Journal of Mathematical Modeling
Vol. 12, No. 3, 2024, pp. 405–417. Research Article

JMM

An efficient numerical method based on cubic B–splines for the time–fractional Black–Scholes European option pricing model

Hamed Payandehdoost Masouleh[†], **Mojgan Esmailzadeh**[‡*]

[†]*Department of Accounting, Bandaranzali Branch, Islamic Azad University, Bandaranzali, Iran*
[‡]*Department of Applied Mathematics, Bandaranzali Branch, Islamic Azad University, Bandaranzali, Iran*
Email(s): Hamed.Payandehdoost@iau.ac.ir, Mojgan.Esmailzadeh@iau.ac.ir

Abstract. In this study, we develop a precise and effective numerical approach to solve the time–fractional Black–Scholes equation, which is used to calculate European options. The method employs cubic B-spline collocation for spatial discretization and a finite difference method for time discretization. An analysis of the method's stability is conducted. Finally, two numerical examples are included to show the effectiveness and applicability of the suggested method.

Keywords: Cubic B-spline, time-fractional, Black-Scholes, European option pricing model.
AMS Subject Classification 2010: 34A34, 65L05.

1 Introduction

The Black-Scholes equation is a mathematical model for pricing options in financial markets. It was developed by Fischer Black and Myron Scholes in 1973 and is based on the assumption that the price of the underlying asset follows a geometric Brownian motion with constant volatility and a risk-free rate [10].

However, some empirical studies have shown that real market data does not always follow the assumptions of the Black-Scholes model. For example, the volatility and a risk-free rate may vary over time, and the asset returns may exhibit long-range dependence or persistence [5]. To overcome these limitations, some researchers have proposed using fractional derivatives to model the dynamics of the asset price. Fractional derivatives are generalizations of ordinary derivatives that can capture the memory and non-local effects of complex systems [9]. The fractional Black-Scholes equations can capture the non-Markovian and non-Gaussian features of asset price dynamics and provide more realistic and

*Corresponding author
Received: 22 January 2024 / Revised: 12 March 2024 / Accepted: 13 March 2024
DOI: 10.22124/jmm.2024.26551.2341

flexible models for option pricing. However, they also introduce some challenges, such as the existence and uniqueness of solutions, stability and convergence of the numerical schemes, the calibration of the fractional parameters, etc [5]. Therefore, extensive research is needed to explore the properties and applications of these equations in financial mathematics.

There are various numerical methods for solving the Black-Scholes equation with fractional derivatives, which is a generalization of the classical Black-Scholes model for option pricing that can capture the non-Markovian and non-Gaussian features of the asset price dynamics. Some of the main types of numerical methods are:

Transform methods: These methods use analytical techniques such as Laplace transform, Fourier transform, or Mellin transform to transform the fractional Black-Scholes equation into an ordinary differential equation or an algebraic equation, which can be solved more easily. Then, the inverse transform is applied to obtain the solution of the original equation. For example, Ampun and Sawangtong [3] used the generalized Laplace homotopy perturbation method to solve the time-fractional Black-Scholes equation, and An et al. [3] used Fourier transform to solve the space-fractional Black-Scholes equation.

Series methods: These methods use analytical techniques such as the homotopy analysis method, homotopy perturbation method, or Adomian decomposition method to construct a series solution of the fractional Black-Scholes equation. The series solution can be truncated to obtain an approximate solution with a desired accuracy. For example, Saratha et al. [10] used the fractional generalized homotopy analysis method to solve the Black-Scholes equation with fractional time and space derivatives, and Guo et al. [11] used the Adomian decomposition method to solve Black-Scholes equation driven by fractional G–Brownian motion.

Finite difference methods: These methods use numerical techniques such as finite difference schemes, weighted finite difference schemes, or finite element schemes to discretize the fractional Black-Scholes equation in space and time domains, and obtain a system of linear or nonlinear equations that can be solved by iterative or direct methods. For example, Eslahchi et al. [7] used the weighted finite difference method to solve the two-dimensional fractional Black-Scholes equation, and Taghipour and Aminikhah [12] used a spectral collocation method based on fractional Pell functions for solving the time–fractional Black–Scholes option pricing model. Also, Kazmi [6] presented a second order numerical method for the time–fractional Black–Scholes European option pricing model.

Splines are piecewise polynomial functions that can approximate smooth curves or surfaces with high accuracy and flexibility. They have many applications in computer graphics, computer-aided design, data interpolation, and numerical analysis. Splines can provide high-order approximations of the unknown solution and its fractional derivatives and can handle singularities, discontinuities, and boundary layers more effectively than other methods. Also, splines can adapt to the shape and smoothness of the unknown solution and can be easily modified to incorporate different boundary conditions, initial conditions, or source terms. Furthermore, splines can reduce the computational cost and storage requirements of solving fractional differential equations, as they can use sparse matrices, fast algorithms, and local refinements [4, 8].

We consider the following time-fractional Black-Scholes equation [6]

$$ {}_{C}D^{\alpha}_{0,t}u = \frac{1}{2}\sigma^2\frac{\partial^2 u}{\partial x^2} + \left(r - \frac{1}{2}\sigma^2\right)\frac{\partial u}{\partial x} - ru + f(x,t), \quad (x,t) \in \Omega, \tag{1} $$

with boundary conditions

$$u(-a,t)=p(t),\quad u(a,t)=q(t),\quad t\in[0,T], \tag{2}$$

and initial condition

$$u(x,0)=h(x),\quad x\in(-a,a), \tag{3}$$

where $\Omega=(-a,a)\times(0,T]$ and

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

is the Caputo fractional derivative.

