Research Article

Shuang Wu, Lot-Kei Chou, Xu Chen, and Siu-Long Lei*

A preconditioned iterative method for coupled fractional partial differential equation in European option pricing

https://doi.org/10.1515/math-2023-0169
received September 15, 2023; accepted December 7, 2023

Abstract: Recently, regime-switching option pricing based on fractional diffusion models has been used, which explains many significant empirical facts about financial markets better. There are many methods to solve the problem, but to the best of our knowledge, effective preconditioners for the second-order schemes have not been proposed. Thus, in this article, an implicit numerical scheme is developed for a regime-switching European option pricing problem under a multi-state tempered fractional model. The scheme is proven to be unconditionally stable and converges quadratically in space and linearly in time. Besides, the resulting linear system is solved using an iterative method, and a preconditioner is proposed to accelerate the rate of convergence. The preconditioner is constructed through circulant approximations to the Toeplitz blocks due to the coefficient matrix, which is is a block matrix with Toeplitz blocks. The spectral analysis of the preconditioned matrix is given, which demonstrates that the spectrum of the preconditioned matrix is clustered around 1. Numerical examples show the efficiency of the proposed method, and an empirical study is also provided.

Keywords: fully implicit finite difference method, preconditioner, regime-switching European option pricing, tempered fractional partial differential equation

MSC 2020: 65F05, 65F08, 65M06, 91G80, 35R11

1 Introduction

Since the 1970s, the classic Black-Scholes (BS) [1] European option pricing model has been proposed for calculating asset values, and is one of the most fundamental and powerful tools in financial mathematics. However, the standard BS model is recognized to have several shortcomings and is unable to account for the phenomenon of many significant empirical events in the financial markets, such as skewed return distribution, and the assumption of constant volatility generates bias. To compensate for the shortcomings of the standard BS model, some alternative models are needed.

Thus, various alternate models have been proposed. Within these models, the Merton jump-diffusion model [2] is one of the earliest alternatives, which is based on a compound Poisson jump process. The Kou jump-diffusion model [3] is based on a double exponential jump-diffusion model. The finite moment logarithmic stability (FMLS) model [4], Carr-Geman-Madan-Yor (CGMY) model [5], and Koponen-Boyarchenko-Levendorski (KoBoL) model [6] are all based on some Lévy processes. The Heston model [7] is used to account for stochastic volatility, where the volatility parameter is stochastic rather than a constant. Recently, a number

*** Corresponding author: Siu-Long Lei,** Department of Mathematics, University of Macau, Macau, China, e-mail: sllei@um.edu.mo
Shuang Wu: Department of Mathematics, University of Macau, Macau, China, e-mail: yc27936@um.edu.mo
Lot-Kei Chou: Department of Mathematics, University of Macau, Macau, China, e-mail: rickyclk16@gmail.com
Xu Chen: School of Economics, Guangdong University of Technology, Guangzhou, 510520, China, e-mail: yukchan@gdut.edu.cn

of models based on regime-switching models have been used to simulate the effects of structural changes in economic conditions or different phases of the business cycle. Hence, to capture more empirical features of the market and respond to state changes effectively, regime-switching models began to emerge. In these models, parameters such as drift and volatility parameters depend on a Markov chain and are allowed to change between regimes.

The option pricing problem based on regime-switching Lévy processes is widely discussed. In [8], the valuation problem of life-contingent lookback options is studied, in which the underlying asset price process is assumed to be an exponential regime-switching Lévy process. The regime-switching option pricing under an exponential Lévy model with coefficients modified by a Markov chain is considered, and the underlying risky asset is controlled through a regime-switching CGMY process [9]. In [10], a regime-switching intensity model for credit risk pricing is considered, where default events are specified by a Poisson process and their intensity is modeled by a Lévy process. Some applications of the option pricing problem based on regime-switching Lévy processes have been proposed in many works of literature. The pricing of some multivariate European options under the Markov-modulated Lévy processes model is studied in [11]. In [12], a problem governing the price of American options followed a geometric regime-switching Lévy process. Besides, the regime-switching model also has lower computational complexity than stochastic volatility models.

For solving the option pricing problem with regime-switching Lévy processes, a number of different numerical methods are studied in [8,12–18]. This study concentrates on resolving the financial problem using the fractional partial differential equation (FPDE)-based approach. In [16], the European option pricing problem under a regime-switching FMLS model governed by a coupled FPDE system is investigated. Two finite difference methods are shown, namely, the implicit finite difference method (IM) and implicit-explicit (IMEX) finite difference method with CGMY and KoBoL models [15]. Besides, the Fourier transform used for the European barrier option value under an FPDE based on some specific Lévy processes has been introduced creatively in [17]. In [14], the European option pricing problem under the FMLS model with stochastic volatility and involving three-dimensional FPDE system is discussed. An explicit closed form for European-style option pricing under the FMLS model based on the FPDE system is discussed in [19]. A fast penalty method is proposed for problems involving the pricing of American options whose underlying asset follows a geometric regime-switching Lévy process in [12]. A power penalty method is proposed for the American option pricing problem based on the geometric Lévy process governed by a nonlinear FPDE system considered in [20]. In [21], the multi-asset option pricing model under the multi-variate CGMY process based on a tempered FPDE is considered. In these studies, the models are essentially first-order discretized in space. To the best of our knowledge, none of them have considered the second-order discretization in space. Hence, one of the primary goals of this research is to consider a second-order implicit finite difference method based on the coupled FPDE.

The contributions of this article are as follows. First, we propose a second-order discretized scheme of space and provide the stability and convergence proof of this scheme. Second, a novel preconditioner is provided for speeding up the Krylov subspace method, and the proof of the spectrum distribution is provided to show that the spectrum of the preconditioned matrix is concentrated around 1. Third, the preconditioner matrix contains the intensity matrix Q, and since the structure of the preconditioner matrix is not easy to invert, we consider the permutation of the matrix and propose an inverse method based on the Sherman-Morrison formula and the incomplete LU (ILU) factorizations. Finally, we performed several experiments, which include an empirical case to compare the numerical results using different preconditioners and fast numerical algorithms, and our numerical results show the high efficiency of this new approach compared to several previously proposed techniques.

This article is organized as follows. In Section 2, the model of the tempered FPDE governing multi-state European option pricing is discretized by the implicit second-order scheme with analysis for stability and convergence. A fast preconditioned iterative method for the discrete second-order system and some proofs as well as the implementation of the preconditioner are given in Section 3. In Section 4, numerical experiments and a real-world experiment are given to demonstrate the efficiency of the block preconditioner. Concluding remarks are given in Section 5.

2 Second-order scheme

In this section, the coupled FPDE in regime-switching European option pricing is introduced. Then, its second-order discretized form and matrix form are proposed. The analysis of two basic properties, i.e., stability and convergence, is also given. In [12], the following is the essential form of the tempered FPDE governing multi-state European option pricing:

$$\mathcal{L}_j V_j(x,t) = \frac{\partial V_j(x,t)}{\partial t} + c_1\frac{\partial V_j(x,t)}{\partial x} + c_2 D_r^{\xi_j,\alpha_j} V_j(x,t) + c_3 D_l^{\lambda_j,\alpha_j} V_j(x,t) - rV_j(x,t) + \sum_{k=1}^{J} q_{j,k}V_k(x,t) = 0, \tag{1}$$

where $1 < \alpha_j < 2,\ j = 1, 2, \dots, J$. In (1), the constants c_1, c_2, and c_3 depend on the model, and the constants $q_{j,k}$ are those that satisfy the conditions: (i) $q_{j,k} \geq 0,\ \forall j \neq k$; (ii) $\sum_{k=1}^{J} q_{j,k} = 0$, and r is the risk-free rate. Furthermore, tempered fractional derivatives $D_r^{\xi_j,\alpha_j}V_j(x,t)$ and $D_l^{\xi_j,\alpha_j}V_j(x,t)$ are defined as follows:

$$D_r^{\xi_j,\alpha_j}V_j(x,t) = \frac{e^{\xi_j x}}{\Gamma(2-\alpha_j)}\frac{\partial^2}{\partial x^2}\int_x^{\infty}\frac{e^{-\xi_j\zeta}V_j(\zeta,t)}{(\zeta - x)^{\alpha_j-1}}\mathrm{d}\zeta - \xi_j^{\alpha_j}V_j(x,t),$$
$$D_l^{\lambda_j,\alpha_j}V_j(x,t) = \frac{e^{-\lambda_j x}}{\Gamma(2-\alpha_j)}\frac{\partial^2}{\partial x^2}\int_{-\infty}^{x}\frac{e^{\lambda_j\zeta}V_j(\zeta,t)}{(x-\zeta)^{\alpha_j-1}}\mathrm{d}\zeta - \lambda_j^{\alpha_j}V_j(x,t),$$

for $1 < \alpha_j < 2$. In this article, with the initial condition $V_j(x,T) = \max\{e^x - K, 0\}$, the CGMY model and the KoBoL model are both taken into consideration. The financial definitions of the parameters c_1, c_2, and c_3 have been given in [17], it is a CGMY model when $c_2 = c_3$, and it is a KoBoL model when $\xi_j = \lambda_j$. Additionally, the boundary conditions are $V_j(x_l,t) = \max\{(e^{x_l} - Ke^{-r(T-t)}), 0\}$ and $V_j(x_r,t) = \max\{(e^{x_r} - Ke^{-r(T-t)}), 0\}$.

