, reducing mispricing and enhancing hedging performance.

On the technical side, the Caputo–Hadamard derivative introduces tunable delay and memory parameters via the fractional order α , granting the model extra flexibility. This leads to more accurate pricing of highly volatile underlyings and fortifies the theoretical foundation for derivative risk management in realistic settings.

Using change of variable $\tau = T - t$, $x = \ln(S)$ and assuming $V(S, t) = u(x, \tau)$ the relations (2)–(4) convert to (see Appendix)

$$\begin{aligned} {}_0^{LCH}D_t^\alpha u(x, \tau) &= \frac{\sigma^2}{2} u_{xx}(x, \tau) + \left(r - \frac{\sigma^2}{2}\right) u_x(x, \tau) - ru(x, \tau) + f(x, \tau), \\ (5) \quad (x, \tau) &\in (a, b) \times (0, T), \end{aligned}$$

with the following initial and boundary conditions

$$(6) \quad u(x, 0) = u_0(x),$$

and

$$(7) \quad u(a, \tau) = \gamma(\tau), \quad u(b, \tau) = \zeta(\tau).$$

It is worth noting that to apply the numerical solution, the initially infinite domain described in (5) is confined to a finite real domain $[a, b]$. Additionally, a source term $f(x, \tau)$ is introduced to derive an exact solution for validating the numerical approach.

2. Numerical Method

To approximate the Liouville-Caputo-Hadamard fractional derivative, we use a technique similar to L1 method (see [14] page 272). To do this, we first divide the interval $[0, T]$ uniformly using the points $t_n = n\Delta t$, $n = 0, 1, \dots, N$ where $\Delta t = \frac{T}{N}$ denotes the temporal step size. To derive a semi-discrete formulation for (1), the fractional derivative may be approximated using the forward scheme for the first-order derivative as shown below.

$$\begin{aligned} {}_a^{LCH}D_t^\alpha f(t) &= \frac{1}{\Gamma(1-\alpha)} \int_a^t \left(\ln \frac{t}{s}\right)^{-\alpha} \frac{df(s)}{ds} ds \\ &= \frac{1}{\Gamma(1-\alpha)} \sum_{i=0}^{n-1} \int_{t_i}^{t_{i+1}} (\ln t_n - \ln s)^{-\alpha} \left(\frac{f(t_{i+1}) - f(t_i)}{\Delta t} \right. \\ &\quad \left. + \left(s - t_{i+\frac{1}{2}}\right) f''(\zeta_j) \right) ds \\ &= \frac{1}{\Delta t \Gamma(1-\alpha)} \sum_{i=0}^{n-1} (f(t_{i+1}) - f(t_i)) \int_{t_i}^{t_{i+1}} (\ln t_n - \ln s)^{-\alpha} ds \\ &\quad + \frac{1}{\Gamma(1-\alpha)} \sum_{i=0}^{n-1} f''(\zeta_j) \int_{t_i}^{t_{i+1}} (\ln t_n - \ln s)^{-\alpha} \left(s - t_{i+\frac{1}{2}}\right) ds \\ &= \frac{1}{\Delta t \Gamma(1-\alpha)} \sum_{i=0}^{n-1} (f(t_{i+1}) - f(t_i)) J_{i,n} + \varepsilon_n, \end{aligned}$$

where

$$\begin{aligned} J_{i,n} &= \int_{t_i}^{t_{i+1}} (\ln t_n - \ln s)^{-\alpha} ds \\ &= t_n \left(\Gamma(1 - \alpha, \ln(\frac{n}{i+1})) - \Gamma(1 - \alpha, \ln(\frac{n}{i})) \right), \\ \varepsilon_n &= \frac{1}{\Gamma(1 - \alpha)} \sum_{i=0}^{n-1} f''(\zeta_j) \int_{t_i}^{t_{i+1}} (\ln t_n - \ln s)^{-\alpha} \left(s - t_{i+\frac{1}{2}} \right) ds, \end{aligned}$$

and $\Gamma(a, x) = \frac{1}{\Gamma(a)} \int_x^\infty t^{a-1} e^{-t} dt$ denotes the incomplete Gamma function available numerically in MATLAB. It is worth noting that $\Gamma(a, 0) = 1$ and $\Gamma(a, \infty) = 0$.

Hence,

$$\begin{aligned} {}^{LCH}_a D_t^\alpha f(t) &\approx \frac{1}{\Delta t \Gamma(1 - \alpha)} \sum_{i=0}^{n-2} (f(t_{i+1}) - f(t_i)) J_{i,n} \\ (8) \quad &+ \frac{1}{\Delta t \Gamma(1 - \alpha)} (f(t_n) - f(t_{n-1})) J_{n-1,n}, \end{aligned}$$

Also, we have

$$\begin{aligned} \varepsilon_n &= \frac{1}{\Gamma(1 - \alpha)} \sum_{i=0}^{n-1} f''(\zeta_i) \int_{t_i}^{t_{i+1}} (\ln t_n - \ln s)^{-\alpha} \left(s - t_{i+\frac{1}{2}} \right) ds \\ &= \frac{n(\Delta t)^2}{2\Gamma(1 - \alpha)} \sum_{i=0}^{n-1} f''(\zeta_i) \left[2^\alpha n C_1 + 2 \left(i + \frac{1}{2} \right) C_2 \right], \end{aligned}$$

where

$$\begin{aligned} C_1 &= \Gamma \left(1 - \alpha, 2 \ln \left(\frac{n}{i+1} \right) \right) - \Gamma \left(1 - \alpha, 2 \ln \left(\frac{n}{i} \right) \right), \\ C_2 &= \Gamma \left(1 - \alpha, \ln \left(\frac{n}{i} \right) \right) - \Gamma \left(1 - \alpha, \ln \left(\frac{n}{i+1} \right) \right). \end{aligned}$$

Therefore $|\varepsilon_n| \leq O(\Delta t)$.

By applying the parametric Crank-Nicholson method to equation (5) and using (8), we have

$$\begin{aligned}
 & \frac{1}{\Delta t \Gamma(1-\alpha)} \sum_{i=0}^{n-1} (u(x, \tau_{i+1}) - u(x, \tau_i)) J_{i,n+1} \\
 & + \frac{1}{\Delta t \Gamma(1-\alpha)} (u(x, \tau_{n+1}) - u(x, \tau_n)) J_{n,n+1} \\
 & = \theta \left(\frac{\sigma^2}{2} u_{xx}(x, \tau_{n+1}) + \left(r - \frac{\sigma^2}{2}\right) u_x(x, \tau_{n+1}) - ru(x, \tau_{n+1}) \right) \\
 & + (1-\theta) \left(\frac{\sigma^2}{2} u_{xx}(x, \tau_n) + \left(r - \frac{\sigma^2}{2}\right) u_x(x, \tau_n) - ru(x, \tau_n) \right) \\
 (9) \quad & + \theta f(x, \tau_{n+1}) + (1-\theta) f(x, \tau_n) + O(\Delta t).
 \end{aligned}$$

Building upon relation (9), a corresponding semi-discrete scheme can be derived. This scheme serves as an intermediate step in the numerical solution process, bridging the analytical framework with the fully discrete computational method. It systematically incorporates the temporal and spatial discretizations while maintaining the integrity of the underlying mathematical model. By employing this formulation, it becomes possible to approximate the solution of the problem efficiently, setting the foundation for further numerical analysis and implementation.

