SIAM J. SCI. COMPUT.
Vol. 32, No. 2, pp. 774–792

SHIFT-INVERT ARNOLDI APPROXIMATION TO THE TOEPLITZ MATRIX EXPONENTIAL*

SPIKE T. LEE†, HONG-KUI PANG†, AND HAI-WEI SUN†

Abstract. The shift-invert Arnoldi method is employed to generate an orthonormal basis from the Krylov subspace corresponding to a real Toeplitz matrix and an initial vector. The vectors and recurrence coefficients produced by this method are exploited to approximate the Toeplitz matrix exponential. Toeplitz matrix inversion formula and rapid Toeplitz matrix-vector multiplications are utilized to lower the computational costs. For convergence analysis, a sufficient condition is established to guarantee that the error bound is independent of the norm of the matrix. Numerical results are given to demonstrate the efficiency of the method.

Key words. Toeplitz matrix, matrix exponential, Krylov subspace, shift-invert Arnoldi method, numerical range

AMS subject classifications. 65L05, 65N22, 65F10, 65F15

DOI. 10.1137/090758064

1. Introduction.

An $n \times n$ *Toeplitz* matrix A_n is defined as follows:

$$A_n = \begin{bmatrix} a_0 & a_{-1} & \cdots & a_{2-n} & a_{1-n} \\ a_1 & a_0 & a_{-1} & \cdots & a_{2-n} \\ \vdots & a_1 & a_0 & \ddots & \vdots \\ a_{n-2} & \cdots & \ddots & \ddots & a_{-1} \\ a_{n-1} & a_{n-2} & \cdots & a_1 & a_0 \end{bmatrix};$$

i.e., A_n is constant along its diagonals. We consider the approximation of the *Toeplitz matrix exponential* (TME):

$$w(t) = \exp(-tA_n)v, \tag{1.1}$$

where A_n is a real Toeplitz matrix, v is a given vector, and $t > 0$ is a scaling factor.

Toeplitz matrices emerge from numerous topics like signal and image processing, numerical solutions of partial differential equations and integral equations, and queueing networks; see [5, 6] and the references therein. Algorithms for solving Toeplitz systems have been under in-depth study over the last 20 years. Apart from Toeplitz system solvers, the TME plays a key role in various application fields. In computational finance, Toeplitz matrices can be seen from the option pricing framework in jump-diffusion models, where a partial integro-differential equation (PIDE) needs to be solved. Tangman et al. [32] reduced the problem to the approximation of a real nonsymmetric TME. In integral equations, the TME also takes part in the numerical solution of Volterra–Wiener–Hopf equations [1].

However, Toeplitz matrices generally are dense. Therefore, the classic methods in [20] for approximating the TME will suffer from $\mathcal{O}(n^3)$ complexity. In [21], the

*Received by the editors May 4, 2009; accepted for publication (in revised form) December 30, 2009; published electronically March 3, 2010. The research was partially supported by the research grant 033/2009/A from FDCT of Macao, UL020/08-Y2/MAT/JXQ01/FST and RG063/08-09S/SHW/FST from University of Macau.
http://www.siam.org/journals/sisc/32-2/75806.html
†Department of Mathematics, University of Macau, Macao, China (ma76522@umac.mo, ya87402@umac.mo, HSun@umac.mo).

updated version of [20], Krylov subspace methods are newly included as one of the dubious ways to compute the exponential of a matrix. In fact, the Krylov subspace methods recently have become an efficient means to approximate the matrix exponential [4, 8, 9, 10, 11, 12, 17, 18, 22, 23, 24, 26, 28, 31], especially when the matrix is very large and sparse. The computational cost can be brought down to $\mathcal{O}(n)$ in some cases. The primary objective of these methods is to construct an orthonormal basis from a Krylov subspace with regard to a certain matrix. This is achieved by the Lanczos process for symmetric matrices or by the Arnoldi process for nonsymmetric matrices, while both processes require only matrix-vector multiplications. Once the basis is constructed, preferably at fewer costs, all that is left to do is to approximate a comparatively smaller matrix exponential. In particular, Moret and Novati [24] improved the Arnoldi method with a shift-invert technique which allowed them to speed up the Arnoldi process. They also presented an error estimation in terms of the numerical range of a matrix. In [10], van den Eshof and Hochbruck also applied a similar idea to revise the Lanczos process for symmetric matrices, though from a different point of view. The brilliant performance of such modified Krylov subspace methods arouses our interest, and what is more, we recall that matrix-vector products are included during the process. It is well known that Toeplitz matrices possess great structures and properties, and their matrix-vector multiplications can be computed by the fast Fourier transform (FFT) with $\mathcal{O}(n \log n)$ complexity [5, 6]. For this reason we expect that the operation cost of TME should be less than $\mathcal{O}(n^3)$.