2.1 Second-order discretization in space on the implicit scheme

In this subsection, the second-order discretization on the implicit scheme is developed. Because both of the operators $D_r^{\xi_j,\alpha_j}$ and $D_l^{\lambda_j,\alpha_j}$ are nonlocal, if the boundary conditions of the model are not zero, it is hard to discretize. To deal with this problem, the approach is similar to [20], where the model is converted into FPDEs with homogeneous Dirichlet boundary conditions with the boundary transform method. Let

$$F(x,t) = \frac{V_j(x_r,t) - V_j(x_l,t)}{e^{x_r} - e^{x_l}}(e^x - e^{x_l}) + V_j(x_l,t).$$

Thus, Model (1) can be converted into:

$$\begin{aligned}\mathcal{L}_j U_j(x,t) = {} & \frac{\partial U_j(x,t)}{\partial t} + c_1\frac{\partial U_j(x,t)}{\partial x} + c_2 D_r^{\xi_j,\alpha_j}U_j(x,t) \\ & + c_3 D_l^{\lambda_j,\alpha_j}U_j(x,t) - rU_j(x,t) + \sum_{k=1}^{J} q_{j,k}U_k(x,t) = f_j(x,t),\end{aligned} \tag{2}$$

where $U_j(x,t) = F(x,t) - V_j(x,t)$ and $f_j(x,t) = \mathcal{L}_j F(x,t)$. Furthermore, the boundary and initial conditions become

$$\begin{aligned} &U_j(x_l,t) = U_j(x_r,t) = 0, \quad t \in [0,T], \\ &U_j(x,T) = F(x,T) - V_j(x,T), \quad x \in (x_l, x_r). \end{aligned}$$

Let N and M be the positive integers. Then, the intervals $[x_l, x_r]$ and $[0,T]$ can be divided into $N+1$ and M sub-intervals, respectively. The spatial and temporal meshes are defined as follows:

$$\begin{aligned} x_i &= x_l + ih, \quad \text{for } i = 0, 1, \dots, N+1, \\ t_m &= T + m\tau, \quad \text{for } m = 0, 1, \dots, M, \end{aligned}$$

where $h = \frac{x_r - x_l}{N+1}$ and $\tau = -T/M < 0$.

The approximation provided in [22] can be used to estimate the two tempered fractional derivatives after the boundary transformation. Let $U_j(x,t) \in L^1(\mathbb{R})$, $D_r^{\xi_j,\alpha_j+1}U_j(x,t)$, $D_l^{\lambda_j,\alpha_j+1}U_j(x,t)$, and their Fourier transform belong to $L^1(\mathbb{R})$, then we obtain the approximation

$$\begin{aligned} D_r^{\xi_j,\alpha_j}U_j(x_i,t_m) &= \frac{1}{h^{\alpha_j}}\sum_{k=0}^{N-i+2} e^{-(k-1)\xi_j h}\omega_k^{\alpha_j,\xi_j}U_j(x_{i+k-1},t_m) + O(h^2),\\ D_l^{\lambda_j,\alpha_j}U_j(x_i,t_m) &= \frac{1}{h^{\alpha_j}}\sum_{k=0}^{i+1} e^{-(k-1)\lambda_j h}\omega_k^{\alpha_j,\lambda_j}U_j(x_{i-k+1},t_m) + O(h^2), \end{aligned} \tag{3}$$

where

$$\begin{aligned} &\omega_0^{\alpha_j,\lambda_j} = r_1 g_0^{\alpha_j}, \quad \omega_1^{\alpha_j,\lambda_j} = r_1 g_1^{\alpha_j} + r_2 g_0^{\alpha_j} - (r_1 e^{\lambda_j h} + r_2)(1-e^{-\lambda_j h})^{\alpha_j},\\ &\omega_j^{\alpha_j,\lambda_j} = r_1 g_j^{\alpha_j} + r_2 g_{j-1}^{\alpha_j}, \quad 2 \le j \le N. \end{aligned} \tag{4}$$

And $w_k^{\alpha_j,\xi_j}$ are defined in the same way. The coefficients are

$$\begin{aligned} &g_0^{\alpha_j} = 1, \quad g_j^{\alpha_j} = \left(1 - \frac{\alpha_j+1}{j}\right) g_{j-1}^{\alpha_j}, \quad j \ge 1,\\ &r_1 = \frac{\alpha_j}{2}, \quad r_2 = 1 - \frac{\alpha_j}{2}. \end{aligned}$$

Furthermore, the central-difference formula is used to discretize the derivative, which is represented as follows:

$$c_1 \frac{\partial U_j(x_i,t_m)}{\partial x} = c_1 \frac{U_j(x_{i+1},t_m) - U_j(x_{i-1},t_m)}{2h} + O(h^2). \tag{5}$$

Let the notation $u_{j,i}^m$ represent the numerical solution of $U_j(x_i,t_m)$ and $f_{j,i}^m = f_j(x_i,t_m)$. With the aforementioned numerical meshes, the converted Model (2) can be discretized by the fully implicit scheme with the approximation (3) and central-difference scheme (5), which is written as:

$$\begin{aligned} \tilde{\mathcal{L}}_j u_{j,i}^m = \frac{u_{j,i}^{m+1} - u_{j,i}^m}{\tau} + c_1 \frac{u_{j,i+1}^{m+1} - u_{j,i-1}^{m+1}}{2h} + \frac{c_2}{h^{\alpha_j}}\sum_{k=0}^{N-i+2} e^{-(k-1)\xi_j h}\omega_k^{\alpha_j,\xi_j}u_{j,i+k-1}^{m+1}\\ + \frac{c_3}{h^{\alpha_j}}\sum_{k=0}^{i+1} e^{-(k-1)\lambda_j h}\omega_k^{\alpha_j,\lambda_j}u_{j,i-k+1}^{m+1} - r u_{j,i}^{m+1} + \sum_{k=1}^{\bar{J}} q_{j,k}u_{j,i}^{m+1} = f_{j,i}^{m+1}, \end{aligned} \tag{6}$$

with error term of $O(|\tau| + h^2)$.

2.2 Matrix form for the second-order scheme

For simplicity, an n-by-n Toeplitz matrix is notated as $T_n(t_{n-1},\ldots,t_1;\, t_0;\, t_{-1},\ldots,t_{1-n})$, which is defined by:

$$T_n(t_{n-1},\ldots,t_1;\, t_0;\, t_{-1},\ldots,t_{1-n}) = \begin{bmatrix} t_0 & t_{-1} & \cdots & t_{2-n} & t_{1-n}\\ t_1 & t_0 & t_{-1} & \cdots & t_{2-n}\\ \vdots & t_1 & t_0 & \ddots & \vdots\\ t_{n-2} & \cdots & \ddots & \ddots & t_{-1}\\ t_{n-1} & t_{n-2} & \cdots & t_1 & t_0 \end{bmatrix}.$$

Let vectors $u^m = [u_{1,1}^m, u_{1,2}^m, \ldots, u_{1,N}^m, u_{2,1}^m, \ldots, u_{2,N}^m, \ldots, u_{\bar{J},N}^m]^T$ and $f^m = [f_{1,1}^m, f_{1,2}^m, \ldots, f_{1,N}^m, f_{2,1}^m, \ldots, f_{2,N}^m, \ldots, f_{\bar{J},N}^m]^T$, and matrix $Q = [q_{j,k}]_{j,k=1}^{\bar{J}}$. Then, the matrix form of the Scheme (6) can be expressed as:

$$[I_{\bar{J}N} + \mathrm{diag}(T_1, T_2, \ldots, T_{\bar{J}}) + \tau Q \otimes I_N]u^{m+1} = u^m + \tau f^{m+1}, \tag{7}$$

where

$$T_j = \frac{\tau c_1}{2h} A_j + \frac{\tau c_2}{h^{\alpha_j}} W_j + \frac{\tau c_3}{h^{\alpha_j}} G_j - \tau r I_N, \tag{8}$$

with

$$\begin{aligned}
A_j &= T_N(0, \dots, 0, -1;\ 0;\ 1, 0, \dots, 0),\\
W_j &= T_N(0, \dots, 0, e^{\xi_j h}\omega_0^{\alpha_j,\xi_j};\ e^0\omega_1^{\alpha_j,\xi_j};\ e^{-\xi_j h}\omega_2^{\alpha_j,\xi_j}, \dots, e^{-(N-1)\xi_j h}\omega_N^{\alpha_j,\xi_j}),\\
G_j &= T_N(e^{-(N-1)\lambda_j h}\omega_N^{\alpha_j,\lambda_j}, \dots, e^{-\lambda_j h}\omega_2^{\alpha_j,\lambda_j};\ e^0\omega_1^{\alpha_j,\lambda_j};\ e^{\lambda_j h}\omega_0^{\alpha_j,\lambda_j}, 0, \dots, 0),
\end{aligned}$$

and I_n represents the n-by-n identity matrix.

2.3 Stability and convergence analysis

In this subsection, the stability, and convergence of the second-order scheme are discussed. In the analysis, the properties of the coefficient matrix are primary. Hence, a lemma about the property of Toeplitz matrices is provided first.