For European call options we put $p(t)=0$, $q(t)=K(e^X-e^{-rt})$ and $h(x)=\max(0,K(e^x-1))$. Also, for European put option we have $p(t)=Ke^{-rt}$, $q(t)=0$ and $h(x)=\max(0,K(1-e^x))$.

The rest of this article is as follows: In the Section 2, a new numerical method based on B-spline foundations is presented for the numerical solution of problem (1)–(3). In Section 3, the stability and convergence of the presented method are investigated. In the Section 4, numerical experiments confirm the effectiveness of the presented method. At the end, some conclusions are expressed.

2 Numerical method

In this section we assume that $t_k=k\Delta t, k=0,1,\ldots,N$ be a partition of interval $[0,T]$ where $\Delta t=\frac{T}{N}$. Also, we assume that $0\le\theta\le 1$, and $t_{k+\theta}=t_k+\theta\Delta t$.

Theorem 1 ([1]). *Assume that $k\ge 1$, $u(x,t)\in C^4[-a,a]\times C^2[0,\infty]$, then*

$$\begin{aligned}{}_CD^{\alpha}_{0,t}u(x,t_{k+\theta})=&\frac{1}{(\Delta t)^{\alpha}\Gamma(2-\alpha)}\sum_{j=0}^{k-1}\left(u(x,t_{j+1})-u(x,t_j)\right)d_{k,j}\\&+\frac{\theta^{1-\alpha}}{(\Delta t)^{\alpha}\Gamma(2-\alpha)}\left(u(x,t_{k+1})-u(x,t_k)\right)+\varepsilon_k,\end{aligned} \tag{5}$$

where $d_{k,j}=(k-j+\theta)^{1-\alpha}-(k-j+\theta-1)^{1-\alpha}$ and $\varepsilon_k\le\frac{2}{\Gamma(3-\alpha)}\max\limits_{0\le s\le t_{k+1}}|u_{tt}(x,s)|\,(\Delta t)^{2-\alpha}$.

Lemma 1. *Suppose g is a sufficiently smooth function, then*

$$g(t_{n+\theta})=(1-\theta)g(t_n)+\theta g(t_{n+1})-(1-\theta)\theta(\Delta t)^2g''(t_{n+\theta})-\theta\left(2\theta^2-3\theta+1\right)(\Delta t)^3g'''(t_{n+\theta})+\cdots.$$

Proof. We know that $t_n-t_{n+\theta}=-\theta\Delta t$ and $t_{n+1}-t_{n+\theta}=(1-\theta)\Delta t$. Then using Taylor expansion we have

$$g(t_n)=g(t_{n+\theta})-\theta\Delta tg'(t_{n+\theta})+(\theta\Delta t)^2g''(t_{n+\theta})-(\theta\Delta t)^3g'''(t_{n+\theta})+\cdots, \tag{6}$$

$$g(t_{n+1})=g(t_{n+\theta})+(1-\theta)\Delta tg'(t_{n+\theta})+((1-\theta)\Delta t)^2g''(t_{n+\theta})+((1-\theta)\Delta t)^3g'''(t_{n+\theta})+\cdots. \tag{7}$$

By multiplying (6) in $(1-\theta)$ and (7) in θ and obtaining the sum of the results will give the desired result. □

Theorem 2. *Assume that* $k \geq 1$, $u(x,t) \in C^4[-a,a] \times C^2[0,\infty]$, *and* $u(x,t)$ *satisfy* (1), *then*

$$(\lambda + \theta r)\, u(x,t_{k+1}) - \theta\gamma u_x(x,t_{k+1}) - \theta\beta u_{xx}(x,t_{k+1}) = r^k - \varepsilon_k, \tag{8}$$

where $\beta = \frac{1}{2}\sigma^2$, $\gamma = r - \beta, \eta = \frac{1}{(\Delta t)^\alpha \Gamma(2-\alpha)}$, $\lambda = \eta\theta^{1-\alpha}$, *and*

$$\begin{aligned} r^k = {} & (1-\theta)\left(\beta u_{xx}(x,t_k) + \gamma u_x(x,t_k) - r u(x,t_k)\right) + \lambda u(x,t_k) \\ & - \eta \sum_{j=0}^{k-1} \left(u(x,t_{j+1}) - u(x,t_j)\right) d_{k,j} + f(x,t_{k+\theta}) \end{aligned} \tag{9}$$

Proof. Using Lemma 1, we get

$$\begin{aligned} {}_C D^\alpha_{0,t} u(x,t_{k+\theta}) = {} & (1-\theta)\, {}_C D^\alpha_{0,t} u(x,t_k) + \theta\, {}_C D^\alpha_{0,t} u(x,t_{k+1}) \\ & - (1-\theta)\theta(\Delta t)^2\, {}_C D^{\alpha+2}_{0,t} u(x,t_{k+\theta}) + O\left((\Delta t)^3\right) \end{aligned} \tag{10}$$

By applying (1), we obtain

$$\begin{aligned} {}_C D^\alpha_{0,t} u(x,t_{k+\theta}) = {} & (1-\theta)\left(\beta \frac{\partial^2 u(x,t_k)}{\partial x^2} + \gamma \frac{\partial u(x,t_k)}{\partial x} - r u(x,t_k)\right) \\ & + \theta\left(\beta \frac{\partial^2 u(x,t_{k+1})}{\partial x^2} + \gamma \frac{\partial u(x,t_{k+1})}{\partial x} - r u(x,t_{k+1})\right) + f(x,t_{k+\theta}) \\ & - (1-\theta)\theta(\Delta t)^2\, {}_C D^{\alpha+2}_{0,t} u(x,t_{k+\theta}) + O\left((\Delta t)^3\right). \end{aligned} \tag{11}$$