$$\begin{aligned}
 & A_n U^{n+1}(x) - B U_{xx}^{n+1}(x) - C U_x^{n+1}(x, \tau_{n+1}) \\
 & = - \sum_{i=0}^{n-1} (U^{i+1}(x) - U^i(x)) J_{i,n+1} \\
 (10) \quad & + D_n U^n(x) + E U_{xx}^n(x) + F U_x^n(x) + h_n(x),
 \end{aligned}$$

where $U^n(x) = U(x, \tau_n)$ is considered as an approximation of $u(x, \tau_n)$ and

$$\begin{aligned}
 A_n &= \Delta t \Gamma(1-\alpha) \theta r + J_{n,n+1}, \quad B = \Delta t \Gamma(1-\alpha) \theta \frac{\sigma^2}{2}, \\
 C &= \Delta t \Gamma(1-\alpha) \theta \left(r - \frac{\sigma^2}{2}\right), \quad D_n = J_{n,n+1} - \Delta t \Gamma(1-\alpha) (1-\theta) r, \\
 E &= \Delta t \Gamma(1-\alpha) (1-\theta) \frac{\sigma^2}{2}, \quad F = \Delta t \Gamma(1-\alpha) (1-\theta) \left(r - \frac{\sigma^2}{2}\right), \\
 h_n(x) &= \Delta t \Gamma(1-\alpha) (\theta f(x, \tau_{n+1}) + (1-\theta) f(x, \tau_n)).
 \end{aligned}$$

To approximate spatial derivatives, we use cubic spline quasi-interpolation. Cubic spline quasi-interpolation offers several significant advantages: it avoids solving linear systems by relying solely on function evaluations, provides high-order accuracy (typically fourth-order for functions in $C^4[a, b]$), ensures smooth approximates due to the C^2 continuity of cubic splines, and benefits from local support, enabling efficient and adaptive computation. Moreover, it integrates

well with numerical schemes and can be naturally extended to multivariate problems using tensor-product constructions (see [8]).

A mesh Ω which is equally divided by knots $x_i = a + i\Delta x$ into M subintervals $[x_i, x_{i+1}]$, $i = 0, 1, \dots, M-1$ such that $\Omega : (x_0, x_1, \dots, x_M)$ with $\Delta x = (b-a)/M$ is used. Also, let $S_4(\Omega)$ be the space of cubic splines on Ω . The corresponding set of cubic B-splines $\{B_1, B_2, \dots, B_{M+3}\}$, where is a basis for $S_4(\Omega)$, are defined as follows [4]

$$(11) \quad B_{j+4}(x) = \begin{cases} \frac{(x-x_j)^3}{(x_{j+1}-x_j)(x_{j+2}-x_j)(x_{j+3}-x_j)}, & x \in [x_j, x_{j+1}), \\ \frac{(x-x_j)^2(x_{j+2}-x)}{(x_{j+2}-x_j)(x_{j+2}-x_{j+1})(x_{j+3}-x_j)} \\ + \frac{(x-x_j)(x_{j+3}-x)(x-x_{j+1})}{(x_{j+3}-x_j)(x_{j+3}-x_{j+1})(x_{j+2}-x_{j+1})} \\ + \frac{(x_{j+4}-x)(x-x_{j+1})^2}{(x_{j+2}-x_{j+1})(x_{j+3}-x_{j+1})(x_{j+4}-x_{j+1})}, & x \in [x_{j+1}, x_{j+2}), \\ \frac{(x-x_j)(x_{j+3}-x)^2}{(x_{j+3}-x_j)(x_{j+3}-x_{j+1})(x_{j+3}-x_{j+2})} \\ + \frac{(x-x_{j+1})(x_{j+3}-x)(x_{j+4}-x)}{(x_{j+3}-x_{j+1})(x_{j+3}-x_{j+2})(x_{j+4}-x_{j+1})} \\ + \frac{(x_{j+4}-x)^2(x-x_{j+2})}{(x_{j+4}-x_{j+1})(x_{j+4}-x_{j+2})(x_{j+3}-x_{j+2})}, & x \in [x_{j+2}, x_{j+3}), \\ \frac{(x_{j+4}-x)^3}{(x_{j+4}-x_{j+1})(x_{j+4}-x_{j+2})(x_{j+4}-x_{j+3})}, & x \in [x_{j+3}, x_{j+4}), \\ 0, & \text{otherwise,} \end{cases}$$

for $j = -3, -2, \dots, M-1$ with additional node points $x_{-3} = x_{-2} = x_{-1} = a$ and $x_{M+3} = x_{M+2} = x_{M+1} = b$. Now, we consider the following B-spline quasi-interpolation approximation for $U^n(x)$ [7]

$$(12) \quad U^n(x) = \sum_{j=1}^{M+3} \mu_j(U^n(x)) B_j(x) + O((\Delta x)^4),$$

where

$$(13) \quad \begin{cases} \mu_1(U^n(x)) = U^n(x_0), \\ \mu_2(U^n(x)) = \frac{1}{18} (7U^n(x_0) + 18U^n(x_1) - 9U^n(x_2) + 2U^n(x_3)), \\ \mu_j(U^n(x)) = \frac{1}{6} (-U^n(x_{j-3}) + 8U^n(x_{j-2}) - U^n(x_{j-1})), \quad 3 \leq j \leq M+1, \\ \mu_{M+2}(U^n(x)) = \frac{1}{18} (2U^n(x_{M-3}) - 9U^n(x_{M-2}) \\ + 18U^n(x_{M-1}) + 7U^n(x_M)), \\ \mu_{M+3}(U^n(x)) = U^n(x_M). \end{cases}$$

By substituting (12) into (10), we have

$$(14) \quad \sum_{j=1}^{M+3} \mu_j(U^{n+1}(x)) P_j^n(x) = R^n(x),$$

where

$$\begin{aligned} R^n(x) &= J_{0,n+1}U^0(x) - \sum_{i=1}^{n-1} (J_{i-1,n+1} - J_{i,n+1})U^i(x) \\ &\quad - D_n U^n(x) + EU_{xx}^n(x) + FU_x^n(x) + h_n(x), \\ P_j^n(x) &= A_n B_j(x) - BB''_j(x) - CB'_j(x). \end{aligned}$$

By placing the node points x_k , $k = 1, 2, \dots, M-1$ in relations (14) and using compact support property of B-splines, we have:

$$(15) \quad \begin{aligned} &\mu_{k+1}(U^{n+1}(x))P_{k+1}^n(x_k) + \mu_{k+2}(U^{n+1}(x))P_{k+2}^n(x_k) \\ &+ \mu_{k+3}(U^{n+1}(x))P_{k+3}^n(x_k) = R^n(x_k). \end{aligned}$$