Lemma 1. *Denote that $B_j = \frac{\tau c_2}{h^{\alpha_j}} W_j + \frac{\tau c_3}{h^{\alpha_j}} G_j$, then Matrix (8) can be written as $T_j = \frac{\tau c_1}{2h} A_j + B_j - \tau r I_N$. $\forall \mathbf{x} \in \mathbb{R}^N$, we have*

$$\mathbf{x}^T (T_j^T + T_j)\mathbf{x} \ge -2\tau r \mathbf{x}^T \mathbf{x}.$$

Proof. For $T_j = \frac{\tau c_1}{2h} A_j + B_j - \tau r I_N$, we have that

$$\mathbf{x}^T (T_j^T + T_j)\mathbf{x} = \frac{\tau c_1}{2h} \mathbf{x}^T (A_j^T + A_j)\mathbf{x} + \mathbf{x}^T (B_j^T + B_j)\mathbf{x} - 2\tau r \mathbf{x}^T \mathbf{x}.$$

Since $A_j = T_N(0, \dots, 0, -1;\ 0;\ 1, 0, \dots, 0)$ in (8), we have $\mathbf{x}^T (A_j^T + A_j)\mathbf{x} = 0$. Besides, $B_j^T + B_j$ is positive definite [22]; thus, $\mathbf{x}^T (B_j^T + B_j)\mathbf{x} \ge 0$, and the result holds. □

According to Lemma 1, the following lemma about the property of the coefficient matrix is obtained to support the analysis.

Lemma 2. *For all $\mathbf{x} \in \mathbb{R}^{\bar{J}N}$, denote $\mathbf{x} = [\mathbf{x}_1;\ \dots;\ \mathbf{x}_{\bar{J}}]$, where $\mathbf{x}_i \in \mathbb{R}^N$ for $i = 1, 2, \dots, \bar{J}$, and use $\tilde{\mathbf{x}}_i^T$ to denote the ith row of the matrix $[\mathbf{x}_1, \dots, \mathbf{x}_{\bar{J}}]$. Let $M = I_{\bar{J}N} + \operatorname{diag}(T_1, \dots, T_{\bar{J}}) + \tau Q \otimes I_N$ in (7), we have*

$$\mathbf{x}^T M^T M \mathbf{x} \ge (1 + K\tau)\mathbf{x}^T \mathbf{x},$$

where $K = 2(\|Q\|_2 - r)$.

Proof. The notation $\|\cdot\|_2$ denotes the matrix 2-norm. For convenience, let $R = \operatorname{diag}(T_1, \dots, T_{\bar{J}}) + \tau Q \otimes I_N$, then $M = I_{\bar{J}N} + R$. Since $R^T R$ is positive definite, it yields that

$$\begin{aligned}
\mathbf{x}^T M^T M \mathbf{x} &= \mathbf{x}^T (I + (R^T + R))\mathbf{x} + \mathbf{x}^T R^T R \mathbf{x}\\
&\ge \mathbf{x}^T \mathbf{x} + \mathbf{x}^T (R^T + R)\mathbf{x}\\
&= \mathbf{x}^T \mathbf{x} + \sum_{j=1}^{\bar{J}} \mathbf{x}_j^T (T_j^T + T_j)\mathbf{x}_j + \tau \sum_{i=1}^{N} \tilde{\mathbf{x}}_i^T (Q^T + Q)\tilde{\mathbf{x}}_i.
\end{aligned} \tag{9}$$

According to the definition of matrix 2-norm, we have

$$\tilde{\mathbf{x}}_i^T (Q^T + Q)\tilde{\mathbf{x}}_i \le \|Q^T + Q\|_2 \tilde{\mathbf{x}}_i^T \tilde{\mathbf{x}}_i \le 2\|Q\|_2 \tilde{\mathbf{x}}_i^T \tilde{\mathbf{x}}_i.$$

Because $\tau < 0$, we obtain that $\tau \sum_{i=1}^{N} \tilde{\mathbf{x}}_i^T (Q^T + Q)\tilde{\mathbf{x}}_i \ge 2\tau \|Q\|_2 \mathbf{x}^T \mathbf{x}$. With Lemma 1, equation (9) becomes

$$\mathbf{x}^T M^T M \mathbf{x} \geq (1 + K\tau)\mathbf{x}^T \mathbf{x},$$

where $K = 2(||Q||_2 - r)$. □

Then, the following lemma about the norm of the inverse of the coefficient matrix can be obtained with Lemma 2.

Lemma 3. *For* $0 > \tau \geq \frac{1}{4(r - ||Q||_2)}$, *we can obtain*

$$||M^{-1}||_2 \leq \frac{1}{\sqrt{1 + 2(||Q||_2 - r)\tau}} \leq 1 - K\tau,$$

where $K = 2(||Q||_2 - r)$.

Proof. Let the vector norm $||\cdot||_2$ be the 2-norm, and it is defined by $||\mathrm{x}||_2 = (\sum_{i=1}^{n}|x_i|^2)^{1/2}$ for any vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T \in \mathbb{C}^n$. With Lemma 2 and the definition of matrix 2-norm, we have

$$||M^{-1}||_2^2 = \max_x \frac{||M^{-1}\mathbf{x}||_2^2}{||\mathbf{x}||_2^2} = \max_{y = M^{-1}\mathbf{x}} \frac{||y||_2^2}{||My||_2^2} = \frac{1}{\min\limits_y \frac{||My||_2^2}{||y||_2^2}} \leq \frac{1}{1 + K\tau},$$

where $K = 2(||Q||_2 - r)$. Then, $||M^{-1}||_2 \leq \frac{1}{\sqrt{1 + K\tau}}$. In order to obtain the condition of τ, making $\frac{1}{\sqrt{1 + K\tau}} \leq 1 - K\tau$. To satisfy $\frac{1}{\sqrt{1 + K\tau}} \leq 1 - K\tau$, we use the basic solution equation to make $K\tau \geq -1/2$, so $\tau \geq \frac{1}{4(r - ||Q||_2)}$ is calculated. □

Remark 1. If $||Q||_2 - r \leq 0$, then $||M^{-1}||_2 \leq 1$. We assume $||Q||_2 - r > 0$ in the remaining context.

According to the aforementioned lemmas, the following theorems about stability and convergence can be analyzed. Let $||\cdot||_{l_2}$ be the l_2-norm, and the vector norm is defined by $||\mathrm{x}||_{l_2} = (h\sum_{i=1}^{n}|x_i|^2)^{1/2}$ for any vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T \in \mathbb{C}^n$, where h is the spatial step.

Theorem 4. (Stability) *The numerical solution obtained from* (6) *satisfies*

$$||u^{m+1}||_{l_2} \leq \exp(KT)||u^0||_{l_2} + T\exp(KT)f_{\max},$$

where $K = 2(||Q||_2 - r)$ *and* $f_{\max} = \max_{m \in \{0, \ldots, M-1\}}\{||f^{m+1}||_{l_2}\}$.

Proof. Since $M = I_{\tilde{J}_N} + \mathrm{diag}(T_1, \ldots, T_{\tilde{J}}) + \tau Q \otimes I_N$ and $Mu^{m+1} = u^m + \tau f^{m+1}$, with Lemma 3, we have

$$\begin{aligned}
||u^{m+1}||_{l_2} &= ||M^{-1}(u^m + \tau f^{m+1})||_{l_2} \\
&\leq ||M^{-1}||_2 ||u^m||_{l_2} + ||M^{-1}||_2 ||\tau f^{m+1}||_{l_2} \\
&\leq (1 - K\tau)||u^m||_{l_2} + (1 - K\tau)|\tau| f_{\max} \\
&\leq (1 - K\tau)^m ||u^0||_{l_2} + m(1 - K\tau)^m |\tau| f_{\max} \\
&\leq \exp(KT)||u^0||_{l_2} + T\exp(KT)f_{\max}.
\end{aligned}$$

□

Theorem 5. (Convergence) *Let* $U_j(x_i, t_m)$ *be the exact solution of* (1) *and* $u_{j,i}^m$ *be the solution of the implicit scheme* (6) *for* $1 < \alpha_j < 2$, *and* $0 \leq m \leq M - 1$, *so there exists a constant C that satisfies the following formula*:

$$||U_j(x_i, t_m) - u_{j,i}^m||_{l_2} \leq \exp(KT)C(|\tau| + h^2),$$

where $K = 2(||Q||_2 - r)$.

Proof. Let $e_{j,i}^m = U_j(x_i, t_m) - u_{j,i}^m$ be the truncation errors and make them satisfy the following formulas:

$$e^m = [e_{1,1}^m, e_{1,2}^m, \ldots, e_{1,N}^m, e_{2,1}^m, \ldots, e_{2,N}^m, \ldots, e_{\tilde{J},1}^m, \ldots, e_{\tilde{J},N}^m]^T,$$

since $e_{j,i}^0 = 0$, with Theorem 4 and error term in (6), it holds

$$\|\mathbf{e}^{m+1}\|_{l_2} \le \exp(KT)\|e^0\|_{l_2} + T\exp(KT)C(|\tau| + h^2) \le T\exp(KT)C(|\tau| + h^2).$$

Hence, the proof for convergence is done. □

With Theorems 4 and 5, we can know that the second-order scheme in (6) is stable and convergent.

3 Fast preconditioned iterative method

The fast Krylov subspace method is widely used to solve Toeplitz linear systems and is regarded as a fast solver [23]. We know that the computational cost per iteration of the fast Krylov subspace is $O(\bar{J}N\log N)$ using the properties of direct summation. In order to ensure the computation efficiency of the Krylov subspace method, it is an important step to solve the linear system using the preconditioners to compute the coefficient matrix efficiently. In this section, a block preconditioner is proposed and some theoretical guarantees for the preconditioner are presented, such as invertibility analysis and spectral analysis. Furthermore, to the best of our knowledge, a theoretically guaranteed preconditioner in a second-order scheme has not yet been proposed, so how to quickly compute the proposed preconditioner is explored in this section.