Now, by substituting (5) in (11) the result will be achieved. □

By eliminating the error term ε_k in (8), we get

$$\delta_1 U^{k+1} + \delta_2 U_x^{k+1} + \delta_3 U_{xx}^{k+1} = R^k, \tag{12}$$

where $\delta_1 = \lambda + \theta r$, $\delta_2 = -\theta\gamma$, $\delta_3 = -\theta\beta$, $U^{k+1} = U(x,t_{k+1})$, $U_x^{k+1} = U_x(x,t_{k+1})$, $U_{xx}^{k+1} = U_{xx}(x,t_{k+1})$, and

$$R^k = (1-\theta)\left(\beta U_{xx}^k + \gamma U_x^k - r U^k\right) + \lambda U^k - \eta \sum_{j=0}^{k-1} \left(U^{j+1} - U^j\right) d_{k,j} + f(x,t_{k+\theta}). \tag{13}$$

Let $x_i = -a + i\Delta x, i = 0,1,\ldots,M$, $\Delta x = \frac{2a}{M}$. By putting $x = x_i$ in (12), we get

$$\delta_1 U_i^{k+1} + \delta_2 \left(U_x\right)_i^{k+1} + \delta_3 \left(U_{xx}\right)_i^{k+1} = R_i^k, \tag{14}$$

where $U_i^{k+1} = U(x_i,t_{k+1})$, $\left(U_x\right)_i^{k+1} = U_x(x_i,t_{k+1})$, $\left(U_{xx}\right)_i^{k+1} = U_{xx}(x_i,t_{k+1})$, and for $k = 0$ from equation (9), we get

$$R_i^k = (1-\theta)\left(\beta h''(x_i) + \gamma h'(x_i) - r h(x_i)\right) + \lambda h(x_i) + f(x_i,t_{k+\theta}), \tag{15}$$

and for $k>0$

$$R_i^k=(1-\theta)\left(\beta\,(U_{xx})_i^k+\gamma(U_x)_i^k-rU_i^k\right)+\lambda U_i^k \\ -\eta\sum_{j=1}^{k-1}\left(U_i^{j+1}-U_i^j\right)d_{k,j}-\eta\left(U_i^1-h(x_i)\right)d_{k,0}+f(x_i,t_{k+\theta}). \tag{16}$$

For approximation U_i^k, $(U_x)_i^k$, and $(U_{xx})_i^k$ we use cubic B-spline approximation as follows

$$U^{k+1}(x_i)=\sum_{k=-1}^{M+1}c_k^{k+1}B_k(x_i),\quad U_x^{k+1}(x_i)=\sum_{k=-1}^{M+1}c_k^{k+1}B'_k(x_i),\quad U_{xx}^{k+1}(x_i)=\sum_{k=-1}^{M+1}c_k^{k+1}B''_k(x_i), \tag{17}$$

where $B_k(x)=B_{j+2}(x)$ and $B_j(x)$, $j=-3,-2,\ldots,M-1$, is the jth cubic B-spline function with bellow define [2]

$$B_j(x)=\begin{cases}\dfrac{(x-x_j)^3}{(x_{j+1}-x_j)(x_{j+2}-x_j)(x_{j+3}-x_j)}, & x\in[x_j,x_{j+1}),\\ \dfrac{(x-x_j)^2(x_{j+2}-x)}{(x_{j+2}-x_j)(x_{j+2}-x_{j+1})(x_{j+3}-x_j)}+\dfrac{(x-x_j)(x_{j+3}-x)(x-x_{j+1})}{(x_{j+3}-x_j)(x_{j+3}-x_{j+1})(x_{j+2}-x_{j+1})}\\ \quad+\dfrac{(x-x_{j+1})^2(x_{j+4}-x)}{(x_{j+2}-x_{j+1})(x_{j+3}-x_{j+1})(x_{j+4}-x_{j+1})}, & x\in[x_{j+1},x_{j+2})\\ \dfrac{(x-x_j)(x_{j+3}-x)^2}{(x_{j+3}-x_j)(x_{j+3}-x_{j+1})(x_{j+3}-x_{j+2})}+\dfrac{(x-x_{j+1})(x_{j+3}-x)(x_{j+4}-x)}{(x_{j+3}-x_{j+1})(x_{j+3}-x_{j+2})(x_{j+4}-x_{j+1})}\\ \quad+\dfrac{(x-x_{j+2})(x_{j+4}-x)^2}{(x_{j+4}-x_{j+1})(x_{j+4}-x_{j+2})(x_{j+3}-x_{j+2})}, & x\in[x_{j+2},x_{j+3})\\ \dfrac{(x_{j+4}-x)^3}{(x_{j+4}-x_{j+1})(x_{j+4}-x_{j+2})(x_{j+4}-x_{j+3})}, & x\in[x_{j+3},x_{j+4})\end{cases} \tag{18}$$

where $\{B_{-1},B_0,\ldots,B_{M+1}\}$ is a basis for the space of cubic splines on $\Delta=\{x_0,x_1,\ldots,x_M\}$ with extra knots $x_{-3}=x_{-2}=x_{-1}=x_0$ and $x_N=x_{N+1}=x_{N+2}=x_{N+3}$.

