Fast exponential time integration scheme for option pricing with jumps

Spike T. Lee, Xin Liu, and Hai-Wei Sun*,†

Department of Mathematics, University of Macau, Macao, China

SUMMARY

A fast exponential time integration scheme is considered for pricing European and double barrier options in jump-diffusion models. After spatial discretization, the option pricing problem is transformed into the product of a matrix exponential and a vector, while the matrix bears a Toeplitz structure. The shift-and-invert Arnoldi method is then employed for fast approximations to such operation. Owing to the Toeplitz structure, the computational cost can be reduced by the fast Fourier transform. Furthermore, the discretized form of option pricing problem satisfies a given condition such that the error bound of the shift-and-invert Arnoldi approximation is unrelated to the norm of the matrix. Numerical tests are carried out to compare the proposed method with other option pricing methods.

KEY WORDS: shift-and-invert Arnoldi method; Toeplitz matrix exponential; generating function; option pricing; jump-diffusion

1. Introduction

In 1973, Black and Scholes [1] proposed a famous formula for pricing options under the pure-diffusion model. Years later, jump-diffusion models were proposed in order to overcome the shortcomings in the Black-Scholes model. For instance, Merton [2] proposed a jump-diffusion model with lognormally distributed jumps, while Kou [3] suggested one with double exponentially distributed jumps. In jump-diffusion models, one of the numerical methods for finding option prices is related to solving a partial integro-differential equation (PIDE). However, most existing methods employ straightforward second-order accurate schemes in spatial direction and time-stepping schemes in time direction. In [4], d'Halluin et al. applied the

*Correspondence to: Department of Mathematics, University of Macau, Macao, China.
†E-mail: HSun@umac.mo

Contract/grant sponsor: FDCT of Macao; contract/grant number: 033/2009/A

Contract/grant sponsor: University of Macau; contract/grant number: UL020/08-Y3/MAT/JXQ01/FST

Contract/grant sponsor: University of Macau; contract/grant number: RG057/09-10S/SHW/FST

second-order accurate Crank-Nicolson scheme plus Rannacher time-stepping to the PIDE with a fixed point iterative procedure as the system solver. Almendral and Oosterlee [5] exploited the second-order backward differentiation formula with regular splitting of matrices for solving resulting systems. Recently, Feng and Linetsky proposed to use the extrapolation approach in combination with implicit-explicit Euler (IMEX-Euler) scheme [6]. Numerical results in [6] show that the extrapolated IMEX-Euler scheme significantly enhances the classical IMEX-Euler scheme and shows strong competence in short-maturity cases. For high-order schemes, Lee and Sun [7] proposed to use the fourth-order compact boundary value method to achieve fourth-order accuracy in both space and time.

It is a common practice to use time-stepping schemes for temporal integration of PIDE. Lately, Tangman et al. [8] applied an exponential time integration (ETI) scheme to solve the PIDE. By the ETI scheme, the time direction of PIDE is directly tackled by a "one step" formula, which means discretization in time is not needed. Moreover, Tangman et al. [8] used the spectral method based on Chebyshev nodes for spatial discretization, which also guarantees a high-order convergence rate. In [8], numerical results show that the ETI scheme with spectral method is exact in time and approximately achieves fourth-order accuracy when pricing different options. Nevertheless, the coefficient matrix of the spectral method is dense and yields no specific structure. Since the ETI scheme in [8] needs to compute a matrix exponential, such action will lead to large computational cost of $\mathcal{O}(n^3)$ [9], where n is the size of the matrix.

In reality, the ETI scheme involves a matrix exponential multiplied by a vector, which has been well studied within the framework of Krylov subspace methods [10, 11]. Recently, Lee et al. [12] proposed a fast approach for computing the Toeplitz matrix exponential (TME), which in fact is the exponential of a Toeplitz matrix $[A_n]_{j,k} = a_{j-k}$ multiplied by a vector. The main idea is to use the shift-and-invert Arnoldi method in [11] and avoid the direct computation of the matrix exponential. Thanks to the Toeplitz structure, one can reduce the computational cost to $\mathcal{O}(n \log n)$ by the noted Gohberg-Semencul formula (GSF) [13] and fast Fourier transform (FFT). Lee et al. [12] have also showed that the iteration number of the shift-and-invert Arnoldi method does not arbitrarily increase when the generating function of the Toeplitz matrix satisfies two conditions. It inspires us to use the shift-and-invert Arnoldi method together with the ETI scheme for option pricing.

In this paper, we aim at pricing European and double barrier call options in Merton's and Kou's jump-diffusion models. To maintain the Toeplitz structure of the semi-discretized ODE system, we choose the second-order central difference scheme for spatial discretization. The Richardson extrapolation strategy is utilized to accelerate the spatial convergence. For temporal integration, we use the ETI scheme in [8] and then employ the shift-and-invert Arnoldi method in [12]. However, the convergence analysis in [12] is not directly applicable in our case. Therefore, we propose a criterion different from [12] to guarantee the efficiency of the shift-and-invert Arnoldi method, and prove that such criterion is satisfied in the option pricing problem. Numerical results show that the proposed scheme is robust and effective, when compared with other PIDE-based option pricing methods. This paper is organized as follows. In Section 2, we demonstrate the PIDE in jump-diffusion models and perform the spatial discretization. In Section 3, we introduce the shift-and-invert Arnoldi method and present an algorithm for fast computation of TME. Convergence analysis of the shift-and-invert Arnoldi method in the option pricing problem is discussed in Section 4. Numerical comparisons between various option pricing methods are given in Section 5. Finally the paper is concluded in Section 6.

2. Formulation and discretization of PIDE

2.1. PIDE in jump-diffusion models

Options are a type of financial derivatives written on the stock price S. For PIDE-based methods, it is natural to use the logarithmic price $x = \log(S/K)$, where K is known as the strike price. In jump-diffusion models, the option price $u(x,t)$ satisfies the following PIDE [4, 5, 6, 8]:

$$u_t = \frac{\sigma^2}{2} u_{xx} + \left(r - \lambda\kappa - \frac{\sigma^2}{2} \right) u_x - (r+\lambda)u + \lambda \int_{-\infty}^{\infty} u(x+y,t)\psi(y)dy, \quad (1)$$

where $x \in \mathbb{R} = (-\infty, \infty)$, $t \in [0,T]$ is the time until maturity T, $\sigma > 0$ is the stock return volatility, $r \geq 0$ is the risk-free interest rate, $\lambda > 0$ is the intensity of the Poisson process, κ is the expected jump size, and $\psi(y)$ is the jump magnitude probability density.

Different options determine different boundary conditions and initial conditions in (1). In this paper, we are interested in European and double barrier call options, which share the same initial condition

$$u(x,0) = \max\{Ke^x - K, 0\}. \quad (2)$$

For European call options, the boundary conditions are

$$u(x,t) \approx \begin{cases} 0, & \text{as } x \to -\infty, \\ Ke^x - Ke^{-rt}, & \text{as } x \to +\infty. \end{cases} \quad (3)$$

For double barrier options, the boundary conditions are

$$u(x_d,t) = u(x_u,t) = 0, \quad (4)$$

where x_d is the lower barrier, x_u is the upper barrier, and the option price vanishes outside $[x_d, x_u]$. Note that the put option case can be formulated in a similar way. Our job is to find the option value $u(x,t)$ at maturity time $t = T$.