In view of the matrix form (7), if the Gaussian elimination approach is used, it yields an algorithm with $O(\bar{J}^3N^3)$ complexity. Then, in order to reduce the computational complexity greatly, a method using fast Fourier transformations (FFT) can be devised.

For the coefficient matrix $M = I_{\bar{J}N} + \mathrm{diag}(T_1, \ldots, T_{\bar{J}}) + \tau Q \otimes I_N \coloneqq B + \tau Q \otimes I_N$, suppose there is any vector x, then the first part of the multiplication Mx can be calculated by FFTs by embedding $I_N + T_j$ into a $2N$-by-$2N$ circulant matrix, and the second part is a Kronecker product $(\tau Q \otimes I_N)v$. Because the complexity of the N-by-N multiplication of Toeplitz matrix and vector is $O(N\log N)$, and based on the properties of the Kronecker product, the cost of Mx is $O(\bar{J}N\log N + \bar{J}^2N)$.

3.1 Block preconditioner

In this subsection, a block preconditioner is proposed. For the accuracy of the preconditioner, a block preconditioner P with matrix Q is proposed as:

$$P = I_{\bar{J}N} + \mathrm{diag}(s(T_1), s(T_2), \ldots, s(T_{\bar{J}})) + \tau Q \otimes I, \tag{10}$$

where $s(T)$ represents the Strang preconditioner for the Toeplitz matrix T, the size of matrix Q is $\bar{J} \times \bar{J}$, and the size of I is $N \times N$. More precisely, with the notation given in (8), $s(T_j)$ is defined as:

$$s(T_j) = \frac{\tau c_1}{h}s(A_j) + \frac{\tau c_2}{h^{\alpha_j}}s(W_j) + \frac{\tau c_3}{h^{\alpha_j}}s(G_j) - \tau r I_N,$$

where

$$s(A_j) = T_N(1, 0, \ldots, 0, -1;\ 0;\ 1, 0, \ldots, 0, -1),$$

$$s(W_j) = T_N\Bigg(e^{-\xi_j h}\omega_2^{\alpha_j,\xi_j}, \ldots, e^{-\left(\left\lfloor\frac{N+1}{2}\right\rfloor - 1\right)\xi_j h}\omega_{\left\lfloor\frac{N+1}{2}\right\rfloor}^{\alpha_j,\xi_j}, 0, \ldots, 0, e^{\xi_j h}\omega_0^{\alpha_j,\xi_j};\ e^0\omega_1^{\alpha_j,\xi_j};$$
$$e^{-\xi_j h}\omega_2^{\alpha_j,\xi_j}, \ldots, e^{-(\left\lfloor\frac{N+1}{2}\right\rfloor - 1)\xi_j h}\omega_{\left\lfloor\frac{N+1}{2}\right\rfloor}^{\alpha_j,\xi_j}, 0, \ldots, 0, e^{\xi_j h}\omega_0^{\alpha_j,\xi_j}\Bigg),$$

$$s(G_j) = T_N\Bigg(e^{\lambda_j h}\omega_0^{\alpha_j,\lambda_j}, 0, \ldots, 0, e^{-\left(\left\lfloor\frac{N+1}{2}\right\rfloor - 1\right)\lambda_j h}\omega_{\left\lfloor\frac{N+1}{2}\right\rfloor}^{\alpha_j,\lambda_j}, \ldots, e^{-\lambda_j h}\omega_2^{\alpha_j,\lambda_j};\ e^0\omega_1^{\alpha_j,\lambda_j};$$
$$e^{\lambda_j h}\omega_0^{\alpha_j,\lambda_j}, 0, \ldots, 0, e^{-\left(\left\lfloor\frac{N+1}{2}\right\rfloor - 1\right)\lambda_j h}\omega_{\left\lfloor\frac{N+1}{2}\right\rfloor}^{\alpha_j,\lambda_j}, \ldots, e^{-\lambda_j h}\omega_2^{\alpha_j,\lambda_j}\Bigg).$$

3.2 Invertibility analysis and spectral distribution

Theorem 6. (Invertibility) *The block preconditioner* $P = I_{\bar{J}N} + \mathrm{diag}(s(T_1), s(T_2), \ldots, s(T_{\bar{J}})) + \tau Q \otimes I$ *is invertible.*

Proof. Let the notation $\lambda_r(s(W_j + W_j^T))$ represent the r-th eigenvalue of $s(W_j + W_j^T)$. With the properties of ω_k in [22], it holds that

$$\begin{aligned}\lambda_r(s(W_j + W_j^T)) &= 2\left(e^0\omega_1^{\alpha_j,\xi_j} + (e^{\xi_j h}\omega_0^{\alpha_j,\xi_j} + e^{-\xi_j h}\omega_2^{\alpha_j,\xi_j})\cos\frac{2\pi(r-1)}{N} + \sum_{k=3}^{\frac{N}{2}} e^{-(k-1)\xi_j h}\omega_k^{\alpha_j,\xi_j}\cos\frac{2\pi(k-1)(r-1)}{N}\right)\\ &\le 2\left(e^0\omega_1^{\alpha_j,\xi_j} + (e^{\xi_j h}\omega_0^{\alpha_j,\xi_j} + e^{-\xi_j h}\omega_2^{\alpha_j,\xi_j}) + \sum_{k=3}^{\frac{N}{2}} e^{-(k-1)\xi_j h}\omega_k^{\alpha_j,\xi_j}\right) < 0, \quad \text{for } r = 1, \ldots, N.\end{aligned} \tag{11}$$

Then, we have

$$\mathbf{x}^T s(W_j + W_j^T)\mathbf{x} \le 0, \quad \forall \mathbf{x} \in \mathbb{R}^N. \tag{12}$$

Similarly, we have $\mathbf{x}^T s(G_j + G_j^T)\mathbf{x} \le 0, \quad \forall \mathbf{x} \in \mathbb{R}^N$. And there is

$$s(T_j) + s(T_j)^T = s(T_j + T_j^T) = \frac{\tau c_2}{h^{\alpha_j}} s(W_j + W_j^T) + \frac{\tau c_3}{h^{\alpha_j}} s(G_j + G_j^T) - 2\tau r I_N, \tag{13}$$

so we can obtain that for $\forall \mathbf{x} \in \mathbb{R}^N$,

$$\mathbf{x}^T(s(T_j) + s(T_j)^T)\mathbf{x} \ge -2\tau r \mathbf{x}^T\mathbf{x}. \tag{14}$$

With Lemma 2, using similar techniques, we have

$$\mathbf{x}^T P^T P\mathbf{x} \ge (1 + K\tau)\mathbf{x}^T\mathbf{x}, \tag{15}$$

where $K = 2(\|Q\|_2 - r)$. Besides, according to Lemma 3, we know that for $\tau \ge \frac{1}{4(r - \|Q\|_2)}$, the minimum singular value of P can be calculated, i.e., $\sigma_{\min}(P) \ge \frac{1}{\sqrt{2}}$. Therefore, we have proved that P is invertible. □

In the following lemmas and theorems, the spectral distribution of the block preconditioner P can be proved.

Lemma 7. *For* $0 > \tau \ge \frac{1}{4(r - \|Q\|_2)}$, *we can obtain*

$$\|P^{-1}\|_2 \le 1 - K\tau,$$

where $K = 2(\|Q\|_2 - r)$.

Proof. With Lemma 3 and Theorem 6, we can know that P^{-1} exists and $\mathbf{x}^T P^T P\mathbf{x} \ge (1 + K\tau)\mathbf{x}^T\mathbf{x}$, for $0 > \tau \ge \frac{1}{4(r - \|Q\|_2)}$. Similar to the proof of Lemma 3, by the definition of matrix 2-norm, $\|P^{-1}\|_2 \le 1 - K\tau$ is proved, where $K = 2(\|Q\|_2 - r)$. □

Lemma 8. [24] *For any* $\varepsilon > 0$, *there exists a constant* N_c, *for all* $N > N_c$ *such that*

$$T_j - s(T_j) = K_j + L_j,$$

where $\|K_j\|_2 \le \varepsilon/\bar{J}$ *and* $\mathrm{rank}\,(L_j) \le N_c$.