Remark 1. *Assume that* $\mathbf{B}=[B_{i,j}]$, $B_{i,j}=B_i(x_j)$, $\mathbf{B}'=[B'_{i,j}]$, $B'_{i,j}=B'_i(x_j)$, $\mathbf{B}''=[B''_{i,j}]$, $B''_{i,j}=B''_i(x_j)$, *for* $i=-1,2,\ldots,M+1$, $j=0,1,\ldots,M$. *Regarding to the definition of* b_k*s, we gain*

$$\mathbf{B}=\frac{1}{12}\begin{pmatrix}12&&&&&&&&&\\ &3&&&&&&&&\\ &7&2&&&&&&&\\ &2&8&2&&&&&&\\ &&\ddots&\ddots&\ddots&&&&&\\ &&&&&2&8&2&&\\ &&&&&&2&7&&\\ &&&&&&&3&&\\ &&&&&&&&12&\end{pmatrix},$$

$$\mathbf{B}' = \frac{1}{4\Delta x}\begin{pmatrix} -12 & & & & & & & & & \\ 13 & -3 & & & & & & & & \\ & 1 & -2 & & & & & & & \\ & 2 & 0 & -2 & & & & & & \\ & & \ddots & \ddots & \ddots & & & & & \\ & & & & & 2 & 0 & -2 & & \\ & & & & & & 2 & -1 & & \\ & & & & & & & 3 & -12 & \\ & & & & & & & & 12 & \end{pmatrix},$$

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

Now let $C_M^{k+1} = \left(c_{-1}^{k+1}, c_0^{k+1}, \ldots, c_{M+1}^{k+1}\right)^T$, $\mathbf{B} = (\mathbf{B}_0, \mathbf{B}_1, \ldots, \mathbf{B}_M)$, $\mathbf{B}' = (\mathbf{B}'_0, \mathbf{B}'_1, \ldots, \mathbf{B}'_M)$, and $\mathbf{B}'' = (\mathbf{B}''_0, \mathbf{B}''_1, \ldots, \mathbf{B}''_M)$, then from (17), we have

$$U^{k+1}(x_i) = \left(C_M^{k+1}\right)^T \mathbf{B}_i, \quad U_x^{k+1}(x_i) = \left(C_M^{k+1}\right)^T \mathbf{B}'_i, \quad U_{xx}^{k+1}(x_i) = \left(C_M^{k+1}\right)^T \mathbf{B}''_i. \tag{19}$$

Regarding to (14) and (19), we get

$$\left(C_M^{k+1}\right)^T \Phi_i = R_i^k, \quad i = 0, 1, \ldots, M, \tag{20}$$

where $\Phi_i = \delta_1 \mathbf{B}_i + \delta_2 \mathbf{B}'_i + \delta_3 \mathbf{B}''_i$. Also, from (2), we have

$$\begin{aligned} \left(C_M^{k+1}\right)^T \mathbf{B}_0 &= p(t_{k+1}), \\ \left(C_M^{k+1}\right)^T \mathbf{B}_M &= q(t_{k+1}), \end{aligned} \tag{21}$$

From (20) and (21), we have

$$\Lambda^T C_M^{k+1} = \mathbf{R}^k, \tag{22}$$

where $\mathbf{R}^k = \left(p(t_{k+1}), R_0^k, R_1^k, \ldots, R_M^k, q(t_{k+1})\right)^T$ and $\Lambda = (\mathbf{B}_0, \Phi_0, \Phi_1, \ldots, \Phi_M, \mathbf{B}_M)$.

3 Error analysis

In this section, we first examine the stability and then the convergence of the presented numerical method.

3.1 Stability analysis

In this subsection, we examine the stability of the presented numerical method. From (22), we have

$$\left\| C_M^{k+1} \right\|_\infty \leq \left\| \Lambda^{-T} \right\|_\infty \left\| \mathbf{R}^k \right\|_\infty. \tag{23}$$

On other hand, from (15) and (16) and considering $\max_{0\le j\le k-1}\left|d_{k,j}\right|\le 1$, we get

$$\left\|\mathbf{R}^0\right\|_\infty \le R_1, \tag{24}$$

$$\left\|\mathbf{R}^1\right\|_\infty \le [P+\eta]\left\|\left(C_M^1\right)^T\right\|_\infty + M_2, \tag{25}$$

$$\left\|\mathbf{R}^k\right\|_\infty \le P\left\|\left(C_M^k\right)^T\right\|_\infty + \eta\sum_{j=1}^{k-1}\left(\left\|\left(C_M^{j+1}\right)^T\right\|_\infty + \left\|\left(C_M^j\right)^T\right\|_\infty\right) + \eta\left\|\left(C_M^1\right)^T\right\|_\infty + M_2, \quad k>1, \tag{26}$$

where

$$R_1 = \max_{0\le i\le M}\left\{(1-\theta)\left(\beta\left|h''(x_i)\right| + \gamma\left|h'(x_i)\right| + r\left|h(x_i)\right|\right) + \lambda\left|h(x_i)\right| + \left|f(x_i,t_{k+\theta})\right|\right\},$$

$$M_2 = \max_{0\le i\le M}\left\{\eta\left|h(x_i)\right| + \left|f(x_i,t_{k+\theta})\right|\right\},$$

$$P = (1-\theta)\left(\frac{\beta}{9(\Delta x)^2} + \frac{3\gamma}{\Delta x} + r\right) + \lambda.$$