By putting different values of $k = 1, 2, \dots, M-1$ in (15) and using (13) and also applying boundary conditions (7), we have:

$$(16) \quad \mathbf{M}^n \mathbf{U}^{n+1} = \mathbf{R}^n,$$

where $\mathbf{U}^{n+1} = (U_1^{n+1}, U_2^{n+1}, \dots, U_{M-1}^{n+1})^T$, $U_k^{n+1} = U^{n+1}(x_k)$, $k = 1, \dots, M-1$ and

$$\mathbf{R}^n = \begin{pmatrix} R^n(x_1) - \Lambda_0^{n,1}\gamma(\tau_{n+1}) \\ R^n(x_2) - \Lambda_0^{n,2}\gamma(\tau_{n+1}) \\ R^n(x_3) \\ R^n(x_4) \\ \vdots \\ R^n(x_{M-3}) \\ R^n(x_{M-2}) - \Lambda_4^{n,M-2}\zeta(\tau_{n+1}) \\ R^n(x_{M-1}) - \Lambda_3^{n,M-1}\zeta(\tau_{n+1}) \end{pmatrix},$$

$$\mathbf{M}^n = \begin{pmatrix} \Lambda_1^{n,1} & \Lambda_2^{n,1} & \Lambda_3^{n,1} & & & & & & \\ \Lambda_1^{n,2} & \Lambda_2^{n,2} & \Lambda_3^{n,2} & \Lambda_4^{n,2} & & & & & \\ \Lambda_0^{n,3} & \Lambda_1^{n,3} & \Lambda_2^{n,3} & \Lambda_3^{n,3} & \Lambda_4^{n,3} & & & & \\ & \Lambda_0^{n,4} & \Lambda_1^{n,4} & \Lambda_2^{n,4} & \Lambda_3^{n,4} & \Lambda_4^{n,4} & & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & & \Lambda_0^{n,M-3} & \Lambda_1^{n,M-3} & \Lambda_2^{n,M-3} & \Lambda_3^{n,M-3} & \Lambda_4^{n,M-3} \\ & & & & \Lambda_0^{n,M-2} & \Lambda_1^{n,M-2} & \Lambda_2^{n,M-2} & \Lambda_3^{n,M-2} \\ & & & & & \Lambda_0^{n,M-1} & \Lambda_1^{n,M-1} & \Lambda_2^{n,M-1} \end{pmatrix},$$

and

$$\begin{aligned}
\Lambda_0^{n,1} &= \frac{7}{18}P_2^n(x_1) - \frac{1}{6}P_3^n(x_1), \quad \Lambda_0^{n,i} = -\frac{1}{6}P_{i+1}^n(x_i), \quad i = 2, 3, \dots, M-2, \\
\Lambda_0^{n,M-1} &= -\frac{1}{6}P_M^n(x_{M-1}) + \frac{1}{9}P_{M+2}^n(x_{M-1}), \\
\Lambda_1^{n,1} &= P_2^n(x_1) + \frac{4}{3}P_3^n(x_1) - \frac{1}{6}P_4^n(x_1), \\
\Lambda_1^{n,i} &= \frac{4}{3}P_{i+1}^n(x_i) - \frac{1}{6}P_{i+2}^n(x_i), \quad i = 2, 3, \dots, M-2, \\
\Lambda_1^{n,M-1} &= \frac{4}{3}P_M^n(x_{M-1}) - \frac{1}{6}P_{M+1}^n(x_{M-1}) - \frac{1}{2}P_{M+2}^n(x_{M-1}), \\
\Lambda_2^{n,1} &= -\frac{1}{2}P_2^n(x_1) - \frac{1}{6}P_3^n(x_1) + \frac{4}{3}P_4^n(x_1), \\
\Lambda_2^{n,i} &= -\frac{1}{6}P_{i+1}^n(x_i) + \frac{4}{3}P_{i+2}^n(x_i) - \frac{1}{6}P_{i+3}^n(x_i), \quad i = 2, 3, \dots, M-2, \\
\Lambda_2^{n,M-1} &= -\frac{1}{6}P_M^n(x_{M-1}) + \frac{4}{3}P_{M+1}^n(x_{M-1}) + P_{M+2}^n(x_{M-1}), \\
\Lambda_3^{n,i} &= -\frac{1}{6}P_{i+2}^n(x_i) + \frac{4}{3}P_{i+3}^n(x_i), \quad i = 2, 3, \dots, M-2, \\
\Lambda_3^{n,M-1} &= -\frac{1}{6}P_{M+1}^n(x_{M-1}) + \frac{7}{18}P_{M+2}^n(x_{M-1}), \\
\Lambda_4^{n,i} &= -\frac{1}{6}P_{i+3}^n(x_i), \quad i = 2, 3, \dots, M-2.
\end{aligned}$$

It is worth noting that the values of U_x and U_{xx} at t_0 are obtained using the initial condition (6).

3. Error analysis

Let us assume that $\tilde{U}(x, \tau)$ and $U(x, \tau)$ are the exact and approximate solutions of Scheme (15), respectively. In this case, we will have from relation (15)

$$\begin{aligned}
&\sum_{j=1}^{M+3} P_j^n(x) \mu_j(\tilde{U}(x, \tau_{n+1})) \\
&= \sum_{j=1}^{M+3} \left[\left(\sum_{i=0}^{n-1} J_{i,n+1} B_j(x) \left(\mu_j(\tilde{U}(x, \tau_i)) - \mu_j(\tilde{U}(x, \tau_{i+1})) \right) \right) \right. \\
(17) \quad &\quad \left. + Q_j^n(x) \mu_j(\tilde{U}(x, \tau_n)) \right] + h_n(x),
\end{aligned}$$

$$\begin{aligned}
& \sum_{j=1}^{M+3} P_j^n(x) \mu_j(U(x, \tau_{n+1})) \\
&= \sum_{j=1}^{M+3} \left[\left(\sum_{i=0}^{n-1} J_{i,n+1} B_j(x) (\mu_j(U(x, \tau_i)) - \mu_j(U(x, \tau_{i+1}))) \right) \right. \\
(18) \quad & \left. + Q_j^n(x) \mu_j(U(x, \tau_n)) \right] + h_n(x).
\end{aligned}$$

By subtracting (18) from (17) and assuming $E^n = \tilde{U}(x, \tau_n) - U(x, \tau_n)$, we obtain

$$\begin{aligned}
& \sum_{j=1}^{M+3} P_j^n(x) \mu_j(E^{n+1}) = \sum_{j=1}^{M+3} \left[\sum_{i=0}^{n-1} J_{i,n+1} B_j(x) \mu_j(E^i - E^{i+1}) \right. \\
(19) \quad & \left. + Q_j^n(x) \mu_j(E^n) \right].
\end{aligned}$$