In this paper, we propose an algorithm to approximate the TME (1.1). Our scheme resembles the one in [24], i.e., to adjust the Arnoldi process for better productivity. Meanwhile, the transformed formulation requires the inverse of the Toeplitz matrix. By making use of the Toeplitz structure and the famous Gohberg–Semencul formula (GSF) in [13], we can reduce the computational cost to $\mathcal{O}(n \log n)$ in total. As in [24], we will establish a sufficient condition for an error bound which is independent of $\|tA_n\|_2$, but in Toeplitz fashion instead. As an application, a TME which stems from a PIDE is considered. Numerical results will illustrate the efficiency and robustness of our method. The rest of the paper is arranged as follows. In section 2 we introduce the background of Toeplitz matrices. In section 3 we first bring out the Arnoldi method for a general matrix exponential, then utilize the shift-invert technique to accelerate the Arnoldi process. Implementation and error estimation of the shift-invert Arnoldi method for Toeplitz matrices are presented in section 4. In section 5 we report the numerical results. At last we give the concluding remarks in section 6.

2. Background of Toeplitz matrices. As a special case of Toeplitz matrix, an $n \times n$ matrix is called *circulant* if it is defined with the following form:

$$C_n = \begin{bmatrix} c_0 & c_{n-1} & \cdots & c_2 & c_1 \\ c_1 & c_0 & c_{n-1} & \cdots & c_2 \\ \vdots & c_1 & c_0 & \ddots & \vdots \\ c_{n-2} & \cdots & \ddots & \ddots & c_{n-1} \\ c_{n-1} & c_{n-2} & \cdots & c_1 & c_0 \end{bmatrix}.$$

Moreover, a circulant matrix can be diagonalized by the Fourier matrix F_n; i.e.,

$$C_n = F_n^* \Lambda_n F_n, \tag{2.1}$$

where the entries of F_n are given by

$$[F_n]_{j,k} = \frac{1}{\sqrt{n}} e^{2\pi ijk/n}, \qquad i \equiv \sqrt{-1}, \qquad 0 \le j,k \le n-1,$$

and Λ_n is a diagonal matrix holding the eigenvalues of C_n.

From (2.1), we can determine Λ_n in $\mathcal{O}(n \log n)$ operations by taking only one n-length FFT of the first column of C_n [5, 6]. Furthermore, we can consider the computation of a circulant matrix multiplied by a vector. Suppose u is the given vector. The multiplication $C_n u$ or $C_n^{-1} u$ is then computed by a couple of FFTs in $\mathcal{O}(n \log n)$ operations provided that Λ_n is already obtained. Similarly, an $n \times n$ *skew-circulant* matrix is defined as

$$\hat{C}_n = \begin{bmatrix} \hat{c}_0 & -\hat{c}_{n-1} & \cdots & -\hat{c}_2 & -\hat{c}_1 \\ \hat{c}_1 & \hat{c}_0 & -\hat{c}_{n-1} & \cdots & -\hat{c}_2 \\ \vdots & \hat{c}_1 & \hat{c}_0 & \ddots & \vdots \\ \hat{c}_{n-2} & \cdots & \ddots & \ddots & -\hat{c}_{n-1} \\ \hat{c}_{n-1} & \hat{c}_{n-2} & \cdots & \hat{c}_1 & \hat{c}_0 \end{bmatrix}.$$

Note that a skew-circulant matrix has the following spectral decomposition:

$$\hat{C}_n = \Omega_n^* F_n^* \hat{\Lambda}_n F_n \Omega_n,$$

where $\Omega_n = \mathrm{diag}[1, e^{-i\pi/n}, \ldots, e^{-i(n-1)\pi/n}]$ and $\hat{\Lambda}_n$ is a diagonal matrix containing the eigenvalues of $\hat{C}_n$. We remark that skew-circulant matrices also could have their matrix-vector multiplications $\hat{C}_n u$ done by FFTs with $\mathcal{O}(n \log n)$ complexity. See [5, 6] for details.