Now, equations (23), (24), (25) and (26) results in

$$\left\|C_M^1\right\|_\infty \le R_1\left\|\Lambda^{-T}\right\|_\infty, \tag{27}$$

$$\left\|C_M^2\right\|_\infty \le \left\|\Lambda^{-T}\right\|_\infty\left([P+\eta]\left\|\left(C_M^1\right)^T\right\|_\infty + M_2\right), \tag{28}$$

$$\left\|C_M^{k+1}\right\|_\infty \le \left\|\Lambda^{-T}\right\|_\infty\left(P\left\|\left(C_M^k\right)^T\right\|_\infty + \eta\sum_{j=1}^{k-1}\left(\left\|\left(C_M^{j+1}\right)^T\right\|_\infty + \left\|\left(C_M^j\right)^T\right\|_\infty\right) + \eta\left\|\left(C_M^1\right)^T\right\|_\infty + M_2\right), \quad k>1. \tag{29}$$

According to Archimedean property in $\mathbb{R}$ there exists a real number K_1, such that $M_2 \le K_1\left\|\left(C_M^1\right)^T\right\|_\infty$, then using (27) and (28), we get $\left\|C_M^2\right\|_\infty \le R_2\left(\left\|\Lambda^{-T}\right\|_\infty\right)^2$, where $R_2 = M_1(P+\eta+K_1)$. Let $\left\|C_M^l\right\|_\infty \le R_l\left(\left\|\Lambda^{-T}\right\|_\infty\right)^l$ for $l=1,2,3,\ldots,k$. Now using (29), we have

$$\left\|C_M^{k+1}\right\|_\infty \le \left(\left\|\Lambda^{-T}\right\|_\infty\right)^2\left((P+\eta)R_k\left(\left\|\Lambda^{-T}\right\|_\infty\right)^{k-1} + (2\eta+K_1)M_1 + \eta\sum_{j=2}^{k-1}R_j\left(\left\|\Lambda^{-T}\right\|_\infty\right)^{j-1}\right). \tag{30}$$

According to Archimedean property in $\mathbb{R}$ there exists a real number K_3, such that $(2\eta+K_1)M_1 \le K_3\left\|\Lambda^{-T}\right\|_\infty$, then it follows from relation (30) that

$$\left\|C_M^{k+1}\right\|_\infty \le \left(\left\|\Lambda^{-T}\right\|_\infty\right)^3\left((P+\eta)R_k\left(\left\|\Lambda^{-T}\right\|_\infty\right)^{k-2} + K_3 + \eta R_2 + \eta\sum_{j=3}^{k-1}R_j\left(\left\|\Lambda^{-T}\right\|_\infty\right)^{j-2}\right).$$

By repeating this process, we can conclude that there exists a real number R_{k+1}, such that

$$\left\|C_M^{k+1}\right\|_\infty \le R_{k+1}\left(\left\|\Lambda^{-T}\right\|_\infty\right)^{k+1}. \tag{31}$$

Now we discuss the eigenvalues of matrix Λ^{-T}. Since the eigenvalues of Λ^{-T} are the inverse of the eigenvalues of Λ^T, we will examine the limits of the eigenvalues of matrix Λ^T and show that they tend to infinity when M tends

to infinity. We know that

$$\Lambda^T = \begin{pmatrix} 1 & & & & & & & \\ a & b & c & & & & & \\ & d & e & f & & & & \\ & & d & e & f & & & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & d & e & f & \\ & & & & & g & h & k \\ & & & & & & & 1 \end{pmatrix},$$

where

$$a = \frac{3\delta_1}{12} + \frac{-3\delta_2}{4\Delta x} + \frac{3\delta_3}{2(\Delta x)^2}, \quad b = \frac{7\delta_1}{12} + \frac{\delta_2}{4\Delta x} + \frac{-5\delta_3}{2(\Delta x)^2}, \quad c = \frac{2\delta_1}{12} + \frac{2\delta_2}{4\Delta x} + \frac{-2\delta_3}{2(\Delta x)^2},$$
$$d = \frac{2\delta_1}{12} + \frac{-2\delta_2}{4\Delta x} + \frac{2\delta_3}{2(\Delta x)^2}, \quad e = \frac{8\delta_1}{12} + \frac{-4\delta_3}{2(\Delta x)^2}, \quad f = \frac{2\delta_1}{12} + \frac{2\delta_2}{4\Delta x} + \frac{2\delta_3}{2(\Delta x)^2},$$
$$g = \frac{2\delta_1}{12} + \frac{-2\delta_2}{4\Delta x} + \frac{2\delta_3}{2(\Delta x)^2}, \quad h = \frac{7\delta_1}{12} + \frac{-\delta_2}{4\Delta x} + \frac{-5\delta_3}{2(\Delta x)^2}, \quad k = \frac{3\delta_1}{12} + \frac{3\delta_2}{4\Delta x} + \frac{3\delta_3}{2(\Delta x)^2}.$$

Using the famous Gershgorin circles theorem, we have

$$\begin{aligned} & z = 1, \\ & \frac{2\delta_1(\Delta x)^2 + 15\delta_2\Delta x + 6\delta_3}{12(\Delta x)^2} \le z \le \frac{12\delta_1(\Delta x)^2 - 15\delta_2\Delta x - 54\delta_3}{12(\Delta x)^2}, \\ & \frac{4\delta_1(\Delta x)^2 + 12\delta_2\Delta x}{12(\Delta x)^2} \le z \le \frac{12\delta_1(\Delta x)^2 - 12\Delta x\delta_2 - 48\delta_3}{12(\Delta x)^2}, \\ & \frac{2\delta_1(\Delta x)^2 + 12\delta_2\Delta x}{12(\Delta x)^2} \le z \le \frac{12\delta_1(\Delta x)^2 - 18\Delta x\delta_2 - 60\delta_3}{12(\Delta x)^2}, \end{aligned} \tag{32}$$