Using (13), we have from the last equation:

$$\begin{aligned}
(20) \quad & \sum_{\kappa=0}^M V_{\kappa}(x) E_{\kappa}^{n+1} = \mu_{n,0} \left(J_{0,n+1} \sum_{\iota=0}^M \Psi_{\iota}(x) E_{\iota}^0 + \sum_{i=1}^{n-1} (J_{i,n+1} - J_{i-1,n+1}) \right. \\
(21) \quad & \left. \times \sum_{\iota=0}^M \Psi_{\iota}(x) E_{\iota}^i - J_{n-1,n+1} \sum_{\iota=0}^M \Psi_{\iota}(x) E_{\iota}^n \right) + \sum_{\kappa=0}^M V'_{\kappa}(x) E_{\kappa}^n,
\end{aligned}$$

where $E_k^n = E^n(x_k)$ and

$$\begin{aligned}
V_0(x) &= P_1^n(x) + \frac{7}{18} P_2^n(x) - \frac{1}{6} P_3^n(x), \quad V_1(x) = P_2^n(x) + \frac{4}{3} P_3^n(x) - \frac{1}{6} P_4^n(x), \\
V_2(x) &= -\frac{1}{2} P_2^n(x) - \frac{1}{6} P_3^n(x) - \frac{1}{6} P_5^n(x) + \frac{4}{3} P_4^n(x), \\
V_3(x) &= \frac{1}{9} P_2^n(x) + \frac{4}{3} P_5^n(x) - \frac{1}{6} P_4^n(x) - \frac{1}{6} P_6^n(x), \\
V_l(x) &= -\frac{1}{6} P_{l+1}^n(x) + \frac{4}{3} P_{l+2}^n(x) - \frac{1}{6} P_{l+3}^n(x), \quad l = 4, 5, \dots, M-4, \\
V_{M-3}(x) &= -\frac{1}{6} P_{M-2}^n(x) + \frac{4}{3} P_{M-1}^n(x) - \frac{1}{6} P_M^n(x) + \frac{1}{9} P_{M+2}^n(x), \\
V_{M-2}(x) &= -\frac{1}{6} P_{M-1}^n(x) + \frac{4}{3} P_M^n(x) - \frac{1}{6} P_{M+1}^n(x) - \frac{1}{2} P_{M+2}^n(x), \\
V_{M-1}(x) &= -\frac{1}{6} P_M^n(x) + \frac{4}{3} P_{M+1}^n(x) + P_{M+2}^n(x), \\
V_M(x) &= -\frac{1}{6} P_{M+1}^n(x) + \frac{7}{18} P_{M+2}^n(x) + P_{M+3}^n(x),
\end{aligned}$$

$$\begin{aligned}
 \Psi_0(x) &= B_1(x) + \frac{7}{18}B_2(x) - \frac{1}{6}B_3(x), & \Psi_1(x) &= B_2(x) + \frac{4}{3}B_3(x) - \frac{1}{6}B_4(x), \\
 \Psi_2(x) &= -\frac{1}{2}B_2(x) - \frac{1}{6}B_3(x) + \frac{4}{3}B_4(x) - \frac{1}{6}B_5(x), \\
 \Psi_3(x) &= \frac{1}{9}B_2(x) - \frac{1}{6}B_4(x) + \frac{4}{3}B_5(x) - \frac{1}{6}B_6(x), \\
 \Psi_l(x) &= -\frac{1}{6}B_{l+1}(x) + \frac{4}{3}B_{l+2}(x) - \frac{1}{6}B_{l+3}(x), \quad l = 4, 5, \dots, M-4, \\
 \Psi_{M-3}(x) &= -\frac{1}{6}B_{M-2}(x) + \frac{4}{3}B_{M-1}(x) - \frac{1}{6}B_M(x) + \frac{1}{9}B_{M+2}(x), \\
 \Psi_{M-2}(x) &= -\frac{1}{6}B_{M-1}(x) + \frac{4}{3}B_M(x) - \frac{1}{6}B_{M+1}(x) - \frac{1}{2}B_{M+2}(x), \\
 \Psi_{M-1}(x) &= -\frac{1}{6}B_M(x) + \frac{4}{3}B_{M+1}(x) + B_{M+2}(x), \\
 \Psi_M(x) &= -\frac{1}{6}B_{M+1}(x) + \frac{7}{18}B_{M+2}(x) + B_{M+3}(x), \\
 \mu_{n,0} &= \begin{cases} 1, n \neq 0, \\ 0, n = 0, \end{cases}
 \end{aligned}$$

where $V'_\kappa(x)$ is obtained by substituting $Q_j^n(x) = D_n B_j(x) + E B''_j(x) + F B'_j(x)$, $j = 1, 2, \dots, M+3$, for P_j s in the relations related to $V_\kappa(x)$.

Using the properties of B-splines, it can be shown that [3]

Lemma 3.1. *For the B-splines defined in relation (11), we have*

$$(22) \quad 0 \leq B_j(x) \leq 1, \quad -\frac{3}{\Delta x} \leq B'_j(x) \leq \frac{3}{\Delta x}, \quad -\frac{9}{(\Delta x)^2} \leq B''_j(x) \leq \frac{6}{(\Delta x)^2}.$$

According to the definition of J_n^i we can see that $0 \leq J_{n,n+1} \leq t_n \leq T$, so

$$(23) \quad \Delta t \Gamma(1 - \alpha) \theta r \leq A_n \leq \Delta t \Gamma(1 - \alpha) \theta r + T.$$

Using Lemma 3.1, relation (23) and definition of $P_n^j(x)$ we obtain

$$(24) \quad K_2 \leq P_j^n(x) \leq K_1,$$

where $K_1 = \Delta t \Gamma(1 - \alpha) \theta \left(r + T + \frac{9\sigma^2}{(\Delta x)^2} + \frac{3}{\Delta x} \left(r - \frac{\sigma^2}{2} \right) \right)$ and $K_2 = \Delta t \Gamma(1 - \alpha) \theta \left(\frac{1}{6}r - \frac{3\sigma^2}{(\Delta x)^2} - \frac{3}{\Delta x} \left(r - \frac{\sigma^2}{2} \right) \right)$. We assume that $r - \frac{\sigma^2}{2} > 0$ and $\frac{1}{6}r < \frac{3\sigma^2}{(\Delta x)^2} + \frac{3}{\Delta x} \left(r - \frac{\sigma^2}{2} \right)$ and therefore $-K_1 < K_2$. Now, from (24) and definitions of $V_\kappa(x)$ s we have

$$(25) \quad V_\kappa(x) \geq \frac{14}{9}K_2.$$

Using Lemma 3.1 and definitions of $\Psi_\iota(x)$ s we can see that

$$(26) \quad \max_{x \in [a, b], 0 \leq \iota \leq M} |\Psi_\iota(x)| \leq \frac{5}{2}.$$

Since $J_{n,n+1} \leq T$ from Lemma 3.1 we conclude that

$$(27) \quad Q_j^n(x) = D_n B_j(x) + E B''_j(x) + F B'_j(x) \leq G,$$

where $G = T + \Delta t \Gamma(1 - \alpha)(1 - \theta) \left(r + \frac{3\sigma^2}{(\Delta x)^2} + \left(r - \frac{\sigma^2}{2} \right) \frac{3}{\Delta x} \right)$. Hence, from definitions of $V'_\kappa(x)$ s we have

$$(28) \quad \max_{x \in [a,b], 0 \leq \kappa \leq M} |V'_\kappa(x)| \leq \frac{5}{2} G.$$

Also, from definitions of $J_{i,n}$ we can see that

$$(29) \quad |J_{i,n+1} - J_{i-1,n+1}| < 1, \quad |J_{n-1,n+1}| < 1, \quad |J_{0,n+1}| < 1.$$