If the Toeplitz matrix-vector product $A_n u$ is wanted, we can first embed A_n into a $2n \times 2n$ circulant matrix; i.e.,

$$\begin{bmatrix} A_n & \times \\ \times & A_n \end{bmatrix} \begin{bmatrix} u \\ 0 \end{bmatrix} = \begin{bmatrix} A_n u \\ \dagger \end{bmatrix}. \tag{2.2}$$

Now that we are back to the circulant case, the multiplication is carried out as discussed before, with $\mathcal{O}(n \log n)$ complexity. Note that the computation of $A_n u$ is approximately two times the cost of $C_n u$ or $\hat{C}_n u$, based on the premise that their spectra are already obtained. Roughly speaking, two FFTs of size n are needed for a circulant or skew-circulant matrix-vector product, while four FFTs of size n are required for a Toeplitz matrix-vector product.

2.1. Generating functions. It is common to assume that the diagonals $\{a_k\}_{k=-n+1}^{n-1}$ of a Toeplitz matrix A_n are the Fourier coefficients of a function f:

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

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

Suppose f is a complex-valued function. We symbolize the real and imaginary parts of f by $Re(f)$ and $Im(f)$ respectively. Let $\mathcal{T}_n[f]$ denote a Toeplitz matrix generated by f. Note that we can write $\mathcal{T}_n[f]$ as

$$\mathcal{T}_n[f] = \mathcal{T}_n[Re(f)] + i \cdot \mathcal{T}_n[Im(f)]. \tag{2.3}$$

By (2.3), one can easily find a way to check whether a Toeplitz matrix is a real matrix. Suppose $Re(f)$ is an even function and $Im(f)$ is an odd function. By the definition of generating function, we conclude that $\mathcal{T}_n[Re(f)]$ is a real symmetric matrix, and $\mathcal{T}_n[Im(f)]$ is a real skew-symmetric matrix multiplied by i. Therefore, by (2.3), a Toeplitz matrix $\mathcal{T}_n[f]$ is a real matrix when $Re(f)$ is an even function and $Im(f)$ is an odd function. We remark that $\mathcal{T}_n[f]$ is reduced to a nonsymmetric Toeplitz matrix in $\mathbb{R}^{n\times n}$ when $Im(f) \neq 0$.

Throughout this paper we consider a Toeplitz matrix in $\mathbb{R}^{n\times n}$. It is assumed that its generating function $f \in \mathcal{C}_{2\pi}$, where $\mathcal{C}_{2\pi}$ contains all the 2π-periodic continuous complex-valued functions. Furthermore, we define

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

and let f satisfy the assumptions below:

$$Re(f) \geq 0 \quad \text{and} \quad \|Im(f)/Re(f)\|_\infty = \mathcal{O}(1). \tag{2.4}$$

The two assumptions in (2.4) will be used later for error estimation.

2.2. Numerical range. Here we first give the definition of numerical range, which will come into use afterward.

DEFINITION 2.1 (see [24]). *The numerical range of a matrix A_n is defined as a subset in the complex plane $\mathbb{C}$:*

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

In addition, we also need the concept of *convex hull* to supplement our work.

DEFINITION 2.2 (see [27]). *The intersection of all the convex sets containing a given set U is called the convex hull of U and is denoted by* $\mathrm{conv}(U)$.

Let

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

be the range of f. According to Theorem 5.1 in [33], there exists a relation between the numerical range and the generating function of a Toeplitz matrix. In this paper, we only need a special case of Theorem 5.1 in [33]. For brevity, it is summarized as the following theorem.