where z is the eigenvalue of matrix Λ^T. We assume that $r > \frac{1}{2}\sigma^2$. Since $\delta_1 = \frac{\theta^{1-\alpha}}{(\Delta t)^\alpha \Gamma(2-\alpha)} + \theta r > 0$, $\delta_2 = -\theta(r - \frac{1}{2}\sigma^2) < 0$, $\delta_3 = -\frac{1}{2}\theta\sigma^2 < 0$, from (32) we have

$$z \in \{1\} \cup \left\{ x \,\middle|\, \frac{2\delta_1(\Delta x)^2 + 15\delta_2\Delta x + 6\delta_3}{12(\Delta x)^2} \le x \le \frac{12\delta_1(\Delta x)^2 - 18\Delta x\delta_2 - 60\delta_3}{12(\Delta x)^2} \right\}. \tag{33}$$

Therefore, Eq. (33) implies that when Δx tends to zeros then $\frac{1}{|z|} \to 0$.

3.2 Convergence analysis

Let $E^k = u(x,t_{k+1}) - U(x,t_{k+1})$, $E^k_x = u_x(x,t_{k+1}) - U_x(x,t_{k+1})$, $E^k_{xx} = u_{xx}(x,t_{k+1}) - U_{xx}(x,t_{k+1})$, denote the errors at time t_{k+1}. By subtracting (8) from (12), we have

$$\delta_1 E^{k+1} + \delta_2 E^{k+1}_x + \delta_3 E^{k+1}_{xx} = (1-\theta)\left(\beta E^k_{xx} + \gamma E^k_x - rE^k\right) + \lambda E^k - \eta \sum_{j=0}^{k-1} \left(E^{j+1} - E^j\right) d_{k,j} - \varepsilon_k. \tag{34}$$

Now we put $E^{k+1}(x_i) = \left(D_M^{k+1}\right)^T \mathbf{B}_i$, $E_x^{k+1}(x_i) = \left(D_M^{k+1}\right)^T \mathbf{B}'_i$, $E_{xx}^{k+1}(x_i) = \left(D_M^{k+1}\right)^T \mathbf{B}''_i$, and use the numerical method on (34). First of all, according to the initial and boundary conditions we have $\left(D_M^{k+1}\right)^T \mathbf{B}_0 = 0$, $\left(D_M^{k+1}\right)^T \mathbf{B}_M = 0$ and $0 = E^0(x_i) = \left(D_M^0\right)^T \mathbf{B}_i$. Also, by putting $k = 0$ in (34), we get

$$\left(D_M^1\right)^T \Phi_i = -\varepsilon_0. \tag{35}$$

In addition, for $k > 0$ from (34) we have

$$\left(D_M^{k+1}\right)^T \Phi_i = y_i^k, \tag{36}$$

where

$$y_i^k = \left(D_M^k\right)^T \left((1-\theta)\left(\beta \mathbf{B}''_i + \gamma \mathbf{B}'_i - r\mathbf{B}_i\right) + \lambda \mathbf{B}_i\right) - \eta \sum_{j=0}^{k-1} \left(\left(D_M^{j+1}\right)^T - \left(D_M^j\right)^T\right) \mathbf{B}_i d_{k,j} - \varepsilon_k. \tag{37}$$

Therefore

$$\Lambda^T D_M^{k+1} = Y^k, \quad k \geq 0, \tag{38}$$

where $Y^0 = -(0, \varepsilon_0, \varepsilon_0, \ldots, \varepsilon_0, 0)^T$ and $Y^k = \left(0, y_0^k, y_1^k, \ldots, y_M^k, 0\right)^T$ for $k > 0$. Using (37) and the definition of Y^0, it results

$$\begin{aligned} &\left\|Y^0\right\|_\infty \leq \varepsilon_0, \\ &\left\|Y^k\right\|_\infty \leq (P+\eta) \left\|\left(D_M^k\right)^T\right\|_\infty + \eta \sum_{j=1}^{k-1} \left\|\left(D_M^j\right)^T\right\|_\infty + \varepsilon_k, \quad k > 0 \end{aligned} \tag{39}$$

From (38) and (39), we have

$$\left\|D_M^1\right\|_\infty \leq \left\|\Lambda^{-T}\right\|_\infty \left\|Y^0\right\|_\infty \leq \varepsilon_0 \left\|\Lambda^{-T}\right\|_\infty. \tag{40}$$

Since $\lim_{\Delta t \to 0} \varepsilon_k = 0$ for all $k \geq 0$, (40) results that $\left\|D_M^1\right\|_\infty \to 0$ as Δt tends to 0. Also, from (38) and (39), we get

$$\left\|D_M^2\right\|_\infty \leq \left\|\Lambda^{-T}\right\|_\infty \left((P+\eta)\,\varepsilon_0 \left\|\Lambda^{-T}\right\|_\infty + \varepsilon_1\right). \tag{41}$$

Given that $\lim_{\Delta t \to 0} \varepsilon_k = 0$ for all $k \geq 0$, from (41) we have $\lim_{\Delta t \to 0} \left\|D_M^2\right\|_\infty = 0$. Let $\lim_{\Delta t \to 0} \left\|D_M^l\right\|_\infty = 0$ for $l = 0, 1, \ldots, k$. From (38) and (39), we have

$$\left\|D_M^{k+1}\right\|_\infty \leq \left\|\Lambda^{-T}\right\|_\infty \left\|Y^k\right\|_\infty \leq \left\|\Lambda^{-T}\right\|_\infty \left((P+\eta) \left\|\left(D_M^k\right)^T\right\|_\infty + \eta \sum_{j=1}^{k-1} \left\|\left(D_M^j\right)^T\right\|_\infty + \varepsilon_k\right). \tag{42}$$

Now, since $\lim_{\Delta t \to 0} \varepsilon_k = 0$ and $\lim_{\Delta t \to 0} \left\|D_M^l\right\|_\infty = 0$, $l = 0, 1, \ldots, k$, from (42), we get $\lim_{\Delta t \to 0} \left\|D_M^{k+1}\right\|_\infty = 0$.