In this work, we are interested in Merton's [2] and Kou's [3] jump-diffusion models. In Merton's model, the density function is a normal distribution function with mean μ and standard deviation σ_J:

$$\psi(y) = \frac{1}{\sqrt{2\pi}\sigma_J} e^{-\frac{(y-\mu)^2}{2\sigma_J^2}},$$

while the parameter κ in (1) is

$$\kappa = e^{\mu + \frac{\sigma_J^2}{2}} - 1.$$

In Kou's model, the density function is a double exponential function

$$\psi(y) = \begin{cases} w\alpha_1 e^{-\alpha_1 y}, & \text{if } y \geq 0, \\ (1-w)\alpha_2 e^{\alpha_2 y}, & \text{if } y < 0, \end{cases}$$

where $0 < w < 1$, α_1 and α_2 are real numbers such that $\alpha_1 > 1$ and $\alpha_2 > 0$, and the parameter κ has the form

$$\kappa = w\frac{\alpha_1}{\alpha_1 - 1} + (1-w)\frac{\alpha_2}{\alpha_2 + 1} - 1.$$

Note that in either model $\psi(y) \geq 0$ is a probability density function and hence

$$\int_{-\infty}^{\infty} \psi(y)dy = 1. \tag{5}$$

2.2. Discretization of PIDE

In this part, we will show how to discretize the PIDE (1) and transform it into a matrix exponential problem. Since the state variable x is unbounded in (1), we first truncate the infinite x-domain $\mathbb{R}$ to a finite domain $[x_{\min}, x_{\max}]$ [4]. In particular, it is a natural choice to assume $[x_{\min}, x_{\max}] = [x_d, x_u]$ in the double barrier case. Once the spatial domain is truncated, we consider a computational grid $\Omega_{\Delta x}$ defined by

$$\Omega_{\Delta x} = \left\{ x_j : x_j = x_{\min} + j\Delta x, j = 1, 2, \ldots, n,\ \Delta x = \frac{x_{\max} - x_{\min}}{n+1} \right\}.$$

We approximate the spatial derivatives in (1) by the second-order central difference scheme. Let $u_j(t)$ denote the approximation to $u(x_j, t)$ for $j = 1, 2, \ldots, n$. For simplicity, we let

$$c_1 = \sigma^2 \quad \text{and} \quad c_2 = r - \lambda\kappa - \frac{\sigma^2}{2}.$$

Then the differential operator of (1) is an $n \times n$ tridiagonal Toeplitz matrix D_n of the form

$$D_n = \text{tridiag}\left(\frac{c_1}{2\Delta x^2} - \frac{c_2}{2\Delta x},\ -\frac{c_1}{\Delta x^2} - r - \lambda,\ \frac{c_1}{2\Delta x^2} + \frac{c_2}{2\Delta x} \right).$$

For the integral term, we first have to divide it into two parts [5], and then evaluate the integral at each inner grid point x_j for $j = 1, 2, \ldots, n$:

$$\begin{aligned} \int_{-\infty}^{\infty} u(x_j + y, t)\psi(y)dy &= \int_{-\infty}^{\infty} u(y, t)\psi(y - x_j)dy \\ &= \int_{x_{\min}}^{x_{\max}} u(y, t)\psi(y - x_j)dy + \int_{\mathbb{R}\setminus[x_{\min}, x_{\max}]} u(y, t)\psi(y - x_j)dy. \end{aligned}$$

For the first integral over $[x_{\min}, x_{\max}]$, we use the second-order composite trapezoidal rule with step size Δx and obtain

$$\begin{aligned} \int_{x_{\min}}^{x_{\max}} u(y, t)\psi(y - x_j)dy &= \frac{\Delta x}{2} \left[u(x_{\min}, t)\psi(x_{\min} - x_j) + u(x_{\max}, t)\psi(x_{\max} - x_j) \right] \\ &\quad + \Delta x \left[\sum_{k=1}^{n} u(x_k, t)\psi(x_k - x_j) \right] + \mathcal{O}(\Delta x^2). \end{aligned} \tag{6}$$

Note that $u(x_{\min}, t)$ and $u(x_{\max}, t)$ are replaced by the boundary conditions in (3) for European call options or equal to zero in (4) for double barrier options. Let J_n denote the corresponding integral operator, which is an $n \times n$ Toeplitz matrix with entries

$$[J_n]_{j,k} = \Delta x \cdot \psi((k - j)\Delta x).$$

In Kou's model, we need to revise the value $\psi(0)$ as $\psi(0) = \dfrac{w\alpha_1 + (1 - w)\alpha_2}{2}$ because the integral (6) can be split at $y = x_j$. The two separated integrals are approximated by the composite trapezoidal rule respectively, which results in an intermediate value of $\psi(0)$.

The second integral over $\mathbb{R}\backslash[x_{\min}, x_{\max}]$ naturally equals to zero for double barrier options according to (4), so we just consider the European call options. From the boundary conditions given in (3), we can explicitly calculate the outer integral in both Merton's and Kou's models. In Merton's model, the outer integral has the form

$$\int_{x_{\max}}^{\infty} u(y,t)\psi(y-x_j)dy = Ke^{x_j+\mu+\frac{\sigma_J^2}{2}}\mathcal{N}\left(\frac{x_j - x_{\max} + \mu + \sigma_J^2}{\sigma_J}\right) - Ke^{-rt}\mathcal{N}\left(\frac{x_j - x_{\max} + \mu}{\sigma_J}\right),$$

where $\mathcal{N}$ is the standard normal cumulative distribution function. In Kou's model, direct calculations give

$$\int_{x_{\max}}^{\infty} u(y,t)\psi(y-x_j)dy = Kw\alpha_1 e^{\alpha_1(x_j - x_{\max})}\left(\frac{e^{x_{\max}}}{\alpha_1 - 1} - \frac{e^{-rt}}{\alpha_1}\right).$$

Note that in either case, the integral has the form

$$\int_{x_{\max}}^{\infty} u(y,t)\psi(y-x_j)dy = \iota_1(x_j) - \iota_2(x_j)e^{-rt},$$

where ι_1 and ι_2 are explicit functions of x_j and independent of time t.