Hence, with Lemma 8, *the spectra distribution of the block preconditioner* P (10) *can be derived.*

Theorem 9. *For any* $\varepsilon > 0$, *there exists a constant* N_c, *for all* $N > N_c$ *such that*

$$P^{-1}M = I_{\bar{J}N} + \widetilde{K}_m + \tilde{L}_m,$$

where $\|\widetilde{K}_m\|_2 \le (1 - K\tau)\varepsilon$ *with* $K = 2(\|Q\|_2 - r)$ *and* $\mathrm{rank}\,(\tilde{L}_m) \le \bar{J}N_c$, *and M is the coefficient matrix.*

Proof. With Lemma 8, it holds

$$\begin{aligned} M-P &= I_{\bar{J}N} + \mathrm{diag}(T_1, T_2, \ldots, T_{\bar{J}}) + \tau Q \otimes I_N - I_{\bar{J}N} - \mathrm{diag}(s(T_1), s(T_2), \ldots, s(T_{\bar{J}})) - \tau Q \otimes I_N \\ &= \mathrm{diag}(T_1 - s(T_1), T_2 - s(T_2), \ldots, T_{\bar{J}} - s(T_{\bar{J}})) \\ &= \mathrm{diag}(K_1, K_2, \ldots, K_{\bar{J}}) + \mathrm{diag}(L_1, L_2, \ldots, L_{\bar{J}}). \end{aligned}$$

Let $\bar{K}_m = \mathrm{diag}(K_1, K_2, \ldots, K_{\bar{J}})$ and $\bar{L}_m = \mathrm{diag}(L_1, L_2, \ldots, L_{\bar{J}})$. It is derived that

$$\|\bar{K}_m\|_2 \le \varepsilon \quad \text{and} \quad \mathrm{rank}\,(\bar{L}_m) \le \bar{J} N_c.$$

Hence, we have

$$P^{-1}M - I_{\bar{J}N} = P^{-1}(M-P) = P^{-1}\bar{K}_m + P^{-1}\bar{L}_m.$$

Denote that $\widetilde{K}_m = P^{-1}\bar{K}_m$ and $\widetilde{L}_m = P^{-1}\bar{L}_m$. Besides, with Lemma 7, we have

$$\|P^{-1}\|_2 \le 1 - 2(\|Q\|_2 - r)\tau \coloneqq 1 - K\tau,$$

where $K = 2(\|Q\|_2 - r)$. Then, according to the inequality, we have

$$\|\tilde{K}_m\|_2 \le \|P^{-1}\|_2 \|\bar{K}_m\|_2 \le (1-K\tau)\varepsilon, \quad \mathrm{rank}\,(\widetilde{L}_m) \le \min\{\mathrm{rank}\,(P^{-1}), \mathrm{rank}\,(\bar{L}_m)\} \le \bar{J} N_c. \square$$

After analyzing the small norm and low rank of the block preconditioner P (10), we are looking forward to the rate of convergence of the iterative method.

3.3 Implementation of P^{-1}

With the guarantee of invertibility, the block preconditioner P (10) can be used to speed up the rate of convergence of the iterative method. Based on the structure of the proposed preconditioner, the matrix-vector product $P^{-1}v$ can be computed by:

$$P^{-1}v = (I_{\bar{J}} \otimes F_N^*)[\mathrm{diag}(\Lambda_1, \Lambda_2, \ldots, \Lambda_{\bar{J}}) + \tau Q \otimes I]^{-1}(I_{\bar{J}} \otimes F_N)v, \tag{16}$$

where Λ_j is a diagonal matrix containing all eigenvalues of $I_{\bar{J}N} + \mathrm{diag}(s(T_1), s(T_2), \ldots, s(T_{\bar{J}}))$. It is worth noting that the entries of Λ_j can be gotten in $O(N \log N)$ operations [23].

However, because it is difficult to deal with the inverse part of $Q \otimes I$ in (16), $Q \otimes I$ can be refactored into $I \otimes Q$ using permutation, making the middle part of the preconditioner P into a block diagonal matrix. At the same time, $\mathrm{diag}(\Lambda_1, \Lambda_2, \ldots, \Lambda_{\bar{J}})$ in the block preconditioner P also needs to be changed into $\mathrm{diag}(\widetilde{\Lambda_1}, \widetilde{\Lambda_2}, \ldots, \widetilde{\Lambda_N})$ accordingly. Let "$\widetilde{P}$" be the permuted form of P.

Then, a permutation matrix Z can be used to make it possible to compute $P^{-1}v$ by computing $\widetilde{P}^{-1}v$. Therefore, after the matrix permutation, the matrix-vector product $\widetilde{P}^{-1}v$ can be computed by:

$$\widetilde{P}^{-1}v = (I_{\bar{J}} \otimes F_N^*)Z^T[\mathrm{diag}(\widetilde{\Lambda_1}, \widetilde{\Lambda_2}, \ldots, \widetilde{\Lambda_N}) + I \otimes \tau Q]^{-1}Z(I_{\bar{J}} \otimes F_N)v. \tag{17}$$

Remark 2. Suppose that there is a permutation matrix Z satisfying $Z(\tau Q \otimes I)Z^T = I \otimes \tau Q$ such that $P^{-1}v = \widetilde{P}^{-1}v$. In order to simplify the notations, $\mathrm{diag}(\Lambda_1, \Lambda_2, \ldots, \Lambda_{\bar{J}})$ is written as D. According to the permutation matrix Z, $\widetilde{P}^{-1}v$ can be written as:

$$\begin{aligned} \widetilde{P}^{-1}v &= (I_{\bar{J}} \otimes F_N^*)Z^T(ZDZ^T + Z(\tau Q \otimes I)Z^T)^{-1}Z(I_{\bar{J}} \otimes F_N)v \\ &= (I_{\bar{J}} \otimes F_N^*)Z^TZ(D + \tau Q \otimes I)^{-1}Z^TZ(I_{\bar{J}} \otimes F_N)v \\ &= (I_{\bar{J}} \otimes F_N^*)(D + \tau Q \otimes I)^{-1}(I_{\bar{J}} \otimes F_N)v \\ &= P^{-1}v. \end{aligned}$$

Then, we can obtain the matrix-vector product $P^{-1}v = \widetilde{P}^{-1}v$ by permutation methods.

After the permutation method transformation in equation (16) to (17), the matrix is changed from a block matrix of $\bar{J}$ blocks $N \times N$ to a matrix of N blocks $\bar{J} \times \bar{J}$. From (17), we can see that the hardest part is the inverse in the product.

In order to clearly represent the notations, $Z(I_{\bar{J}} \otimes F_N)v$ can be recorded as $V := [V_{1,1}, V_{2,1}, \ldots, V_{\bar{J},1}, V_{1,2}, \ldots, V_{\bar{J},2}, V_{1,N}, \ldots, V_{\bar{J},N}]^T := [V_1, V_2, \ldots, V_N]^T$. Then, $[\Lambda_P + I \otimes \tau Q]^{-1} Z(I_{\bar{J}} \otimes F_N)v$ can be written as $[\Lambda_P + I \otimes \tau Q]^{-1} V$. Besides, denote the $\mathrm{diag}(\Lambda_1, \Lambda_2, \ldots, \Lambda_{\bar{J}})$ in (16) as Λ_P, and $\mathrm{diag}(\widetilde{\Lambda_1}, \widetilde{\Lambda_2}, \ldots, \widetilde{\Lambda_N})$ in (17) as $\Lambda_{\tilde{P}}$, then we have

$$[\Lambda_{\tilde{P}} + I \otimes \tau Q]^{-1} V = \begin{pmatrix} (\tau Q + \widetilde{\Lambda_1})^{-1} V_1 & & & \\ & (\tau Q + \widetilde{\Lambda_2})^{-1} V_2 & & \\ & & \ddots & \\ & & & (\tau Q + \widetilde{\Lambda_N})^{-1} V_N \end{pmatrix}. \tag{18}$$

In order to speed up the matrix-vector product calculation, the Sherman-Morrison formula can be considered to solve the block diagonal matrix (18).

3.3.1 Case 1: rank (Q) = 1

The Sherman-Morrison formula [25] can only be used when the rank of the matrix is 1. Therefore, the case is considered when the rank of Q is 1.

Lemma 10. (Sherman-Morrison formula) [25] *Suppose that $A \in C^{n\times n}$ is an invertible matrix and $u, v \in C^n$ are column vectors, then $A + uv^T$ is invertible iff $1 + v^T A^{-1} u \neq 0$. In this case, $(A + uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^T A^{-1}}{1 + v^T A^{-1} u}$.*

The Sherman-Morrison formula $(A + uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^T A^{-1}}{1 + v^T A^{-1} u}$ can be used to solve the inverse $(\tau Q + \widetilde{\Lambda_i})^{-1} V_i$ in each block (18) for $i = 1, 2, \ldots, N$. In this case, "A" in the Sherman-Morrison formula equals every $\widetilde{\Lambda_i}$ in (18), and uv^T is τQ. Then, we have: $[\Lambda_{\tilde{P}} + I \otimes \tau Q]^{-1} V = \mathrm{diag}((\widetilde{\Lambda_1} + uv^T)^{-1} V_1, \ldots, (\widetilde{\Lambda_N} + uv^T)^{-1} V_N)$, where $(\widetilde{\Lambda_i} + uv^T)^{-1} = (\widetilde{\Lambda_i})^{-1} - \frac{(\widetilde{\Lambda_i})^{-1} uv^T (\widetilde{\Lambda_i})^{-1}}{1 + v^T (\widetilde{\Lambda_i})^{-1} u}$ for $i = 1, 2, \ldots, N$.

Because $\Lambda_{\tilde{P}}$ is a diagonal matrix, it is easy to calculate $\Lambda_{\tilde{P}}^{-1} V$. Then, the point is to calculate $\frac{(\widetilde{\Lambda_i})^{-1} uv^T (\widetilde{\Lambda_i})^{-1}}{1 + v^T (\widetilde{\Lambda_i})^{-1} u} V_i$ for each block. For faster computation, the Hadamard product can be used. The Hadamard product $A \circ B$ is a matrix with elements $(A \circ B)_{ij} = (A)_{ij}(B)_{ij}$ for two matrices A and B of the same dimension $m \times n$, which is ".*" command in Matlab.

Remark 3. Λ_j is a diagonal matrix that contains all of the eigenvalues of the matrix $I_{\bar{J}N} + \mathrm{diag}(s(T_1), s(T_2), \ldots, s(T_{\bar{J}}))$, and by Theorem 6, similar to the proof of invertibility. It is obvious that $I_N + s(T_j)$ is invertible, so all Λ_j are not 0.