Now, from (20) and using (24), we obtain

$$(30) \quad \begin{aligned} & \min_{x \in [a,b], 0 \leq \kappa \leq M} V_\kappa(x) \sum_{\kappa=0}^M E_\kappa^{n+1} \leq \mu_{n,0} \left(\max_{0 \leq n \leq N} |J_{0,n+1}| \right. \\ & \quad \times \max_{x \in [a,b], 0 \leq \iota \leq M} |\Psi_\iota(x)| \sum_{\iota=0}^M E_\iota^0 \\ & \quad + \max_{x \in [a,b], 0 \leq \iota \leq M} |\Psi_\iota(x)| \\ & \quad \times \max_{1 \leq i \leq n-1, 0 \leq n \leq N} |J_{i,n+1} - J_{i-1,n+1}| \sum_{\iota=0}^M \sum_{i=1}^{n-1} E_\iota^i \\ & \quad + \max_{0 \leq n \leq N} |J_{n-1,n+1}| \max_{x \in [a,b], 0 \leq \iota \leq M} |\Psi_\iota(x)| \sum_{\iota=0}^M E_\iota^n \Big) \\ & \quad + \max_{x \in [a,b], 0 \leq \kappa \leq M} |V'_\kappa(x)| \sum_{\kappa=0}^M E_\kappa^n. \end{aligned}$$

Therefore,

$$\begin{aligned}
 (31) \quad & \min_{x \in [a, b], 0 \leq \kappa \leq M} V_{\kappa}(x) \sum_{\kappa=0}^M E_{\kappa}^{n+1} \leq \mu_{n,0} \max_{x \in [a, b], 0 \leq \iota \leq M} |\Psi_{\iota}(x)| \left(\right. \\
 & \quad \max_{0 \leq n \leq N} |J_{0,n+1}| \sum_{\iota=0}^M E_{\iota}^0 \\
 & \quad + \max_{1 \leq i \leq n-1, 0 \leq n \leq N} |J_{i,n+1} - J_{i-1,n+1}| \sum_{\iota=0}^M \sum_{i=1}^{n-1} E_{\iota}^i \\
 & \quad \left. + \max_{0 \leq n \leq N} |J_{n-1,n+1}| \sum_{\iota=0}^M E_{\iota}^n \right) \\
 & \quad + \max_{x \in [a, b], 0 \leq \kappa \leq M} |V'_{\kappa}(x)| \sum_{\kappa=0}^M E_{\kappa}^n.
 \end{aligned}$$

We put

$$\begin{aligned}
 C &= \max \left\{ \begin{array}{l} \max_{0 \leq n \leq N} |J_{0,n+1}|, \max_{1 \leq i \leq n-1, 0 \leq n \leq N} |J_{i,n+1} - J_{i-1,n+1}| \\ \max_{0 \leq n \leq N} |J_{n-1,n+1}| \end{array} \right\}, \\
 \xi &= \min_{x \in [a, b], 0 \leq \kappa \leq M} V_{\kappa}(x).
 \end{aligned}$$

Now, from (31)

$$\begin{aligned}
 (32) \quad & \xi \sum_{\kappa=0}^M E_{\kappa}^{n+1} \leq \mu_{n,0} C \max_{x \in [a, b], 0 \leq \iota \leq M} |\Psi_{\iota}(x)| \\
 & \quad \times \sum_{\iota=0}^M \left(E_{\iota}^0 + \sum_{i=1}^{n-1} E_{\iota}^i + E_{\iota}^n \right) \\
 & \quad + \max_{x \in [a, b], 0 \leq \kappa \leq M} |V'_{\kappa}(x)| \sum_{\kappa=0}^M E_{\kappa}^n.
 \end{aligned}$$

Let $\vartheta = \max \left\{ C \max_{x \in [a, b], 0 \leq \iota \leq M} |\Psi_{\iota}(x)|, \max_{x \in [a, b], 0 \leq \kappa \leq M} |V'_{\kappa}(x)| \right\}$, from (32) we have

$$(33) \quad \xi \sum_{\kappa=0}^M E_{\kappa}^{n+1} \leq \vartheta \sum_{\iota=0}^M \left(\mu_{n,0} \left(E_{\iota}^0 + \sum_{i=1}^{n-1} E_{\iota}^i + E_{\iota}^n \right) + E_{\iota}^n \right).$$

According to the relation (33), we put

$$(34) \quad TE^n = \sum_{\iota=0}^M \left(\xi E_{\iota}^{n+1} - \vartheta \left(\mu_{n,0} \left(E_{\iota}^0 + \sum_{i=1}^{n-1} E_{\iota}^i + E_{\iota}^n \right) + E_{\iota}^n \right) \right).$$

By applying Taylor expansion, from relation (34) we obtain

$$\begin{aligned}
 TE^n &= \sum_{\iota=0}^M \left(\xi E_\iota^0 + \xi(n+1)O(\Delta t) \right. \\
 &\quad \left. - \vartheta \left(\mu_{n,0} \left(E_\iota^0 + \sum_{i=1}^{n-1} (E_\iota^0 + iO(\Delta t)) + E_\iota^0 + nO(\Delta t) \right) \right. \right. \\
 &\quad \left. \left. + E_\iota^0 + nO(\Delta t) \right) \right) \\
 &= (\xi - \vartheta (\mu_{n,0}(n+1) + 1)) \sum_{\iota=0}^M E_\iota^0 \\
 &\quad + (M+1) \left(\xi(n+1) - \vartheta \left(\frac{1}{2} \mu_{n,0} n(n+1) + n \right) \right) O(\Delta t) \\
 &\leq (|\xi| + \vartheta (\mu_{n,0}(n+1) + 1)) (M+1) \|E^0\|_{M,\infty} \\
 (35) \quad &+ (M+1) \left(|\xi| (n+1) + \vartheta \left(\frac{1}{2} \mu_{n,0} n(n+1) + n \right) \right) O(\Delta t),
 \end{aligned}$$

where $\|E^n\|_{M,\infty} = \max_{0 \leq \iota \leq M} |E_\iota^n|$. Using (26), (28) and (29) we have

$$\begin{aligned}
 \vartheta &= \max \left\{ C \max_{x \in [a,b], 0 \leq \iota \leq M} |\Psi_\iota(x)|, \max_{x \in [a,b], 0 \leq \kappa \leq M} |V'_\kappa(x)| \right\} \\
 (36) \quad &\leq \frac{5}{2} \max \{1, G\} = \frac{5}{2} G.
 \end{aligned}$$

Now, from (35) and by applying (25) and (36) we obtain

$$\begin{aligned}
 TE^n &\leq \left(\frac{14}{9} |K_2| + \frac{5}{2} G (\mu_{n,0}(n+1) + 1) \right) (M+1) \|E^0\|_{M,\infty} \\
 (37) \quad &+ (M+1) \left(\frac{14}{9} |K_2| (n+1) + \frac{5}{2} G \left(\frac{1}{2} \mu_{n,0} n(n+1) + n \right) \right) O(\Delta t).
 \end{aligned}$$

Hence, from (37) if $\frac{\Delta t}{(\Delta x)^2} \rightarrow 0$ then $TE^n \rightarrow 0$.

4. Numerical results

In this section, we present some numerical examples to illustrate the performance and accuracy of the proposed method in solving Problem (5)–(7). These examples are designed to demonstrate the efficiency, stability, and convergence behavior of the scheme under various settings. In particular, we compare the obtained results with exact solutions to validate the effectiveness of the approach. Additionally, we examine the error norms and convergence rates to confirm the theoretical properties discussed earlier.

In addition, in order to make a comparison with known numerical methods, we discretized the spatial derivatives using the central finite difference approximation and presented the numerical results in tabular form.