4 Numerical experiments

This section contains two numerical illustrations. We apply the time-fractional Black-Scholes equation with smooth initial conditions to the first example. The second example deals with the valuation of a European put option with a non-smooth payoff function based on the time-fractional Black-Scholes equation of order α.

All computations are performed by MATLAB R2019a software running on a Sony VAIO Laptop (Intel Core i5-2410 M Processor 2.30 GHz with Turbo Boost up to 2.90 GHz, 8 GB of RAM, 64-bit).

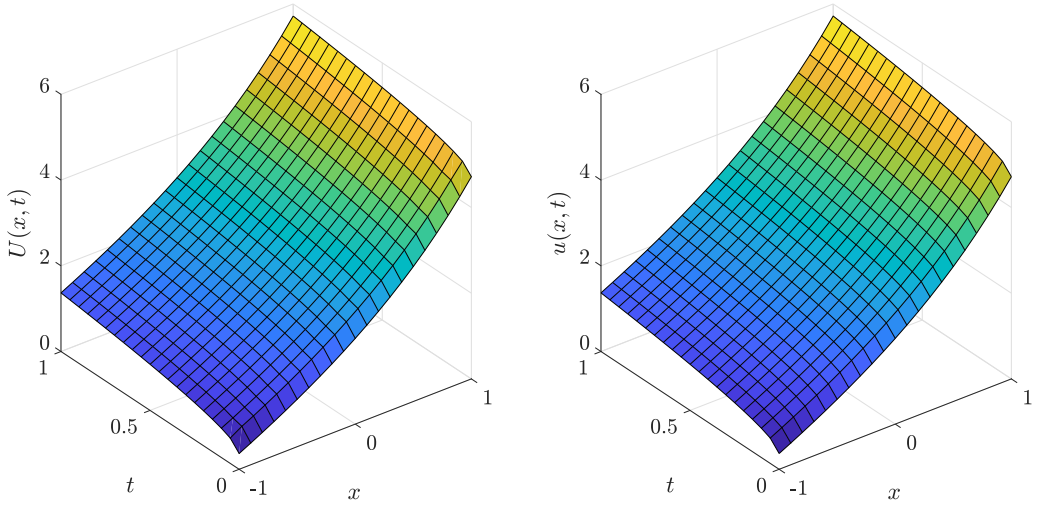

Figure 1: Surface plots with $M = N = 20$, $\alpha = 0.5$ for Example 1.

Example 1. In this example, we consider the following time-fractional partial differential equation:

$$
\begin{aligned}
&{}_{C}D^{\alpha}_{0,t}u = \frac{1}{2}\sigma^2\frac{\partial^2 u}{\partial x^2} + \left(r - \frac{1}{2}\sigma^2\right)\frac{\partial u}{\partial x} - ru + f(x,t), \quad (x,t) \in (-1,1)\times(0,1],\\
&u(-1,t) = t^{\alpha} + e^{-1}, \quad u(1,t) = t^{\alpha} + e + 2, \quad t \in (0,1],\\
&u(x,0) = e^{x} + x + 1, \quad x \in (-1,1),
\end{aligned}
$$

where $\theta = 0.5$, $\sigma = 0.1$, and $r = 0.06$. The reaction term $f(x,t)$ is chosen such that the exact solution is $u(x,t) = t^{\alpha} + e^{x} + x + 1$. The surface plots of the analytical solution and the numerical solution obtained by using the proposed technique for $\alpha = 0.5$ are depicted in Figure 1.

We measure the accuracy of the numerical solution at time $t = t_n$ by computing the discrete L_2 norm of the error

$$
e^{M,N} = \sqrt{\sum_{k=1}^{M-1} \Delta x \left| u(x_k,t_n) - U_k^n \right|^2},
$$

where $U^{M,N}$ denotes the numerical solution computed using the spatial discretization parameter M and the temporal discretization parameter N. The numerical convergence rate is then computed as

$$
R^{M,N} = \log_2\left(\frac{e^{M,N}}{e^{2M,2N}}\right).
$$

The discrete L_2 errors and the convergence rates for the numerical solution at the final time $t = 1$ for $\alpha = 0.5$ are given in Table 1.

Table 1: Error $e^{M,N}$ and convergence rate $R^{M,N}$ at $t = 1$ with $M = 50$ for Example 1.