After collecting all the terms together, we obtain the semi-discretized form of (1), which is an ODE system

$$\widehat{u}'(t) = -A_n\widehat{u}(t) + \zeta(t), \quad 0 \le t \le T, \tag{7}$$

where

$$A_n = -D_n - \lambda J_n$$

is an $n \times n$ real nonsymmetric Toeplitz matrix,

$$\widehat{u}(t) = [u_1(t), u_2(t), \ldots, u_n(t)]^\intercal,$$

and similar to [8], $\zeta(t)$ equals to zero for double barrier options, or has the following form for European call options:

$$\zeta(t) = \epsilon_1 - \epsilon_2 e^{-rt}, \tag{8}$$

where $\epsilon_1, \epsilon_2 \in \mathbb{R}^n$ have determined entries independent of time t as follows:

$$\epsilon_1 = \left(\frac{c_1}{2\Delta x^2} + \frac{c_2}{2\Delta x}\right)\begin{bmatrix} 0 \\ 0 \\ \vdots \\ Ke^{x_{\max}} \end{bmatrix} + \lambda \begin{bmatrix} \iota_1(x_1) \\ \iota_1(x_2) \\ \vdots \\ \iota_1(x_n) \end{bmatrix} + \frac{\lambda \Delta x K e^{x_{\max}}}{2}\begin{bmatrix} \psi(x_{\max} - x_1) \\ \psi(x_{\max} - x_2) \\ \vdots \\ \psi(x_{\max} - x_n) \end{bmatrix},$$

and

$$\epsilon_2 = \left(\frac{c_1}{2\Delta x^2} + \frac{c_2}{2\Delta x}\right)\begin{bmatrix} 0 \\ 0 \\ \vdots \\ K \end{bmatrix} + \lambda \begin{bmatrix} \iota_2(x_1) \\ \iota_2(x_2) \\ \vdots \\ \iota_2(x_n) \end{bmatrix} + \frac{\lambda \Delta x K}{2}\begin{bmatrix} \psi(x_{\max} - x_1) \\ \psi(x_{\max} - x_2) \\ \vdots \\ \psi(x_{\max} - x_n) \end{bmatrix}.$$

2.3. Integrating the ODE system

For the semi-discretized ODE system (7), one can choose different ODE solvers for temporal integration, like the popular time-stepping methods. The number of timesteps in time-stepping methods decides how many systems await to be solved. Alternatively, we can simply integrate (7) from time 0 to T to obtain the option price at $t = T$. This approach, which is known as the ETI scheme, is considered in [8] for option pricing. In this regard, the option price at maturity time T becomes

$$\widehat{u}(T) = e^{-TA_n}\widehat{u}(0) + e^{-TA_n}\int_0^T e^{tA_n}\zeta(t)dt. \tag{9}$$

Note that the option value at $t = 0$ is the payoff, therefore $\widehat{u}(0) \in \mathbb{R}^n$ is a known vector containing the discrete values of payoff function (2). Tangman et al. [8] proposed a lemma to explicitly calculate the integral in (9) owing to the special formulation of the nonhomogeneous term (8) in option pricing.

Lemma 1. (see [8], Lemma 1) *The integral term in* (9) *can be calculated by*

$$e^{-TA_n}\int_0^T e^{tA_n}\zeta(t)dt = -A_n^{-1}\left(e^{-TA_n} - I_n\right)\epsilon_1 - (rI_n - A_n)^{-1}\left(e^{-TA_n} - e^{-rT}I_n\right)\epsilon_2,$$

where I_n is an identity matrix of size n.

To further simplify the expression in Lemma 1, we note that the matrices e^{-TA_n} and A_n^{-1} commute, so do e^{-TA_n} and $(rI_n - A_n)^{-1}$. Using Lemma 1 in (9) and carrying out some simple calculations, we have the final formula for option pricing:

$$\begin{aligned}\widehat{u}(T) &= e^{-TA_n}\left[\widehat{u}(0) - A_n^{-1}\epsilon_1 - (rI_n - A_n)^{-1}\epsilon_2\right]\\ &\quad + \left[A_n^{-1}\epsilon_1 + e^{-rT}(rI_n - A_n)^{-1}\epsilon_2\right]\\ &\equiv e^{-TA_n}\eta_1 + \eta_2,\end{aligned} \tag{10}$$

where η_1 and η_2 denote the vectors

$$\eta_1 = \widehat{u}(0) - A_n^{-1}\epsilon_1 - (rI_n - A_n)^{-1}\epsilon_2 \quad \text{and} \quad \eta_2 = A_n^{-1}\epsilon_1 + e^{-rT}(rI_n - A_n)^{-1}\epsilon_2.$$

Note that A_n is a Toeplitz matrix, and so is $rI_n - A_n$. Therefore, the terms $A_n^{-1}\epsilon_1$ and $(rI_n - A_n)^{-1}\epsilon_2$ can be derived by fast Toeplitz system solvers with preconditioning techniques; see [14, 15] for details. Also the two vectors η_1 and η_2 are only needed once and they are not a major concern.

By formula (10), we mainly need to find the matrix exponential e^{-TA_n} to obtain the option price $\widehat{u}(T)$. This approach is totally unlike traditional time-stepping methods because no discretization in time is required. The ETI method in [8] chooses to compute the matrix exponential e^{-TA_n} directly. But in fact, the first term in (10) is the TME in our case. Therefore, our goal can be summarized as approximating the TME of the form

$$w(t) = e^{-tA_n}v, \tag{11}$$

where A_n is a real Toeplitz matrix, v is a real vector, and $t > 0$ is a scaling factor, which represents the time in option pricing.

Remark: We should point out that owing to the use of central difference discretization, the formula (10) is only supposed to reach second-order convergence in space. To match up the exact computation in time direction, we try to accelerate the spatial convergence by using the extrapolation method. Extrapolation methods are widely used to improve the convergence properties of many processes. In our numerical experiment, we implement the famous Richardson extrapolation; see [16] for its formula and details.

3. Shift-and-invert Arnoldi method for computing TME

3.1. Shift-and-invert Arnoldi method

Numerical methods for matrix exponential include the Padé approximation [9], or the popular scaling and squaring method [17]. These methods concentrate more on the approximation to the whole matrix exponential and require computational cost of $\mathcal{O}(n^3)$ for a general $n \times n$ dense matrix. In [18], Saad proposed to use the Krylov subspace method for computing the matrix exponential multiplied by a vector. The concept is to approximate the exponential function by a polynomial in Π_{m-1}, which contains all algebraic polynomials of degree less than $m-1$. In this sense, the exponential of an $n \times n$ matrix A_n is approximated by

$$e^{A_n}v \approx p_{m-1}(A_n)v, \quad p_{m-1} \in \Pi_{m-1}.$$

Note that this polynomial approximation belongs to an m-dimensional Krylov subspace generated by A_n and v:

$$\mathcal{K}_m \equiv \text{span}\left\{v, A_n v, A_n^2 v, \ldots, A_n^{m-1} v\right\}.$$

The next step is to construct a basis of $\mathcal{K}_m$ with either the Lanczos process or the Arnoldi process. Meanwhile, one will encounter repeated actions of a matrix-vector product at each iteration. After the iterative process, the vectors and recurrence coefficients are employed to approximate $e^{A_n}v$, or furthermore $e^{-tA_n}v$; see [18] for details. However, the decisive factor of Krylov subspace method lies in m, the size of the Krylov subspace. If m is not reasonably small compared to the matrix size n, this method will be regarded as unacceptable. Unfortunately, past analysis [10] showed that the iterative number m could get quite large when $\|A_n\|_2$ increases.