Similar to the Hadamard product, another symbol $\bar{\circ}$ is defined. $A \,\bar{\circ}\, B$ is a matrix with elements $(A \,\bar{\circ}\, B)_{ij} = (A)_{ij}/(B)_{ij}$ for two matrices A and B of the same dimension $m \times n$, which is "./" command in Matlab. Then, the fractional part of the formula becomes

$$\begin{pmatrix} \dfrac{((\widetilde{\Lambda_1})^{-1} uv^T (\widetilde{\Lambda_1})^{-1} V_1)^T}{1 + v^T (\widetilde{\Lambda_1})^{-1} u} \\ \dfrac{((\widetilde{\Lambda_2})^{-1} uv^T (\widetilde{\Lambda_2})^{-1} V_2)^T}{1 + v^T (\widetilde{\Lambda_2})^{-1} u} \\ \vdots \\ \dfrac{((\widetilde{\Lambda_N})^{-1} uv^T (\widetilde{\Lambda_N})^{-1} V_N)^T}{1 + v^T (\widetilde{\Lambda_N})^{-1} u} \end{pmatrix} = R_1 \,\bar{\circ}\, R_2, \tag{19}$$

where

$$R_1 = \begin{pmatrix} \frac{1}{\Lambda_{1,1}} & \cdots & \frac{1}{\Lambda_{\bar{J},1}} \\ \frac{1}{\Lambda_{1,2}} & \cdots & \frac{1}{\Lambda_{\bar{J},2}} \\ \vdots & & \vdots \\ \frac{1}{\Lambda_{1,N}} & \cdots & \frac{1}{\Lambda_{\bar{J},N}} \end{pmatrix} \circ \begin{pmatrix} u^T \\ u^T \\ \vdots \\ u^T \end{pmatrix} \circ \left(\begin{pmatrix} \frac{1}{\Lambda_{1,1}} & \cdots & \frac{1}{\Lambda_{\bar{J},1}} \\ \frac{1}{\Lambda_{1,2}} & \cdots & \frac{1}{\Lambda_{\bar{J},2}} \\ \vdots & & \vdots \\ \frac{1}{\Lambda_{1,N}} & \cdots & \frac{1}{\Lambda_{\bar{J},N}} \end{pmatrix} \circ \begin{pmatrix} V_{1,1} & \cdots & V_{\bar{J},1} \\ V_{1,2} & \cdots & V_{\bar{J},2} \\ \vdots & & \vdots \\ V_{1,N} & \cdots & V_{\bar{J},N} \end{pmatrix} v \right),$$

and

$$R_2 = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} + \begin{pmatrix} v^T \\ v^T \\ \vdots \\ v^T \end{pmatrix} \circ \begin{pmatrix} \frac{1}{\Lambda_{1,1}} & \cdots & \frac{1}{\Lambda_{\bar{J},1}} \\ \frac{1}{\Lambda_{1,2}} & \cdots & \frac{1}{\Lambda_{\bar{J},2}} \\ \vdots & & \vdots \\ \frac{1}{\Lambda_{1,N}} & \cdots & \frac{1}{\Lambda_{\bar{J},N}} \end{pmatrix} u. \tag{20}$$

Besides, the part of Formula (19) has repetitions in the numerator and denominator, in the calculation, a part of the calculation can be reduced. And a large number of loop computing can be avoided during calculation using the Hadamard product since vector multiplication can be done directly, which greatly reduces the amount of calculation and speeds up the calculation. Therefore, the operation cost of the matrix-vector product $\widetilde{P}^{-1}v$ (17) is $O(\bar{J}N\log N + \bar{J}N)$. For each iteration, the cost of this linear system using the preconditioner by the fast Krylov subspace approach is $O(\bar{J}N\log N + \bar{J}^2N + \bar{J}N)$.

Remark 4. The ILU factorization is used when rank $(Q) > 1$. In Lemma 10, since $\tau Q = uv^T$ can only be used when rank Q is 1, the application of the Sherman-Morrison formula is limited when the rank $(Q) > 1$. Thus, a more general approach is considered, which is to solve $\widetilde{P}^{-1}v$ in (17) directly by decomposing the permuted matrix via ILU factorization.

4 Numerical experiments

For the experiments, all numerical experiments are carried out in MATLAB (R2019b) on a Laptop with configuration: Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz and 16.0GB RAM. In order to better compare the effectiveness of the proposed preconditioner, the Strang preconditioner is used for comparison [15], which is

$$P_m = I_{\bar{J}N} + \mathrm{diag}(s(T_1), s(T_2), \ldots, s(T_{\bar{J}})). \tag{21}$$

When solving the linear system, the generalized minimum residual (GMRES) approach is selected as the Krylov subspace method, the restart is 20, and the stopping criterion is 10^{-10}. In addition, the zero vector is chosen as the initial guess of the iterative method.

4.1 Example 1

A coupled tempered FPDE with a known exact solution is taken into consideration in order to illustrate the precision of the suggested implicit method and is presented as:

$$\begin{cases} \dfrac{\partial U_j(x,t)}{\partial t} - D_l^{\lambda_j,\alpha_j} U_j(x,t) - \sum\limits_{k=1}^{\bar{J}} q_{j,k} U_j(x,t) = f_j(x,t), \\ U_j(0,t) = 0, \quad 0 < t \le 1, \\ U_j(1,t) = e^{-t-\lambda_j}, \quad 0 < t \le 1, \\ U_j(x,0) = e^{-\lambda_j x} x^{2+\alpha_j}, \quad 0 \le x \le 1, \end{cases}$$

with $f_j(x,t) = -e^{-t-\lambda_j x}\left(\frac{\Gamma(3+\alpha_j)}{\Gamma(3)}x^2 + x^{2+\alpha_j} - \lambda_j^{\alpha_j} x^{2+\alpha_j}\right) - \sum_{k=1}^{\bar{J}} q_{j,k} U_j(x,t)$, and the exact answer is $U_j(x,t) = e^{-t-\lambda_j x} x^{2+\alpha_j}$. The parameters $q_{j,k}$, α_j, and λ_j for the examples below are created at random by MATLAB and are provided as follows:

(a) $Q = [-127, 127;\ 115, -115]$, $\alpha = [1.8, 1.9]$, $\lambda = [0.21, 4.76]$, $\bar{J} = 2$;

(b) $Q = [-170, 28, 142;\ 10, -70, 60;\ 122, 71, -193]$, $\alpha = [1.9, 1.5, 1.6]$, $\lambda = [0.68, 3.4, 0.86]$, $\bar{J} = 3$;

(c) $Q = [-191, 74, 1, 116;\ 87, -160, 71, 2;\ 13, 62, -164, 89;\ 55, 1, 101, -157]$, $\alpha = [1.9, 1.7, 1.9, 1.6]$, $\lambda = [1.7, 0.4, 0.86, 1.63]$, $\bar{J} = 4$;

(d) $Q = [-403, 4, 61, 55, 4, 6, 135, 138;\ 3, -494, 74, 67, 7, 24, 150, 169;\ 114, 74, -377, 8, 112, 58, 2, 9;\ 119, 132, 2, -385, 53, 71, 2, 6;\ 5, 8, 63, 53, -396, 9, 127, 131;\ 7, 6, 55, 67, 5, -411, 136, 135;\ 66, 65, 4, 7, 58, 118, -335, 17;\ 70, 90, 8, 3, 59, 105, 35, -370]$, $\alpha = [1.7, 1.7, 1.9, 1.7, 1.7, 1.6, 1.6, 1.7]$, $\lambda = [2.04, 4.1, 3.6, 2.65, 2.66, 1.63, 0.53, 3.06]$, $\bar{J} = 8$.

In Table 1, we show the error and convergence rate of the implicit second-order scheme. In this table, the Strang preconditioner P_m (21) is used to speed the convergence rate of the GMRES technique when computing errors. Furthermore, "Error" symbolizes the l_2-norm of the errors, while "Rate" describes the convergence rates.

From Table 1, it is apparent that the scheme is stable with a second-order convergence rate when $N^2 = M$. And as the number of grid points increases, the convergence rate approaches two in four cases.

In Table 2, because the length of space is significantly more than the length of time in a real financial market, the N we chose in Table 2 is bigger than M. We analyze the second-order scheme and use "IM-nP," "IM-P," and "IM-b-P" to signify the GMRES method without preconditioners, the GMRES approach with the Strang circulant preconditioner P_m (21), and the GMRES method using the Sherman-Morrison formula with the block preconditioner P (10) when $\bar{J}$ is 2 and ILU factorization when $\bar{J} > 2$. Moreover, "Ite," and "CPU" represent the average iterations and CPU time, respectively.

From Table 2, it can be seen that when the number of states $\bar{J}$ is 2, the GMRES method using the proposed preconditioner P (10) with the Sherman-Morrison formula has the best performance in terms of both the number of iterations and CPU time. Besides, when $\bar{J} > 2$, the GMRES method with the block preconditioner P (10) using ILU factorization is clearly best, although the number of iterations using the GMRES method with the Strang preconditioner P_m (21) is already much optimized compared to that of without preconditioner. By the CPU time, the GMRES method using the proposed preconditioner P (10) is the fastest method in these methods. In particular, in case (d), the GMRES method with the proposed preconditioner P (10) using ILU factorization takes 5 s, while the GMRES method without preconditioners takes more than 200 min.