N	$e^{M,N}$	$R^{M,N}$
4	$2.994995e-03$	–
8	$8.512802e-04$	1.8148
16	$2.262609e-04$	1.9117
32	$6.592457e-05$	1.7791
64	$2.024992e-05$	1.7029
128	$6.539345e-06$	1.6307
256	$2.427722e-06$	1.4295

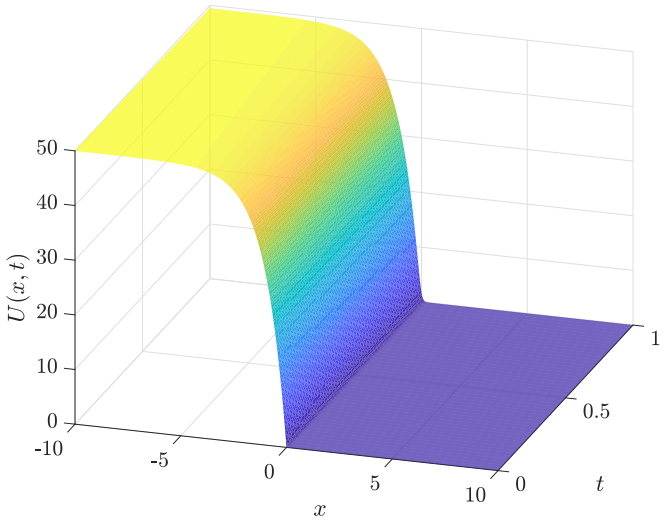

Figure 2: Surface plots with $N = 50$ for Example 2.

Example 2. Here, we consider the following problem

$$
\begin{aligned}
&{}_cD^{\alpha}_{0,t}u = \frac{1}{2}\sigma^2\frac{\partial^2 u}{\partial x^2} + \left(r - \frac{1}{2}\sigma^2\right)\frac{\partial u}{\partial x} - ru, \quad (x,t) \in (-X,X)\times(0,T],\\
&u(-X,t) = Ke^{-rt}, \quad u(X,t) = 0, \quad t \in (0,T],\\
&u(x,0) = \max K - Ke^{x}, 0, \quad x \in (-X,X),
\end{aligned}
$$

where $\theta = \alpha = 0.5$, $\sigma = 0.1$, $r = 0.01$, $K = 50$, and $M = 1000$. The surface plots of the numerical solution for $X = 10$ and $N = 50$ obtained by using the proposed method is depicted in Figure 2.

As the exact solution of this problem is not available, we use the error by computing the discrete $L^N_\infty = \left\|U^{M,2N} - U^{M,N}\right\|$ norm, and the spatial convergence rate is then estimated as

$$R^N_\infty = \log_2\left(\frac{L^N_\infty}{L^{2N}_\infty}\right).$$

Table 2: Error L_∞^N and convergence rate R_∞^N at $t = 1$ with $M = 50$ for Example 2.

N	L_∞^N	R_∞^N
16	$1.272127e-02$	–
16	$6.153785e-03$	1.0477
32	$2.984136e-03$	1.0442
64	$1.491086e-03$	1.0009
128	$7.098834e-04$	1.0707
256	$3.528315e-04$	1.0086
512	$1.766154e-04$	0.9984
1024	$8.838717e-05$	0.9987

Table 2 list the temporal error estimates and convergence rates of the numerical method for various values of N.

5 Conclusion

In this paper, we introduced a new numerical technique for finding the solution of the time–fractional Black–Scholes equation, which is a mathematical model that describes the behavior of European options in financial markets. The technique used a combination of cubic B-spline collocation and finite difference methods to discretize the equation in both space and time dimensions. We established the stability and convergence properties of the technique both theoretically and numerically. We also demonstrated the effectiveness and accuracy of the technique by applying it on two different examples.

References

[1] J. Alavi, H. Aminikhah, *An efficient parametric finite difference and orthogonal spline approximation for solving the weakly singular nonlinear time-fractional partial integro-differential equation*, Comput. Appl. Math. **42** (2023) 350.

[2] H. Aminikhah, J. Alavi, *An efficient B-spline difference method for solving system of nonlinear parabolic PDEs*, SeMA Journal **75** (2018) 335–348.

[3] S. Ampun, P. Sawangtong, *The Approximate Analytic Solution of the Time-Fractional Black-Scholes Equation with a European Option Based on the Katugampola Fractional Derivative*, Mathematics **9** (2021) 214.

[4] P. Assari, S. Cuomo, *The numerical solution of fractional differential equations using the Volterra integral equation method based on thin plate splines*, Eng. Comput. **35** (2019) 1391–1408.

[5] G. Changhong, F. Shaomei, H. Yong, *Derivation and Application of Some Fractional BlackScholes Equations Driven by Fractional G-Brownian Motion*, Comput. Econ. **61** (2023) 1681–1705.

[6] K. Kazmi, *A second order numerical method for the time-fractional BlackScholes European option pricing model*, J. Comput. Appl. Math. **418** (2023) 114647.

[7] M.N. Koleva, L.G. Vulkov, *Numerical solution of time-fractional BlackScholes equation*, Comp. Appl. Math. **36** (2017) 1699–1715.

[8] F. Mirzaee, K. Sayevand, S. Rezaei, N. Samadyar, *Finite Difference and Spline Approximation for Solving Fractional Stochastic Advection-Diffusion Equation*, Iran J. Sci. Technol. Trans. Sci. **45** (2021) 607–617.

[9] D. Prathumwan, K. Trachoo, *On the solution of two-dimensional fractional Black-Scholes equation for European put option*, Adv. Differ. Equ. **2020** (2020) 146.

[10] S.R. Saratha, G.S. Sundara Krishnan, M. Bagyalakshmi, C.P. Lim, *Solving BlackScholes equations using fractional generalized homotopy analysis method*, Comp. Appl. Math. **39** (2020) 262.

[11] M. She, L. Li, R. Tang, D.F. Li, *A novel numerical scheme for a time fractional BlackScholes equation*, J. Appl. Math. Comput. **66** (2021) 853–870.

[12] M. Taghipour, H. Aminikhah, *A spectral collocation method based on fractional Pell functions for solving timefractional BlackScholes option pricing model*, Chaos Solit. Fractals **163** (2022) 112571.