To improve the standard Krylov subspace method, one has to reduce the size of m and also detach m from $\|A_n\|_2$. Moret and Novati [11] proposed to use the shift-and-invert technique to speed up the convergence of Krylov subspace method by emphasizing the eigenvalues of greater importance. Instead of using A_n as the generating matrix of the Krylov subspace, the shifted-and-inverted matrix

$$Z_n \equiv (I_n + \gamma A_n)^{-1}, \tag{12}$$

with a shift parameter $\gamma > 0$, is chosen as the generating matrix and hence the Krylov subspace switches to

$$\mathcal{K}_m \equiv \text{span}\left\{v, Z_n v, Z_n^2 v, \ldots, Z_n^{m-1} v\right\}.$$

One way to find the orthonormal basis of $\mathcal{K}_m$ is by the Arnoldi process. Since now the generating matrix of $\mathcal{K}_m$ becomes Z_n, the Arnoldi process turns into the following algorithm.

Algorithm 1. (*Shift-and-invert Arnoldi process*)

1. *Initialize*: *Compute* $v_1 = \dfrac{v}{\|v\|_2}$
2. *Iterate*: *Do* $k = 1, \ldots, m$
   I. Compute $\widehat{v} := Z_n v_k = (I_n + \gamma A_n)^{-1} v_k$
   II. Do $j = 1, \ldots, k$
   IIa. Compute $h_{j,k} := \widehat{v}^{\intercal} v_j$
   IIb. Compute $\widehat{v} := \widehat{v} - h_{j,k} v_j$
   III. Compute $h_{k+1,k} := \|\widehat{v}\|_2$ *and* $v_{k+1} := \dfrac{\widehat{v}}{h_{k+1,k}}$

Let $\{v_1, v_2, \ldots, v_m\}$ be the formed orthonormal basis of $\mathcal{K}_m$ by Algorithm 1. These vectors satisfy

$$Z_n V_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^{\intercal},$$

where $V_m = [v_1, v_2, \ldots, v_m]$ is an $n \times m$ matrix , e_m is the m-th vector of the canonical basis of $\mathbb{R}^m$, and H_m is an $m \times m$ upper Hessenberg matrix having entries $[H_m]_{j,k} = h_{j,k}$. With all the recurrence vectors and coefficients in hand, we approximate $e^{-tA_n} v$ by [11, 12]:

$$e^{-tA_n} v \approx \beta V_m e^{-t\widehat{H}_m} e_1, \quad \beta = \|v\|_2, \quad \widehat{H}_m = \frac{1}{\gamma}(H_m^{-1} - I_m).$$

Let $w_m(t)$ denote the shift-and-invert Arnoldi approximation

$$w_m(t) \equiv \beta V_m e^{-t\widehat{H}_m} e_1, \tag{13}$$

which literally is a smaller matrix exponential of size m. The shift-and-invert technique lays more stress on the significant eigenvalues and reduces the size of the Krylov subspace. With m satisfactorily small, one can apply all kinds of existing methods [9] to the Hessenberg matrix exponential in (13). This treatment for computing $e^{-tA_n} v$ is called the shift-and-invert Arnoldi method [11, 12]. Next we consider how to implement the method when approximating the TME (11) with A_n being a Toeplitz matrix.

3.2. Implementation for TME

The implementation of the shift-and-invert Arnoldi method mainly concerns repeated actions of

$$Z_n v_k = (I_n + \gamma A_n)^{-1} v_k$$

in each iteration of Algorithm 1. For a general dense matrix A_n, the computation of the inverse brings about computational cost of $\mathcal{O}(n^3)$. For a Toeplitz matrix A_n, the situation will be much better if we manipulate a classic trick related to Toeplitz matrices.

In the computation of $Z_n v_k$, the matrix is fixed in each iteration and only the vector varies. This fact is realized in [11, 12] and generally one thinks of finding the inverse matrix Z_n first before heading into the iterative procedure. For instance, the LU decomposition is an approved choice for a general matrix [11]. In [12], the GSF is used in the implementation of the shift-and-invert Arnoldi method for TME. The GSF is about the inverse of a Toeplitz matrix, and its usage includes solving repeated Toeplitz systems quickly.

The GSF tells us that for any Toeplitz matrix A_n, its inverse A_n^{-1} can be represented by the first and last columns of A_n^{-1}. Since $I_n + \gamma A_n$ is a Toeplitz matrix as well, Z_n can be represented by the first and last columns of Z_n. Suppose the first and last columns of Z_n are $p = [p_1, p_2, \ldots, p_n]^\intercal$ and $q = [q_1, q_2, \ldots, q_n]^\intercal$ respectively. Let

$$P_n = \begin{bmatrix} p_1 & 0 & \cdots & 0 \\ p_2 & p_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ p_n & p_{n-1} & \cdots & p_1 \end{bmatrix}, \quad Q_n = \begin{bmatrix} q_n & 0 & \cdots & 0 \\ q_{n-1} & q_n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ q_1 & q_2 & \cdots & q_n \end{bmatrix},$$

$$\widehat{P}_n = \begin{bmatrix} 0 & \cdots & 0 & 0 \\ p_n & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ p_2 & \cdots & p_n & 0 \end{bmatrix}, \quad \text{and} \quad \widehat{Q}_n = \begin{bmatrix} 0 & \cdots & 0 & 0 \\ q_1 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ q_{n-1} & \cdots & q_1 & 0 \end{bmatrix}.$$

Then Z_n has the following expression [13]

$$Z_n = \frac{1}{p_1}\left(P_n Q_n^\intercal - \widehat{Q}_n \widehat{P}_n^\intercal\right), \tag{14}$$

in which P_n, Q_n, $\widehat{P}_n$, and $\widehat{Q}_n$ are all lower triangular Toeplitz matrices. It is well known that Toeplitz matrix-vector products are completed by FFTs with $\mathcal{O}(n \log n)$ complexity [14]. Therefore, the computational cost of $Z_n v_k$ can be reduced to $\mathcal{O}(n \log n)$ by resorting to FFTs. In detail, it can be reduced to about six FFTs of size n in order to obtain $Z_n v_k$; see [12] for technical details.

To apply the GSF, we first need to find p and q, the first and last columns of Z_n. This is accomplished by solving two Toeplitz systems

$$(I_n + \gamma A_n)p = e_1 \quad \text{and} \quad (I_n + \gamma A_n)q = e_n.$$

By using iterative Toeplitz solvers with preconditioning techniques [14, 15], these two Toeplitz systems can be solved efficiently and rapidly with $\mathcal{O}(n \log n)$ complexity. Also we notice that these two systems are only solved once in the entire process. Finally we conclude the algorithm of the shift-and-invert Arnoldi method for TME [12]:

Algorithm 2. (*Shift-and-invert Arnoldi method for TME*)

1. *Use fast Toeplitz solvers to solve* $(I_n + \gamma A_n)p = e_1$ *and* $(I_n + \gamma A_n)q = e_n$
2. *Perform the shift-and-invert Arnoldi process* (*Algorithm* 1) *during which* $Z_n v_k$ *is calculated by six FFTs in each iteration*
3. *Obtain the recurrence terms and evaluate* $w_m(t) = \beta V_m e^{-t\widehat{H}_m} e_1$

4. Convergence analysis of shift-and-invert Arnoldi method

4.1. Error estimation of shift-and-invert Arnoldi method

In spite of faster convergence because of the shift-and-invert technique, we still need to face the problem whether the iteration number m is related to $\|A_n\|_2$. In [11, 12], it is shown theoretically and numerically that the shift-and-invert Arnoldi method also suffers from this plight unless the matrix A_n meets certain conditions. In [11], Moret and Novati carried out a detailed error estimate for the shift-and-invert Arnoldi method by the numerical range. Therefore, we first review several important concepts.