It is notable that, we perform all of the computations by MATLAB® R2019a software (V9.6.0.1072779, 64-bit (win64), License Number: 968398, MathWorks Inc., Natick, MA) running on a Sony VAIO Laptop (Intel® Core™ i5-2410M Processor 2.30 GHz with Turbo Boost up to 2.90 GHz, 8 GB of RAM, 64-bit) PC.

Example 4.1. *Let us consider $\sigma = 0.25$, $\theta = 1$, $r = 0.05$, $T = 1$, $a = 0$, $b = 1$, $\gamma(\tau) = \zeta(\tau) = 0$ with*

$$\begin{aligned} f(x, t) &= (t+1)^2 \left(-\frac{\sigma^2}{2}(2-6x) - \left(r - \frac{\sigma^2}{2}\right)(2x-3x^2) + rx^2(1-x) \right) \\ &\quad + \frac{2x^2(1-x)}{\Gamma(1-\alpha)}(t^2 2^{\alpha-1} + t). \\ u_0(x) &= x^2(1-x). \end{aligned}$$

From this assumptions the exact solution can be calculated by $u(x, t) = x^2(1-x)(1+t)^2$ [19].

The plot of the approximate solution and the absolute error at all node points with $N = 500$, $M = 100$ and $\alpha = 0.1$ are shown in Figure 1. The versatility and the accuracy of the method is measured using the maximum absolute error norm L_∞ , defined by

$$L_\infty(M, N) = \max_{0 \leq i \leq M} |U(x_i, t_N) - U_{exact}(x_i, t_N)|,$$

where $U(x_i, t_N)$ and $U_{exact}(x_i, t_N)$ are numerical and exact solutions at (x_i, t_N) point. The time convergence orders are defined as follows

$$\text{Rate}^t = \frac{\log(L_\infty(M, N_1)) - \log(L_\infty(M, N_2))}{\log(N_2) - \log(N_1)}.$$

Table 1 presents the method's error alongside the temporal order error for various values of α with $\theta = 0.5$. Also, Table 2 presents the numerical results obtained from using the central finite difference approximation to approximate spatial derivatives.

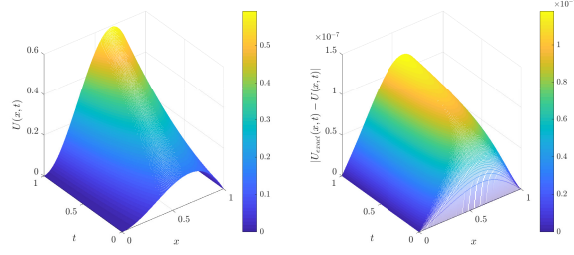


FIGURE 1. The graph of the approximate solution (left) and the absolute error at all node points (right) for example 4.1 with $N = 500$, $M = 100$ and $\alpha = \theta = 0.5$.

TABLE 1. The maximum-norm errors and temporal convergence orders with $M = 100$ for Example 4.1.

N	$\alpha = 0.9$		$\alpha = 0.5$		$\alpha = 0.1$		CPU time (Seconds)
	$L_\infty(M, N)$	Rate ^t	$L_\infty(M, N)$	Rate ^t	$L_\infty(M, N)$	Rate ^t	
10	1.420014e-04	—	1.399927e-03	—	2.120399e-03	—	0.27
20	4.135733e-05	1.779	5.029028e-04	1.477	9.900246e-04	1.099	0.55
40	1.189007e-05	1.798	1.802796e-04	1.480	4.620870e-04	1.099	1.04
80	3.384293e-06	1.812	6.430545e-05	1.487	2.156310e-04	1.099	3.97
160	9.556336e-07	1.824	2.286700e-05	1.491	1.006105e-04	1.099	9.51
320	2.681009e-07	1.833	8.115564e-06	1.494	4.693987e-05	1.099	29.34
640	7.481198e-08	1.841	2.876581e-06	1.496	2.189902e-05	1.099	105.43
1280	2.078170e-08	1.847	1.018761e-06	1.497	1.021645e-05	1.100	471.03
2560	5.750758e-09	1.853	3.606014e-07	1.498	4.765779e-06	1.100	1746.30

Example 4.2. Let us consider $\sigma = 0.2$, $\theta = 0.5$, $r = 0.02$, $T = 1$, $a = 0$, $b = 1$, with

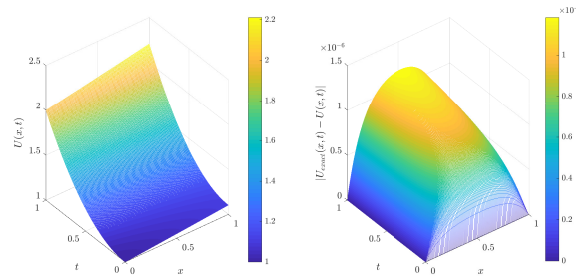
$$f(x, t) = \exp(\alpha x) \left(\frac{t^{2\alpha}}{\Gamma(1-\alpha)} + (t^2 + 1) \left(r - \frac{\alpha\sigma}{2} (r\sigma + \alpha\sigma) \right) \right).$$

$$u_0(x) = \exp(\alpha x).$$

From this assumptions the exact solution can be calculated by $u(x, t) = (t^2 + 1) \exp(\alpha x)$. The plot of the approximate solution and the absolute error at all node points with $N = 500$, $M = 100$ and $\alpha = 0.5$ are shown in Figure 2. Table 3 presents the method's error alongside the temporal order error for various values of α with $\theta = 0.5$. Also, Table 4 presents the numerical results obtained from using the central finite difference approximation to approximate spatial derivatives.

TABLE 2. The maximum-norm errors $L_\infty(M, N)$ with $M = 100$ using central finite difference approximation for Example 4.1.

N	$\alpha = 0.9$	$\alpha = 0.5$	$\alpha = 0.1$
10	$1.255499e-03$	$9.616565e-03$	$1.984517e-02$
20	$4.272821e-04$	$3.507691e-03$	$9.922583e-03$
40	$1.453155e-04$	$1.278562e-03$	$4.961291e-03$
80	$4.938651e-05$	$4.657158e-04$	$2.363157e-03$
160	$1.678435e-05$	$1.696369e-04$	$1.125617e-03$
320	$5.704281e-06$	$6.179019e-05$	$5.361526e-04$
640	$1.937297e-06$	$2.249147e-05$	$2.536156e-04$
1280	$6.570363e-07$	$8.175494e-06$	$1.216422e-04$
2560	$2.225259e-07$	$2.967619e-06$	$5.826267e-05$

FIGURE 2. The graph of the approximate solution (left) and the absolute error at all node points (right) for example 4.2 with $N = 500$, $M = 100$ and $\alpha = 0.5$.TABLE 3. The maximum-norm errors and temporal convergence orders with $M = 100$ for Example 4.2.