Table 1: Errors and convergence rate for Example 1

N	M	Case (a)		Case (b)		Case (c)		Case (d)	
		Error	**Rate**	**Error**	**Rate**	**Error**	**Rate**	**Error**	**Rate**
2^4	2^8	1.5048×10^{-4}	—	3.4497×10^{-4}	—	4.3970×10^{-4}	—	3.2565×10^{-4}	—
2^5	2^{10}	4.0101×10^{-5}	1.9079	9.1258×10^{-5}	1.9184	1.1631×10^{-4}	1.9185	8.6051×10^{-5}	1.9201
2^6	2^{12}	1.0336×10^{-5}	1.9560	2.3469×10^{-5}	1.9592	2.9931×10^{-5}	1.9583	2.2122×10^{-5}	1.9597
2^7	2^{14}	2.6227×10^{-6}	1.9786	5.9502×10^{-6}	1.9797	7.5924×10^{-6}	1.9790	5.6092×10^{-6}	1.9796

4.2 Example 2

In this example, the multi-state KoBoL model (1) is used to evaluate the European call option. The following are the basic parameters: $x_r = \ln(120)$, $x_l = \ln(0.1)$, $K = 70$, $r = 0.05$, and $T = 1$. Four different cases are investigated, and their parameter components are generated at random using Matlab, as follows:

(a) $Q = [-48, 48;\ 45, -45]$, $\alpha = [1.7, 1.6]$, $\lambda = [3.92, 2.66]$, $\sigma = [0.99, 0.21]$, $p = [0.91, 0.42]$, $\bar{J} = 2$;

(b) $Q = [-80, 80;\ 52, -52]$, $\alpha = [1.9, 1.7]$, $\lambda = [1.02, 3.76]$, $\sigma = [0.79, 0.34]$, $p = [0.81, 0.31]$, $\bar{J} = 2$;

(c) $Q = [-170, 28, 142;\ 10, -70, 60;\ 122, 71, -193]$, $\alpha = [1.9, 1.5, 1.6]$, $\lambda = [0.68, 3.4, 0.86]$, $\sigma = [0.87, 0.90, 0.64]$, $p = [0.83, 0.5, 0.3]$, $\bar{J} = 3$;

(d) $Q = [-191, 74, 1, 116;\ 87, -170, 81, 2;\ 13, 62, -164, 89;\ 75, 1, 111, -187]$, $\alpha = [1.9, 1.9, 1.1, 1.8]$, $\lambda = [1.7, 3.5, 2.88, 1.63]$, $\sigma = [0.97, 0.30, 0.43, 0.64]$, $p = [0.2, 0.9, 0.4, 0.8]$, $\bar{J} = 4$.

In Example 2, the different methods with preconditioners are compared in four different cases. In Table 3, we can see that although the Strang preconditioner P_m (21) has improved a lot of iterations, the proposed preconditioner P (10) is better in terms of both iterations and CPU time in these four cases. In particular, the GMRES method using the proposed preconditioner P (10) with the Sherman-Morrison formula is the fastest among these methods by CPU time when $\bar{J}$ is 2. However, when $\bar{J}$ is greater than 2, the GMRES method with the proposed preconditioner P (10) using ILU factorization takes the least amount of time and iterations.

4.3 Example 3

In Example 3, the efficacy of the tempered fractional model and the precision of the proposed numerical approach are demonstrated in a real-world experiment. In this example, the problem of calibrating the European options is compared using the two-state KoBoL model and the BS model. We use the market data

Table 2: Iteration numbers and CPU time of three methods for Example 1

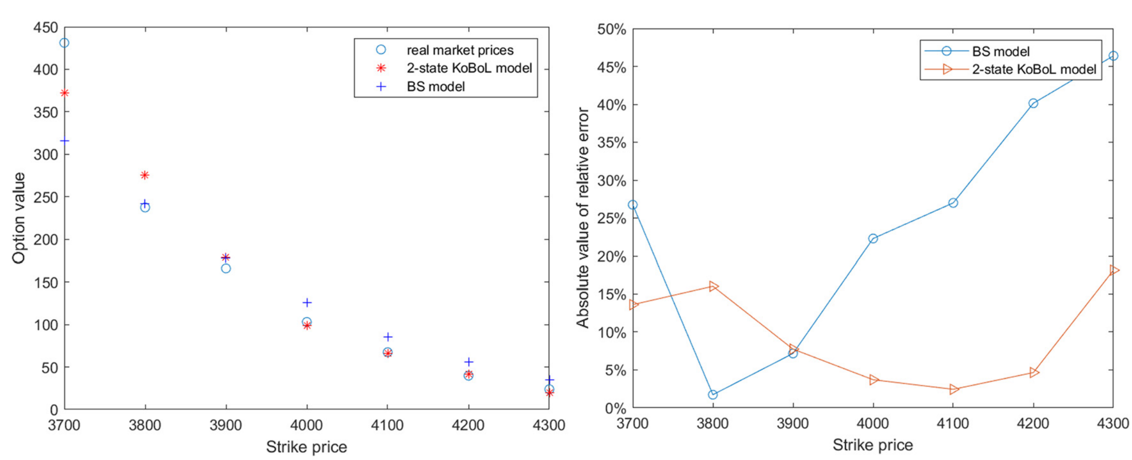
N	M	IM-nP		IM-P		IM-b-P	
		Ite	CPU	Ite	CPU	Ite	CPU
Case (a)							
2^8	2^4	1335.3	1.12 s	19.0	0.05 s	8.0	0.02 s
2^9	2^5	3855.7	9.00 s	19.0	0.08 s	8.0	0.05 s
2^{10}	2^6	9765.9	75.68 s	18.0	0.25 s	8.0	0.14 s
2^{11}	2^7	22051.0	1103.24 s	18.0	1.13 s	8.0	0.58 s
Case (b)							
2^8	2^4	4323.9	4.64 s	40.0	0.07 s	10.0	0.03 s
2^9	2^5	9403.1	28.41 s	37.0	0.18 s	10.0	0.09 s
2^{10}	2^6	20435.0	383.35 s	32.0	0.60 s	10.0	0.26 s
2^{11}	2^7	36195.0	3380.75 s	27.0	2.64 s	10.0	0.29 s
Case (c)							
2^8	2^4	2943.4	3.57 s	54.0	0.14 s	12.0	0.04 s
2^9	2^5	7840.5	31.19 s	45.0	0.32 s	12.0	0.13 s
2^{10}	2^6	18621.0	505.71 s	35.0	1.11 s	11.0	0.54 s
2^{11}	2^7	36383.0	4169.38 s	29.0	3.98 s	11.0	1.95 s
Case (d)							
2^8	2^4	3058.9	6.01 s	138.0	0.39 s	17.0	0.09 s
2^9	2^5	8194.8	138.55 s	100.0	2.13 s	17.0	0.50 s
2^{10}	2^6	19886.0	962.88 s	81.0	5.00 s	16.0	1.55 s
2^{11}	2^7	40039.0	14661.27 s	60.0	10.83 s	16.0	5.10 s

Table 3: Iteration numbers and CPU time of three methods for Example 2

N	M	IM-nP		IM-P		IM-b-P	
		Ite	CPU	Ite	CPU	Ite	CPU
Case (a)							
2^8	2^4	49.3	0.06 s	18.7	0.05 s	7.0	0.02 s
2^9	2^5	63.7	0.20 s	15.7	0.08 s	7.0	0.04 s
2^{10}	2^6	86.8	0.80 s	12.9	0.18 s	8.0	0.13 s
2^{11}	2^7	118.9	5.62 s	11.9	0.73 s	8.0	0.60 s
Case (b)							
2^8	2^4	64.6	0.09 s	24.9	0.08 s	7.0	0.02 s
2^9	2^5	88.9	0.29 s	17.81	0.09 s	8.0	0.05 s
2^{10}	2^6	127.0	1.09 s	14.9	0.23 s	8.0	0.14 s
2^{11}	2^7	185.2	8.91 s	13.0	0.78 s	8.0	0.65 s
Case (c)							
2^8	2^4	73.6	0.30 s	38.50	0.22 s	9.0	0.04 s
2^9	2^5	98.9	1.12 s	29.44	0.41 s	10.0	0.13 s
2^{10}	2^6	142.0	6.90 s	22.50	1.36 s	10.0	0.59 s
2^{11}	2^7	214.9	39.07 s	17.94	3.82 s	10.9	2.42 s
Case (d)							
2^8	2^4	83.7	0.15 s	40.8	0.15 s	9.0	0.03 s
2^9	2^5	114.5	0.51 s	31.4	0.22 s	9.8	0.10 s
2^{10}	2^6	166.5	3.66 s	25.6	0.83 s	10.0	0.43 s
2^{11}	2^7	254.3	25.93 s	19.9	2.84 s	10.9	1.84 s

of options contract IO2305-C.CFE on March. 10, 2023, to compare the results of calibrating the two-state KoBoL model and the BS model. The parameter settings are as follows: $N = 2^{10}$, $M = 2^8$, $S_{\max} = 8{,}000$, $S_{\min} = 1$, $r = 0.02$, and $T = 44/234$, and scaling the data by dividing by 1,000. We use the particle swarm optimization (PSO) algorithm as the objective function for calibration. The PSO algorithm is a type of evolutionary algorithm with efficient performance, and the algorithm has convergence which can ensure that the algorithm solves the global optimal solution.

In Figure 1, the calibrated parameter of the BS model is 0.1957, which is often referred to as the implied volatility, and the parameters for the two-state KoBoL model are $Q = [-0.708, 0.708;\ 0.010, -0.010]$, $\alpha = [1.379, 1.944]$, $p = [0.001, 0.003]$, $\sigma = [0.800, 0.023]$, and $\lambda = [100.000, 99.635]$. The ranges of these parameters are $\alpha \in [1.101, 1.99]$, $q_{j,k} \in [0.01, 60]$, $\lambda \in [1.01, 100]$, $\sigma \in [0.01, 0.8]$, and $p \in [0.001, 0.999]$.