Definition 1. (see [19]) *The numerical range of a matrix A_n is defined as a subset in the complex plane $\mathbb{C}$:*

$$W(A_n) \equiv \{v^* A_n v,\ v \in \mathbb{C}^n,\ v^* v = 1\}.$$

Definition 2. (see [11]) *Define*

$$\Sigma_{\vartheta,\alpha} = \left\{ z : |\arg(z-\alpha)| < \vartheta,\ 0 < \vartheta < \frac{\pi}{2},\ \alpha \geq 0 \right\}.$$

Then an operator A_n is called a sectorial operator if

$$W(A_n) \subseteq \overline{\Sigma_{\vartheta,\alpha}}.$$

Furthermore, let $S_{\vartheta,\rho}$ denote a bounded sector with vertex being the origin

$$S_{\vartheta,\rho} = \{z : z \in \Sigma_{\vartheta,0},\ 0 < |z| \leq \rho\}.$$

In [11], a sufficient condition is provided for error estimate in terms of $W(Z_n)$, where Z_n is given in (12). According to [11, 12] and the discussions therein, we can summarize the theoretical result in [11] by the following proposition.

Proposition 1. (see [11], Proposition 2.1 and Proposition 3.2) *Let $W(Z_n) \subset \overline{S_{\vartheta,\rho}}$. Then the following error bound holds:*

$$\|w(t) - w_m(t)\|_2 \leq \left[\pi \sin\left(\frac{\pi}{4} - \frac{\vartheta}{2}\right)\right]^{-1} \Phi_m,$$

where $\Phi_m \to 0$ as $m \to \infty$ and its convergence is independent of $\|A_n\|_2$.

Proposition 1 states that $\|A_n\|_2$ is not involved in the convergence of the shift-and-invert Arnoldi approximation $w_m(t)$. Therefore, the iteration number m is not disturbed by $\|A_n\|_2$. The only thing left for us is to determine whether the sufficient condition is met, i.e., $W(Z_n)$ belongs to a bounded sector $\overline{S_{\vartheta,\rho}}$.

Note that Z_n in (12) is obtained from A_n through the scalar transformation $z = (1+\gamma a)^{-1}$ in a matrix sense. Suppose A_n is a sectorial operator

$$W(A_n) \subseteq \overline{\Sigma_{\vartheta,\alpha}}.$$

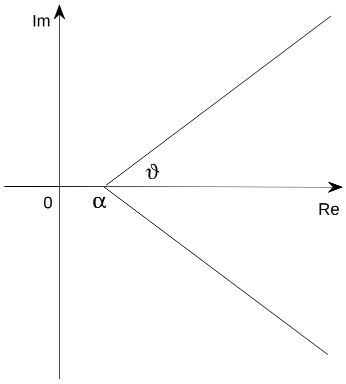
The transformation $z = (1+\gamma a)^{-1}$ creates a mapping such that

$$W(Z_n) \subseteq \overline{\Sigma_{\vartheta,0}} \cap D_{(1+\gamma\alpha)^{-1}/2} \subset \overline{S_{\vartheta,(1+\gamma\alpha)^{-1}}},$$

where $D_{(1+\gamma\alpha)^{-1}/2}$ is a disk of center and radius $(1+\gamma\alpha)^{-1}/2$. In such case, $W(Z_n)$ belongs to $\overline{S_{\vartheta,(1+\gamma\alpha)^{-1}}}$, which meets the condition of Proposition 1. Therefore, another sufficient condition for Proposition 1 is derived. If A_n is a sectorial operator, then Proposition 1 also holds true and the error bound does not depend on $\|A_n\|_2$; see the illustration of sectors in Figure 1 and [11] for more details.

4.2. Error estimation of TME

In this part, we will discuss when the TME satisfies the condition which makes m independent of $\|A_n\|_2$. According to Proposition 1 and its following discussions, it involves the idea of sectorial operator and numerical range. The good news is that there are two ways to make



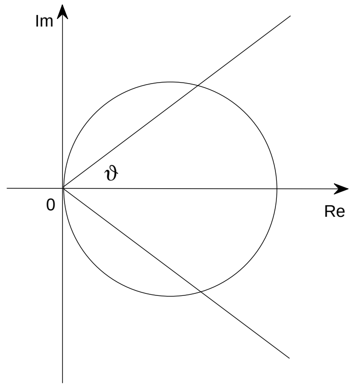


Figure 1. Left picture: the sector $\overline{\Sigma_{\vartheta,\alpha}}$; Right picture: the enclosed area is $\overline{\Sigma_{\vartheta,0}} \cap D_{(1+\gamma\alpha)^{-1}/2}$, which is further contained in $\overline{S_{\vartheta,(1+\gamma\alpha)^{-1}}}$.

Proposition 1 work. One is to make sure $W(Z_n)$ belongs to a bounded sector, and the other one is to show that A_n is a sectorial operator. Practically speaking, the numerical range is not easy to compute, not to mention using it to detect whether a matrix is a sectorial operator. In [12], Lee et al. considered the specific occasion when A_n is a Toeplitz matrix. They make use of the scarce relation between the numerical range and the generating function of Toeplitz matrices. Thus we first review the definition of generating function.

Definition 3. (see [14]) *Suppose the diagonals $\{a_k\}_{k=-n+1}^{n-1}$ of a Toeplitz matrix A_n are the Fourier coefficients of a function f:*

$$a_k = a_k(f) \equiv \frac{1}{2\pi}\int_{-\pi}^{\pi} f(\theta)e^{-ik\theta}d\theta, \quad i \equiv \sqrt{-1}, \quad k = -n+1, \ldots, n-1.$$

Then the function f is known as the generating function of A_n.

It is often assumed that a generating function $f \in \mathcal{C}_{2\pi}$, where $\mathcal{C}_{2\pi}$ contains all the 2π-periodic continuous complex-valued functions. As usual, we let $\mathrm{Re}(f)$ and $\mathrm{Im}(f)$ denote the real and imaginary parts of f respectively. Let

$$\mathrm{range}(f) \equiv \{f(\theta),\ \forall \theta \in [-\pi, \pi]\}$$

be the range of f. Moreover, we define a Toeplitz matrix generated by f as $\mathcal{T}_n[f]$, and

$$\|f\|_\infty = \max_{\theta\in[-\pi,\pi]} |f(\theta)|.$$

In [12], a sufficient condition is established to tell when a Toeplitz matrix is a sectorial operator.