N	$\alpha = 0.9$		$\alpha = 0.5$		$\alpha = 0.1$		CPU time (Seconds)
	$L_\infty(M, N)$	Rate ^t	$L_\infty(M, N)$	Rate ^t	$L_\infty(M, N)$	Rate ^t	
5	$4.737141e-03$	—	$1.176424e-02$	—	$1.632291e-02$	—	0.12
10	$1.408346e-03$	1.750	$3.679856e-03$	1.676	$5.243402e-03$	1.638	0.20
20	$4.111604e-04$	1.776	$1.128937e-03$	1.704	$1.651542e-03$	1.666	0.36
40	$1.184392e-04$	1.796	$3.416836e-04$	1.724	$5.133132e-04$	1.685	1.16
80	$3.376751e-05$	1.810	$1.023863e-04$	1.738	$1.580206e-04$	1.699	3.19
160	$9.548660e-06$	1.822	$3.044691e-05$	1.749	$4.829915e-05$	1.710	7.11
320	$2.682184e-06$	1.832	$8.999802e-06$	1.758	$1.468149e-05$	1.718	26.55
640	$7.492502e-07$	1.840	$2.647231e-06$	1.765	$4.442851e-06$	1.724	121.62
1280	$2.083101e-07$	1.847	$7.752923e-07$	1.771	$1.338983e-06$	1.730	398.19

TABLE 4. The maximum-norm errors $L_\infty(M, N)$ with $M = 100$ using central finite difference approximation for Example 4.2.

N	$\alpha = 0.9$	$\alpha = 0.5$	$\alpha = 0.1$
5	$8.613114e - 03$	$7.315945e - 02$	$1.254858e - 01$
10	$2.947588e - 03$	$2.683367e - 02$	$5.287012e - 02$
20	$1.008027e - 03$	$9.835322e - 03$	$2.225999e - 02$
40	$3.444902e - 04$	$3.602435e - 03$	$9.365661e - 03$
80	$1.176469e - 04$	$1.318568e - 03$	$3.937775e - 03$
160	$4.017759e - 05$	$4.826242e - 04$	$1.655631e - 03$
320	$1.366411e - 05$	$1.759177e - 04$	$6.932179e - 04$
640	$4.647063e - 06$	$6.412240e - 05$	$2.902526e - 04$
1280	$1.581528e - 06$	$2.338897e - 05$	$1.216140e - 04$

In these two examples, it can be seen that better approximations are obtained for values of α close to 1.

Example 4.3. Let us consider $\sigma = 0.1$, $\theta = 1$, $r = 0.01$, $T = 2$, $a = 0$, $b = 5$, $\gamma(\tau) = 0$ and

$$\begin{aligned}\zeta(\tau) &= \exp(-rt) \max\{\exp(b) - K, 0\}, \\ u_0(x) &= \exp(-rT) \max\{\exp(x) - K, 0\},\end{aligned}$$

where $K = 10$ [16].

The figure of the approximate solution at all node points with $N = 200$, $M = 100$ and $\alpha = 0.1$ are shown in Figure 3. Also, in Figure 4 the approximate solution at all node points are shown for different values of K .

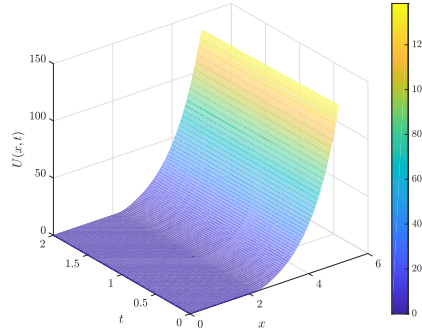


FIGURE 3. The graph of the approximate solution for example 4.3 with $N = 200$, $M = 100$, $K = 10$ and $\alpha = 0.1$.

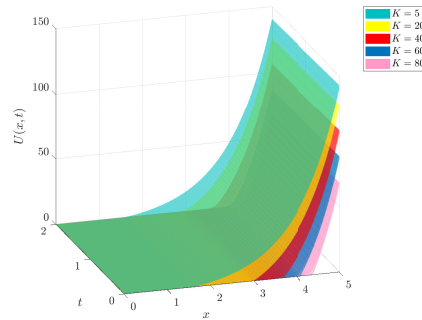


FIGURE 4. The graph of the approximate solution for different values of K with $N = 200$, $M = 100$ and $\alpha = 0.1$ in example 4.3.

5. Conclusion

In this study, we introduced a novel numerical scheme for the Black–Scholes model incorporating a Caputo–Hadamard fractional time derivative. The finite-difference approximation of the temporal fractional derivative ensures both accuracy and numerical stability in capturing the model’s dynamics. Spatial discretization via cubic B-spline quasi-interpolation delivers a robust and efficient algorithmic framework. Rigorous error analysis provides theoretical guarantees for the method’s reliability, and extensive numerical experiments confirm its effectiveness in addressing the challenges posed by fractional differential equations in financial modeling. The proposed method demonstrates superior performance in preserving accuracy under low-volatility and low-liquidity conditions, achieving faster convergence compared to classical approaches, and maintaining stability even with large time steps. Moreover, the alignment of simulation results with observed market behavior under anomalous conditions highlights the method’s potential for practical option pricing and risk management applications.

Future works.

- Extend the algorithm to support multi-dimensional fractional derivatives and apply it to multi-asset pricing models
- Develop higher-order temporal integration schemes to enhance time-accuracy while reducing computational cost
- Integrate the fractional framework with complex stochastic processes such as stochastic volatility models or jump-diffusion dynamics
- Design node-adaptive methods in both time and space to concentrate computational effort in regions with steep gradients

- Perform parameter estimation of fractional orders using real market data to broaden validation and enable deployment in automated trading systems

Appendix

If we put $\tau = T - t$ and $x = \ln(S)$, we get $\frac{d\tau}{dt} = -1$ and $\frac{dx}{dS} = \frac{1}{S}$. Since $V(S, t) = u(x, \tau)$, we have

$$\frac{\partial V}{\partial t} = \frac{\partial u}{\partial t} = \frac{\partial u}{\partial \tau} \frac{d\tau}{dt} = -\frac{\partial u}{\partial \tau},$$

and hence

$$\begin{aligned} {}_0^{LCH} D_t^\alpha V(S, t) &= \frac{1}{\Gamma(1-\alpha)} \int_a^t \left(\ln \frac{t}{y} \right)^{-\alpha} \frac{\partial V(S, y)}{dy} dy \\ (38) \quad &= -\frac{1}{\Gamma(1-\alpha)} \int_a^t \left(\ln \frac{t}{y} \right)^{-\alpha} \frac{\partial u(x, y)}{dy} dy = -{}_0^{LCH} D_t^\alpha u(x, \tau). \end{aligned}$$

Also,

$$(39) \quad \frac{\partial V}{\partial S} = \frac{\partial u}{\partial S} = \frac{\partial u}{\partial x} \frac{dx}{dS} = \frac{1}{S} \frac{\partial u}{\partial x},$$

and

$$\begin{aligned} \frac{\partial^2 V(S, t)}{\partial S^2} &= \frac{\partial}{\partial S} \left(\frac{\partial V}{\partial S} \right) = \frac{\partial}{\partial S} \left(\frac{1}{S} \frac{\partial u}{\partial x} \right) = \frac{1}{S} \frac{\partial}{\partial S} \left(\frac{\partial u}{\partial x} \right) + \left(\frac{-1}{S^2} \right) \frac{\partial u}{\partial x} \\ &= \frac{1}{S} \left(\frac{\partial}{\partial x} \left(\frac{\partial u}{\partial x} \right) \frac{dx}{dS} \right) - \left(\frac{1}{S^2} \right) \frac{\partial u}{\partial x} = \frac{1}{S^2} \frac{\partial^2 u}{\partial x^2} - \left(\frac{1}{S^2} \right) \frac{\partial u}{\partial x} \\ (40) \quad &= \frac{1}{S^2} \left(\frac{\partial^2 u}{\partial x^2} - \frac{\partial u}{\partial x} \right). \end{aligned}$$

By putting equations (38), (39), and (40) into the equation (2), we will have

$$-{}_0^{LCH} D_t^\alpha u(x, \tau) + \frac{\sigma^2}{2} S^2 \frac{1}{S^2} \left(\frac{\partial^2 u}{\partial x^2} - \frac{\partial u}{\partial x} \right) + rS \frac{1}{S} \frac{\partial u}{\partial x} - ru(x, \tau) = 0.$$

Therefore

$${}_0^{LCH} D_t^\alpha u(x, \tau) = \frac{\sigma^2}{2} \frac{\partial^2 u(x, \tau)}{\partial x^2} + \left(r - \frac{\sigma^2}{2} \right) \frac{\partial u(x, \tau)}{\partial x} - ru(x, \tau) = 0.$$

Data Availability Statement

Data supporting the findings of this study are available upon request from the corresponding author.