(a) Option value and market data

(b) Absolute value of relative errors

Figure 1: Comparison between two models.

Table 4: Iteration numbers and CPU time of three methods for Example 3

N	M	IM-nP		IM-P		IM-b-P	
		Ite	CPU	Ite	CPU	Ite	CPU
2^8	2^4	183.2	0.30 s	6.0	0.03 s	5.0	0.02 s
2^9	2^5	32.7	0.20 s	6.0	0.05 s	5.0	0.03 s
2^{10}	2^6	15.2	0.28 s	5.0	0.13 s	5.0	0.10 s
2^{11}	2^7	11.2	0.74 s	5.8	0.44 s	5.0	0.43 s

From Figure 1(a), the option prices of the two-state KoBoL model are closer to the real market price. And from Figure 1(b), the absolute value of the relative error of the two-state KoBoL model is smaller than that of the BS model, although there are some sudden movements in the real-world market. It is clear that the two-state KoBoL model is better, and many fundamental empirical facts of financial markets, such as skewed and unexpected huge swings in stock prices, are explained by it.

In Table 4, the GMRES methods with two preconditioners and three different methods are compared with these parameters. It is clear that although the GMRES method using the Strang preconditioner P_m (21) has improved a lot of iterations, the GMRES method using the proposed preconditioner P (10) with the Sherman-Morrison formula is still the best in terms of both iterations and CPU time in this case.

5 Conclusion

In this study, the regime-switching European option pricing based on the fractional diffusion model is studied. In order to reduce the calculation cost of solving the model, a second-order scheme of the implicit finite difference method is used, the analysis of related stability and convergence is given, and a special structure coefficient matrix is provided. The proposed preconditioner of the generalized minimal residual method solves the linear system $Mx = b$, which is accelerated by FFT. And then the Sherman-Morrison formula and ILU factorization are used to improve the calculation efficiency. Besides, the rationality of this preconditioner is also ensured by the theoretical analysis and finally proves the effectiveness of the proposed second-order scheme with the block preconditioner through numerical examples including an empirical example.

In addition, since the application of the Sherman-Morrison formula is limited by the number of states $\bar{J}$, the GMRES method using the proposed preconditioner P (10) with the Sherman-Morrison formula is chosen when $\bar{J}$ is 2 and the GMRES method with the preconditioner P (10) using ILU factorization is chosen when $\bar{J}$ is large.

Acknowledgement: The authors would like to express their sincere thanks to the referees for their valuable suggestions that led to an improved version of this manuscript.

Funding information: This work was supported in part by research grants of University of Macau (File Nos. MYRG2020-00208-FST and MYRG2022-00262-FST), Guangdong Basic and Applied Basic Foundation (Grant No. 2020A1515110991), the National Natural Science Foundation of China (Grant No. 12101137), and the Science and Technology Development Fund, Macau SAR under Funding Scheme for Postdoctoral Researchers of Higher Education Institutions 2021 (File No. 0032/APD/2021).

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and approved its submission.

Conflict of interest: The authors state no conflict of interest.

Data availability statement: All data generated or analysed during this study are included in this published article.

References

[1] F. Black and M. Scholes, *The pricing of options and corporate liabilities*, J. Pol. Econ. **81** (1973), no. 3, 637–654, DOI: https://doi.org/10.1086/260062.

[2] R. C. Merton, *Theory of rational option pricing*, Bell J. Econ. Manage. Sci. **4** (1973), 141–183, DOI: https://doi.org/10.2307/3003143.

[3] S. G. Kou, *A jump-diffusion model for option pricing*, Manage. Sci. **48** (2002), no. 8, 955–1101, DOI: https://doi.org/10.1287/mnsc.48.8.1086.166.

[4] P. Carr and L. Wu, *The finite moment log stable process and option pricing*, J. Finance **58** (2003), no. 2, 753–777, DOI: https://doi.org/10.1111/1540-6261.00544.

[5] P. Carr, H. Geman, D. B. Madan, and M. Yor, *The fine structure of asset returns: An empirical investigation*, J. Bus. **75** (2002), no. 2, 305–332, DOI: https://doi.org/10.1086/338705.

[6] S. I. Boyarchenko and S. Z. Levendorskiĭ, Non-Gaussian Merton-Black-Scholes Theory, Statistical Science & Applied Probability, Vol. 9, World Scientific, Singapore, 2002.

[7] S. L. Heston, *A closed-form solution for options with stochastic volatility with applications to bond and currency options*, Rev. Financ. Stud. **6** (1993), no. 2, 327–343, DOI: https://doi.org/10.1093/rfs/6.2.327.

[8] R. J. Elliott and C. J. U. Osakwe, *Option pricing for pure jump processes with Markov switching compensators*, Finance Stoch. **10** (2006), no. 2, 250–275, DOI: https://doi.org/10.1007/s00780-006-0004-6.

[9] P. Asiimwe, C. W. Mahera, and O. Menoukeu-Pamen, *On the price of risk under a regime switching CGMY process*, Asia-Pac. Financ. Mark. **23** (2016), 305–335, DOI: https://doi.org/10.1007/s10690-016-9219-5.

[10] D. Hainaut and O. Le Courtois, *An intensity model for credit risk with switching Lévy processes*, Quant. Finance **14** (2014), no. 8, 1453–1465, DOI: https://doi.org/10.1080/14697688.2012.756583.

[11] G. Deelstra and M. Simon, *Multivariate European option pricing in a Markov-modulated Lévy framework*, J. Comput. Appl. Math. **317** (2017), 171–187, DOI: https://doi.org/10.1016/j.cam.2016.11.040.

[12] S. Lei, W. Wang, X. Chen, and D. Ding, *A fast preconditioned penalty method for American options pricing under regime-switching tempered fractional diffusion models*, J. Sci. Comput. **75** (2018), no. 3, 1633–1655, DOI: https://doi.org/10.1007/s10915-017-0602-9.

[13] X. Li, P. Lin, X. C. Tai, and J. Zhou, *Pricing two-asset options under exponential Lévy model using a finite element method*, arXiv:1511.04950v1, 2015, https://doi.org/10.48550/arXiv.1511.04950.

[14] X. He and S. Lin, *A fractional Black-Scholes model with stochastic volatility and European option pricing*, Expert Syst. Appl. **178** (2021), 114983, DOI: https://doi.org/10.1016/j.eswa.2021.114983.

[15] X. Chen, D. Ding, S. Lei, and W. Wang, *An implicit-explicit preconditioned direct method for pricing options under regime-switching tempered fractional partial differential models*, Numer. Algorithms **87** (2021), no. 3, 939–965, DOI: https://doi.org/10.1007/s11075-020-00994-7.

[16] S. Lin and X. J. He, *A regime switching fractional Black-Scholes model and European option pricing*, Commun. Nonlinear Sci. Numer. Simul. **85** (2020), 105222, DOI: https://doi.org/10.1016/j.cnsns.2020.105222.

[17] A. Cartea and D. del Castillo-Negrete, *Fractional diffusion models of option prices in markets with jumps*, Phys. A **374** (2007), no. 2, 749–763, DOI: https://doi.org/10.1016/j.physa.2006.08.071.

[18] Z. Zhou, J. Ma, and X. Gao, *Convergence of iterative Laplace transform methods for a system of fractional PDEs and PIDEs arising in option pricing*, East Asian J. Appl. Math. **8** (2018), no. 4, 782–808, DOI: https://doi.org/10.4208/eajam.130218.290618.

[19] W. Chen, X. Xu, and S. P. Zhu, *Analytically pricing European-style options under the modified Black-Scholes equation with a spatial-fractional derivative*, Quart. Appl. Math. **72** (2014), no. 3, 597–611, DOI: https://doi.org/10.1090/S0033-569X-2014-01373-2.

[20] W. Chen and S. Wang, *A penalty method for a fractional order parabolic variational inequality governing American put option valuation*, Comput. Math. Appl. **67** (2014), no. 1, 77–90, DOI: https://doi.org/10.1016/j.camwa.2013.10.007.

[21] X. Guo, Y. Li, and H. Wang, *Tempered fractional diffusion equations for pricing multi-asset options under CGMYe process*, Comput. Math. Appl. **76** (2018), no. 6, 1500–1514, DOI: https://doi.org/10.1016/j.camwa.2018.07.002.

[22] M. Chen and W. Deng, *High order algorithms for the fractional substantial diffusion equation with truncated Lévy flights*, SIAM J. Sci. Comput. **37** (2015), no. 2, A890–A917, DOI: https://doi.org/10.1137/14097207X.

[23] R. H. Chan and X. Jin, *An introduction to iterative Toeplitz solvers*, Fundamentals of Algorithms, Vol. 5, SIAM, Philadelphia, 2007.

[24] S. L. Lei, D. Fan, and X. Chen, *Fast solution algorithms for exponentially tempered fractional diffusion equations*, Numer. Methods Partial Differential Equations **34** (2018), no. 4, 1301–1323, DOI: https://doi.org/10.1002/num.22259.

[25] M. S. Bartlett, *An inverse matrix adjustment arising in discriminant analysis*, Ann. Math. Statist. **22** (1951), 107–111, DOI: https://doi.org/10.1214/aoms/1177729698.