Theorem 1. (see [12], Theorem 4.2) *Suppose $A_n = \mathcal{T}_n[f] \in \mathbb{R}^{n\times n}$ with $f \in \mathcal{C}_{2\pi}$. If f satisfies*

$$\mathrm{Re}(f) \geq 0 \quad and \quad \left\|\frac{\mathrm{Im}(f)}{\mathrm{Re}(f)}\right\|_\infty = \mathcal{O}(1), \tag{15}$$

then A_n is a sectorial operator.

We know from Theorem 1 that as long as the two assumptions in (15) are met, then the Toeplitz matrix generated by f is a sectorial operator, which indirectly satisfies the condition in Proposition 1. In the end, (15) becomes the sufficient condition for Toeplitz matrices to guarantee the independence between m and $\|A_n\|_2$ in the shift-and-invert Arnoldi method.

For our discretized problem, it is already proved in [12] that the generating function of A_n satisfies (15) when the interest rate r is strictly greater than zero. Unfortunately when r potentially equals to zero, there is a chance that the second condition in (15) is violated. Therefore, we have to think of another way to go around this obstacle. Now we no longer attempt to show that A_n is a sectorial operator, but instead go back to the original format to show that $W(Z_n)$ belongs to a bounded sector. The convex hull is needed in our proof and here we give its definition.

Definition 4. *Given a set U, then its convex hull* $\mathrm{conv}(U)$ *is the intersection of all the convex sets containing U.*

First we have to introduce a lemma about $W(I_n + \gamma A_n)$ when A_n is a Toeplitz matrix.

Lemma 2. *Suppose $A_n = \mathcal{T}_n[f] \in \mathbb{R}^{n\times n}$ with $f \in \mathcal{C}_{2\pi}$. If f satisfies*

$$\mathrm{Re}(f) \geq 0 \quad \textit{and} \quad \left\| \frac{\mathrm{Im}(f)}{\widehat{\gamma} + \mathrm{Re}(f)} \right\|_\infty = \mathcal{O}(1), \quad \widehat{\gamma} > 0, \tag{16}$$

then we have $W(I_n + \gamma A_n) \subseteq \overline{\Sigma_{\vartheta,\alpha}}$ for any $\gamma \leq \dfrac{1}{2\widehat{\gamma}}$ with $0 < \vartheta < \dfrac{\pi}{2}$ and $\alpha > 0$.

Proof: Note that $I_n + \gamma A_n$ is still a Toeplitz matrix and $A_n = \mathcal{T}_n[f]$, therefore its generating function is $1 + \gamma f$. From [20], or Theorem 2.3 in [12], there exists the following relation:

$$W(I_n + \gamma A_n) \subseteq \overline{\mathrm{conv}(\mathrm{range}(1 + \gamma f))}, \tag{17}$$

which links the numerical range of a Toeplitz matrix and its own generating function. The second half is to prove that the above convex hull is contained in a sector with positive vertex.

Considering $\gamma \leq \dfrac{1}{2\widehat{\gamma}}$, we let

$$\alpha = 1 - \gamma\widehat{\gamma} \geq \frac{1}{2} > 0 \quad \text{and} \quad 0 < \vartheta \equiv \arctan \left\| \frac{\mathrm{Im}(f)}{\widehat{\gamma} + \mathrm{Re}(f)} \right\|_\infty < \frac{\pi}{2}.$$

For any $z \in \mathrm{range}(1 + \gamma f)$, we use (16) and derive

$$|\arg(z - \alpha)| \leq \arctan \left\| \frac{\gamma \mathrm{Im}(f)}{\gamma\widehat{\gamma} + \gamma \mathrm{Re}(f)} \right\|_\infty = \arctan \left\| \frac{\mathrm{Im}(f)}{\widehat{\gamma} + \mathrm{Re}(f)} \right\|_\infty = \vartheta.$$

According to the definition of $\overline{\Sigma_{\vartheta,\alpha}}$, it means $z \in \overline{\Sigma_{\vartheta,\alpha}}$ and further implies that

$$\overline{\mathrm{conv}(\mathrm{range}(1 + \gamma f))} \subseteq \overline{\Sigma_{\vartheta,\alpha}}. \tag{18}$$

Combining (17) and (18), we obtain

$$W(I_n + \gamma A_n) \subseteq \overline{\Sigma_{\vartheta,\alpha}},$$

where $0 < \vartheta < \pi/2$ and $\alpha > 0$. The proof is completed. □

Lemma 2 tells us that $W(I_n + \gamma A_n)$ lies in a sector with positive vertex. This is different from standard sectorial operator because the vertex does not need to be strictly greater than zero.

Next we consider what will happen if we invert the matrix $I_n + \gamma A_n$. Suppose $W(A_n) \subseteq \overline{\Sigma_{\vartheta,\alpha}}$ for some $\alpha > 0$. We consider a mapping $z = a^{-1}$ in a scalar case, which has the following effect:

$$W(A_n^{-1}) \subseteq \overline{\Sigma_{\vartheta,0}} \cap D_{\alpha^{-1}/2} \subset \overline{S_{\vartheta,\alpha^{-1}}}.$$

We then write down the following lemma.

Lemma 3. *If $W(A_n) \subseteq \overline{\Sigma_{\vartheta,\alpha}}$ with $\alpha > 0$, then we have $W(A_n^{-1}) \subset \overline{S_{\vartheta,\alpha^{-1}}}$.*

Using Lemma 2 and Lemma 3, we can prove the following theorem.

Theorem 2. *Suppose $A_n = \mathcal{T}_n[f] \in \mathbb{R}^{n\times n}$ with $f \in \mathcal{C}_{2\pi}$. If f satisfies the two assumptions in* (16)*, then the error bound of the shift-and-invert Arnoldi method is independent of $\|A_n\|_2$.*

Proof: Since f satisfies the two conditions in (16) of Lemma 2, it leads to

$$W(I_n + \gamma A_n) \subseteq \overline{\Sigma_{\vartheta,\alpha}}$$

for some $\alpha > 0$. By Lemma 3, we have

$$W(Z_n) = W((I_n + \gamma A_n)^{-1}) \subset \overline{S_{\vartheta,\alpha^{-1}}},$$

which satisfies the condition in Proposition 1. Thus, by Proposition 1, we get that the error bound of the shift-and-invert Arnoldi method is independent of $\|A_n\|_2$. □

4.3. Theoretical analysis of the PIDE operator

Theorem 2 basically is the generalization of Theorem 1 [12], and efficiently fills up the hole when $\mathrm{Re}(f)$ is potentially equal to zero. The key point is to make sure the generating function f of A_n meets the revised assumptions in (16). Now we move on to see whether (16) is satisfied with $r \geq 0$ in our discretized problem. First we recall the assumptions on the parameters: $\sigma > 0$, $r \geq 0$, and $\lambda > 0$. Let f denote the corresponding generating function of $A_n = -D_n - \lambda J_n$, where both D_n and J_n are Toeplitz matrices. Recall that

$$c_1 = \sigma^2 > 0 \quad \text{and} \quad c_2 = r - \lambda\kappa - \frac{\sigma^2}{2}.$$