Conflict of interest

The authors declare that they have no conflicts of interest related to this manuscript.

References

- [1] Abdi, N., Aminikhah, H., & Sheikhan, A. R. (2022). High-order compact finite difference schemes for the time-fractional Black-Scholes model governing European options. *Chaos, Solitons & Fractals*, 162, 112423. <https://doi.org/10.1016/j.chaos.2022.112423>
- [2] Akram, T., Abbas, M., Abualnaja, K. M., Iqbal, A., & Majeed, A. (2022). An efficient numerical technique based on the extended cubic B-spline functions for solving time fractional Black-Scholes model. *Engineering with Computers*, 38(Suppl 2), 1705-1716. <https://doi.org/10.1007/s00366-021-01436-1>
- [3] Alavi, J., & Aminikhah, H. (2021). Numerical study of the boundary layer problem over a flat plate by orthogonal cubic spline basis functions. *The European Physical Journal Plus*, 136(7), 780. <https://doi.org/10.1140/epjp/s13360-021-01788-z>
- [4] Alavi, J., & Aminikhah, H. (2021). Orthogonal cubic spline basis and its applications to a partial integro-differential equation with a weakly singular kernel. *Computational and Applied Mathematics*, 40(2), 55. <https://doi.org/10.1007/s40314-021-01442-5>
- [5] Alazemi, F., Alsenafi, A., & Najafi, A. (2025). A spectral approach using fractional Jaiswal functions to solve the mixed time-fractional Black-Scholes European option pricing model with error analysis. *Numerical Algorithms*, 98(1), 347-371. <https://doi.org/10.1007/s11075-024-01797-w>
- [6] Aljethi, R. A., & Kılıçman, A. (2023). Analysis of fractional differential equation and its application to realistic data. *Chaos, Solitons & Fractals*, 171, 113446. <https://doi.org/10.1016/j.chaos.2023.113446>
- [7] Aminikhah, H., & Alavi, J. (2016). Numerical solution of convection-diffusion equation using cubic B-spline quasi-interpolation. *Thai Journal of Mathematics*, 14(3), 599-613. <http://thaijmath.in.cmu.ac.th/index.php/thaijmath/article/view/255>
- [8] Aminikhah, H., & Alavi, J. (2017). B-spline collocation and quasi-interpolation methods for boundary layer flow and convection heat transfer over a flat plate. *Calcolo*, 54, 299-317. <https://doi.org/10.1007/s10092-016-0188-x>
- [9] Aminikhah, H., & Alavi, S. J. (2019). A new approach to using the cubic B-spline functions to solve the Black-Scholes equation. <https://sanad.iau.ir/fa/Article/797725> [In Persian]
- [10] Bian, L., & Li, Z. (2021). Fuzzy simulation of European option pricing using sub-fractional Brownian motion. *Chaos, Solitons & Fractals*, 153, 111442. <https://doi.org/10.1016/j.chaos.2021.111442>
- [11] Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3), 637-654. <https://www.journals.uchicago.edu/doi/abs/10.1086/260062>
- [12] Deng, B., Zhang, J., & Zhang, Z. (2019). Superconvergence points of integer and fractional derivatives of special Hermite interpolations and its applications in solving FDEs. *ESAIM: Mathematical Modelling and Numerical Analysis*, 53(3), 1061-1082. <https://doi.org/10.1051/m2an/2019012>
- [13] Diethelm, K., & Ford, N. J. (2002). Analysis of fractional differential equations. *Journal of Mathematical Analysis and Applications*, 265(2), 229-248. <https://doi.org/10.1006/jmaa.2000.7194>
- [14] Guo, B., Pu, X., & Huang, F. (2015). *Fractional Partial Differential Equations And Their Numerical Solutions*. World Scientific.
- [15] Karami, S., Heydari, M. H., & Baleanu, D. (2024). Fractional logarithmic Chebyshev cardinal functions: Application for Caputo-Hadamard fractional optimal control problems. *Journal of Vibration and Control*, 10775463241272962. <https://doi.org/10.1177/10775463241272962>
- [16] Lee, Y., Kim, Y., & Lee, J. (2020). Pricing various types of power options under stochastic volatility. *Symmetry*, 12(11), 1911. <https://doi.org/10.3390/sym12111911>

- [17] Liu, Y., Zhu, Y., & Lu, Z. (2021). On Caputo-Hadamard uncertain fractional differential equations. *Chaos, Solitons & Fractals*, 146, 110894. <https://doi.org/10.1016/j.chaos.2021.110894>
- [18] Nagy, A. M., Makhlof, A. B., Alsenafi, A., & Alazemi, F. (2021). Combination synchronization of fractional systems involving the Caputo-Hadamard derivative. *Mathematics*, 9(21), 2781. <https://doi.org/10.3390/math9212781>
- [19] Nikan, O., Rashidinia, J., & Jafari, H. (2025). Numerically pricing American and European options using a time fractional Black-Scholes model in financial decision-making. *Alexandria Engineering Journal*, 112, 235-245. <https://doi.org/10.1016/j.aej.2024.10.083>
- [20] Pan, G., & Zhang, S. (2023). A meshless multiquadric quasi-interpolation method for time fractional Black-Scholes model. *International Journal of Financial Engineering*, 10(02), 2350008. <https://doi.org/10.1142/S2424786323500081>
- [21] Podlubny, I. (1998). *Fractional Differential Equations: An Introduction To Fractional Derivatives, Fractional Differential Equations, To Methods Of Their Solution And Some Of Their Applications* (Vol. 198). Elsevier.
- [22] Samko, S. G. (1993). *Fractional Integrals And Derivatives: Theory And Applications*. Gordon and Breach Science.
- [23] Zhang, H., Zhang, M., Liu, F., & Shen, M. (2024). Review of the fractional Black-Scholes equations and their solution techniques. *Fractal and Fractional*, 8(2), 101. <https://doi.org/10.3390/fractalfract8020101>

ROYA MONTAZERI

ORCID NUMBER: 0000-0002-4971-2661

DEPARTMENT OF MATHEMATICS

PAYAME NOOR UNIVERSITY (PNU)

P.O.BOX 19395-4697, TEHRAN, IRAN

Email address: `roya.mn@pnu.ac.ir`