Analogous to [12], the generating function f of A_n is

$$f = \frac{c_1}{\Delta x^2}(1 - \cos\theta) - i\frac{c_2}{\Delta x}\sin\theta + r + \lambda - \lambda g,$$

where g denotes the generating function of the integral part J_n and satisfies the following properties

$$|\mathrm{Re}(g)| \leq 1 \quad \text{and} \quad |\mathrm{Im}(g)| \leq 1.$$

For any $\theta \in [-\pi, \pi]$, we have

$$\mathrm{Re}(f) = \frac{c_1}{\Delta x^2}(1 - \cos\theta) + r + \lambda[1 - \mathrm{Re}(g)] \geq 0.$$

According to Theorem 2, we let $\widehat{\gamma} > 0$ be a positive number, then

$$\begin{aligned} \left|\frac{\mathrm{Im}(f)}{\widehat{\gamma} + \mathrm{Re}(f)}\right| &= \frac{|-c_2\Delta x\sin\theta - \lambda\Delta x^2\mathrm{Im}(g)|}{c_1(1-\cos\theta) + (\widehat{\gamma}+r)\Delta x^2 + \lambda\Delta x^2[1-\mathrm{Re}(g)]} \\ &\leq \frac{c_2^2\sin^2\theta + (1+2\lambda)\Delta x^2}{4c_1\sin^2\frac{\theta}{2} + 2(\widehat{\gamma}+r)\Delta x^2} = \mathcal{O}(1), \end{aligned}$$

where $\widehat{\gamma}$ is independent of Δx. In conclusion, the generating function f of A_n meets (16), hence the error bound of the shift-and-invert Arnoldi approximation should be independent of $\|A_n\|_2$. As a result, we can neglect the side effects brought by $\|A_n\|_2$ when n gets large in numerical experiments.

5. Numerical results

In this section, we aim to solve the PIDE (1) by the option pricing formula (10) and the TME approximation (11). The shift-and-invert Arnoldi method is implemented to approximate the TME according to Algorithm 2; see more details in [12]. All the encountered Toeplitz systems are solved by the BICGSTAB iterative method [21] with tridiagonal preconditioner [15] with tolerance 10^{-12} and maximum iteration number 20. It is reported in [15] that tridiagonal preconditioners work well in the current option pricing models. The shift parameter in (12) is chosen as $\gamma = T/10$ similarly to [11, 12]. In our numerical tests, we set a maximum iteration number $m = \min\{n, 20\}$ in Algorithm 1 [12]. Results show that in most cases this m leads to a small enough error when approximating the TME, and hence does not affect the behavior of spatial convergence that we want to observe.

Let "TME" denote the proposed method in this paper, "ETI-Cheb" denote the ETI scheme with spectral method in [8], and "Extrap-IMEX" denote the extrapolated IMEX-Euler scheme [6]. The comparison between the three methods is achieved by measuring the CPU time spent to reach similar level of accuracy. It is easy to see that both the TME and ETI-Cheb methods are time-independent and a large T will not have a significant impact on them. The Extrap-IMEX method is time-dependent, therefore the computational cost is twice expensive when the maturity time T is doubled. Thus, we only have to consider a large maturity time $T = 2$ in the following experiments.

For the TME method, we have already proved that the iteration number m is not related to the matrix norm, which makes things easier when we refine the spatial grid. The ETI-Cheb method is already given in MATLAB code in [8], which is a convenience for comparison. For the Extrap-IMEX method, we will perform it on the semi-discretized system (7); see [6] for details. Since $T = 2$, the Extrap-IMEX method requires dividing the time domain $[0, 2]$ into 4 sections and then performing extrapolation strategy on every single one of them [6].

Both European and double barrier call options are considered in Merton's and Kou's jump-diffusion models. The computational domain $[-4, 4]$ is used for European call options. For double barrier call options, we set the upper barrier $x_u = 0.5$ and lower barrier $x_d = -1$, which approximately represents the stock price S ranging from $[K \cdot 0.36, K \cdot 1.64]$. The exact prices of European call options under Merton's model are computed through an analytic formula [2]. In this case, the approximate error can be obtained via the difference between true solution and approximation under the infinity norm. For other cases without analytic solution, we calculate the infinity norm error between any level and its following level as we refine the spatial grid. The columns "error" and "time" display the errors and CPU times (in seconds) respectively. The column "timesteps" only belongs to the Extrap-IMEX method, as it contains the total number of timesteps. The tags "Euro" and "DB" stand for European and double barrier call options respectively.

5.1. Merton's Model

In this experiment, we test a set of parameters which is used in [5]: $K = 1$, $\sigma = 0.2$, $r = 0$, $\lambda = 0.1$, $\mu = 0$, and $\sigma_J = 0.5$. In [8], the ETI-Cheb method is coupled with the spectral method for spatial discretization, and approximately achieves fourth-order convergence in space in Merton's model. Both the TME method and the Extrap-IMEX method are based on the second-order central difference scheme in space, but they are mended by the Richardson extrapolation [16].

Table I. Pricing European and double barrier call options in Merton's jump-diffusion model with $T = 2$.

	n	TME		ETI-Cheb		Extrap-IMEX		
		error	time	error	time	timesteps	error	time
Euro	31	5.7e-4	0.10	1.8e-4	0.01	100	5.6e-4	0.02
	63	2.6e-5	0.11	1.1e-5	0.02	144	2.6e-5	0.03
	127	1.4e-6	0.14	7.2e-7	0.07	196	1.4e-6	0.09
	255	8.5e-8	0.22	4.5e-8	0.48	292	8.6e-8	0.45
	511	4.6e-9	0.38	3.7e-8	3.87	400	9.6e-9	1.88
DB	14	2.0e-5	0.03	4.6e-5	0.01	124	2.0e-5	0.01
	29	1.2e-6	0.04	2.8e-6	0.01	196	1.2e-6	0.02
	59	7.5e-8	0.04	1.8e-7	0.02	256	7.5e-8	0.03
	119	4.7e-9	0.05	1.1e-8	0.06	324	5.0e-9	0.06
	239	2.9e-10	0.08	7.0e-10	0.41	400	2.4e-10	0.14

From the results in Table I, all three methods seem to get nearly fourth-order convergence rate when pricing both European and double barrier call options in Merton's model. For a large n, the ETI-Cheb method goes into trouble because the matrix exponential is computed directly with $\mathcal{O}(n^3)$ complexity. For a maturity time as large as $T = 2$, the Extrap-IMEX method is time-dependent and takes a longer time to finish. In conclusion, the ETI-Cheb and Extrap-IMEX methods have issues with either large n or large T.

5.2. Kou's Model

In Kou's jump-diffusion model, we take a set of parameters which is also used in [5]: $K = 1$, $\sigma = 0.2$, $r = 0$, $\lambda = 0.2$, $w = 0.5$, $\alpha_1 = 3$, and $\alpha_2 = 2$. The ETI-Cheb method is only considered for Merton's model in [8]. Thus, for Kou's model, we accordingly revise the MATLAB code in [8]. The TME and Extrap-IMEX methods are enhanced by the Richardson extrapolation [16] to reach higher than second-order accuracy in space. In Table II, the ETI-Cheb method only achieves second-order accuracy in space. It is because the ETI-Cheb method calculates the integral by the Gaussian quadrature based on Chebyshev nodes, and hence the discontinuous density function in Kou's model hampers the convergence rate. The other numerical results are similar to the ones in Table I. The time-independent TME method remains efficient, while the ETI-Cheb suffers from large n and the Extrap-IMEX method is bothered by long-maturity.

Table II. Pricing European and double barrier call options in Kou's jump-diffusion model with $T = 2$.

	n	TME		ETI-Cheb		Extrap-IMEX		
		error	time	error	time	timesteps	error	time
Euro	31	1.6e-3	0.08	3.5e-1	0.01	100	1.6e-3	0.01
	63	1.1e-4	0.10	8.5e-2	0.02	144	1.1e-4	0.02
	127	7.3e-6	0.13	2.1e-2	0.06	196	7.3e-6	0.08
	255	4.6e-7	0.21	5.3e-3	0.43	292	4.7e-7	0.43
	511	2.9e-8	0.38	1.3e-3	2.97	360	2.0e-8	1.70
DB	14	1.6e-5	0.03	7.2e-5	0.01	168	1.6e-5	0.02
	29	1.0e-6	0.03	2.2e-5	0.01	196	1.0e-6	0.02
	59	6.3e-8	0.04	6.0e-6	0.02	256	6.3e-8	0.03
	119	3.9e-9	0.05	1.5e-6	0.06	324	3.6e-9	0.06
	239	2.4e-10	0.08	3.8e-7	0.41	360	2.0e-10	0.12

6. Concluding remarks

In this paper, we have applied the shift-and-invert Arnoldi method for option pricing with the complexity limited to $\mathcal{O}(n \log n)$ by using the GSF. In our algorithm, the matrix exponential is not computed directly, which is an improvement over [8]. We also give a proof which guarantees the efficiency of the shift-and-invert Arnoldi method in our application.

We remark that the PIDE-based method discussed in this paper is not the mainstream technique for the options and models under consideration. The proposed method is more like a showcase for TME in a practical application. The popular methods in financial industry are the Carr-Madan method in [22] and the COS method in [23, 24]. For European options and some discretely-monitored barrier options with Lévy driven asset, the COS method converges with exponential accuracy [23, 24]. Furthermore, the COS method can be combined with an extrapolation strategy on the Bermudan options for American option pricing. On the other hand, PIDE-based methods are also used for pricing American options recently [25, 26]. Therefore the current work is a stepping-stone to American option pricing and more advanced models like the stochastic volatility with correlated and contemporaneous jumps in return and variance (SVCJ) model [6].

ACKNOWLEDGEMENTS

The authors thank Raymond H. Chan and Tao Wu for inspiring this paper, and also Hong-Kui Pang and Jun Liu for their valuable advices. The authors are also thankful for the referee's helpful comments and suggestions.

REFERENCES

1. Black F, Scholes MS. The pricing of options and corporate liabilities. *Journal of Political Economy* 1973; **81**(3):637–654.
2. Merton RC. Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics* 1976; **3**:125–144.
3. Kou SG. A jump-diffusion model for option pricing. *Management Science* 2002; **48**(8):1086–1101.
4. d'Halluin Y, Forsyth PA, Vetzal KR. Robust numerical methods for contingent claims under jump diffusion processes. *IMA Journal of Numerical Analysis* 2005; **25**(1):87–112.
5. Almendral A, Oosterlee CW. Numerical valuation of options with jumps in the underlying. *Applied Numerical Mathematics* 2005; **53**(1):1–18.
6. Feng L, Linetsky V. Pricing options in jump-diffusion models: an extrapolation approach. *Operations Research* 2008; **56**(2):304–325.
7. Lee ST, Sun HW. Fourth order compact boundary value method for option pricing with jumps. *Advances in Applied Mathematics and Mechanics* 2009; **1**(6):845–861.
8. Tangman DY, Gopaul A, Bhuruth M. Exponential time integration and Chebychev discretisation schemes for fast pricing of options. *Applied Numerical Mathematics* 2008; **58**(9):1309–1319.
9. Moler C, Van Loan C. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Review* 2003; **45**(1):3–49.
10. Hochbruck M, Lubich C. On Krylov subspace approximations to the matrix exponential operator. *SIAM Journal on Numerical Analysis* 1997; **34**(5):1911–1925.
11. Moret I, Novati P. RD-rational approximations of the matrix exponential. *BIT Numerical Mathematics* 2004; **44**(3):595–615.
12. Lee ST, Pang HK, Sun HW. Shift-invert Arnoldi approximation to the Toeplitz matrix exponential. *SIAM Journal on Scientific Computing* 2010; **32**(2):774–792.
13. Gohberg I, Semencul AA. On the inversion of finite Toeplitz matrices and their continuous analogues (in Russian). *Matematicheskie Issledovaniya* 1972; **7**(2):201–223.
14. Chan RH, Jin XQ. *An Introduction to Iterative Toeplitz Solvers*. SIAM: Philadelphia, PA, 2007.
15. Pang HK, Zhang YY, Jin XQ. Tri-diagonal preconditioner for pricing options. *Manuscript*.
16. Garbey M, Shyy W. A least square extrapolation method for improving solution accuracy of PDE computations. *Journal of Computational Physics* 2003; **186**(1):1–23.
17. Higham NJ. The scaling and squaring method for the matrix exponential revisited. *SIAM Review* 2009; **51**(4):747–764.
18. Saad Y. Analysis of some Krylov subspace approximations to the matrix exponential operator. *SIAM Journal on Numerical Analysis* 1992; **29**(1):209–228.
19. Gustafson KE, Rao DKM. *Numerical Range: The Field of Values of Linear Operators and Matrices*. Springer: New York, 1996.
20. Tilli P. Singular values and eigenvalues of non-Hermitian block Toeplitz matrices. *Linear Algebra and its Applications* 1998; **272**(1–3):59–89.
21. Saad Y. *Iterative Methods for Sparse Linear Systems*. SIAM: Philadelphia, PA, 2003.
22. Carr P, Madan DB. Option valuation using the fast Fourier transform. *Journal of Computational Finance* 1999; **2**(4):61–73.
23. Fang F, Oosterlee CW. A novel pricing method for European options based on Fourier-cosine series expansions. *SIAM Journal on Scientific Computing* 2008; **31**(2):826–848.
24. Fang F, Oosterlee CW. Pricing early-exercise and discrete barrier options by Fourier-cosine series expansions. *Numerische Mathematik* 2009; **114**(1):27–62.
25. Rambeerich N, Tangman DY, Gopaul A, Bhuruth M. Exponential time integration for fast finite element solutions of some financial engineering problems. *Journal of Computational and Applied Mathematics* 2009; **224**(2):668–678.
26. Toivanen J. A high-order front-tracking finite difference method for pricing American options under jump-diffusion models. *Journal of Computational Finance* 2010; **13**(3):61–79.