THEOREM 2.3 (see [33, Theorem 5.1]). *Let $W(A_n)$ be the numerical range of $A_n = \mathcal{T}_n[f]$, where $f \in \mathcal{C}_{2\pi}$. Then $W(A_n)$ is a subset of the closure of* $\mathrm{conv}(\Omega(f))$*; i.e.,*

$$W(A_n) \subseteq \overline{\mathrm{conv}(\Omega(f))}.$$

2.3. Gohberg–Semencul formula. The Toeplitz matrix inversion is also studied thoroughly beside Toeplitz matrix-vector multiplication [15]. In [13], Gohberg and Semencul discovered the GSF for the inverse of a Toeplitz matrix A_n. The formula shows that the inverse A_n^{-1} can be *explicitly* represented by its first column $x = [x_1, x_2, \ldots, x_n]^\intercal$ and last column $y = [y_1, y_2, \ldots, y_n]^\intercal$ provided that $x_1 \neq 0$. The GSF is given by

$$A_n^{-1} = \frac{1}{x_1}\left(X_n Y_n^\intercal - \hat{Y}_n \hat{X}_n^\intercal\right), \tag{2.5}$$

where X_n, Y_n, $\hat{X}_n$, and $\hat{Y}_n$ all are lower triangular Toeplitz matrices given by

$$X_n = \begin{bmatrix} x_1 & 0 & \cdots & 0 \\ x_2 & x_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x_n & x_{n-1} & \cdots & x_1 \end{bmatrix}, \quad Y_n = \begin{bmatrix} y_n & 0 & \cdots & 0 \\ y_{n-1} & y_n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ y_1 & y_2 & \cdots & y_n \end{bmatrix},$$

$$\hat{X}_n = \begin{bmatrix} 0 & \cdots & 0 & 0 \\ x_n & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ x_2 & \cdots & x_n & 0 \end{bmatrix}, \quad \text{and} \quad \hat{Y}_n = \begin{bmatrix} 0 & \cdots & 0 & 0 \\ y_1 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ y_{n-1} & \cdots & y_1 & 0 \end{bmatrix}.$$

Note that x and y can also be regarded as the solutions of two linear systems,

$$A_n x = e_1 \quad \text{and} \quad A_n y = e_n, \tag{2.6}$$

where e_1 and e_n are the first and last columns of an identity matrix. By the GSF (2.5), the inverse of A_n is decided through the condition $x_1 \neq 0$, and the solvability of two linear systems (2.6), both of which hold the Toeplitz matrix A_n as the coefficient matrix.

Particularly if we want to solve plenty of Toeplitz systems which share the same coefficient matrix A_n, we only need to solve two Toeplitz systems (2.6) to obtain the first and last column of A_n^{-1}. Then the GSF (2.5) yields an explicit representation of A_n^{-1} in terms of four triangular Toeplitz matrices, and all the desired solutions are derived via Toeplitz matrix-vector multiplications (2.2) instead of solving many Toeplitz systems. Nevertheless, four Toeplitz matrix-vector products require about sixteen FFTs of size n to complete. To relieve that burden, we can make use of the fact that two lower (or upper) triangular Toeplitz matrices commute, and then the four Toeplitz matrices in (2.5) can be factorized into two circulant matrices and two skew-circulant matrices [25]:

$$\begin{aligned} A_n^{-1} &= \frac{1}{x_1}\left(X_n Y_n^\intercal - \hat{Y}_n \hat{X}_n^\intercal\right) \\ &= \frac{1}{2x_1}\left[\left(X_n + \hat{X}_n^\intercal\right)\left(Y_n^\intercal - \hat{Y}_n\right) + \left(Y_n^\intercal + \hat{Y}_n\right)\left(X_n - \hat{X}_n^\intercal\right)\right], \end{aligned} \tag{2.7}$$

which has the following spectral decomposition:

$$\frac{1}{2x_1} F_n^* \left[\Lambda_n^{(1)} F_n \Omega_n^* F_n^* \Lambda_n^{(2)} + \Lambda_n^{(3)} F_n \Omega_n^* F_n^* \Lambda_n^{(4)}\right] F_n \Omega_n, \tag{2.8}$$

where $\Lambda_n^{(1)}$ and the others are diagonal matrices holding the eigenvalues of $X_n + \hat{X}_n^\intercal$ and so on. Therefore, the solutions can also be obtained through (2.8), and only about six FFTs of length n are needed [5, 6, 25].

2.4. Fast Toeplitz solvers. The GSF (2.5) gives an exact representation of the inverse of a Toeplitz matrix, but instead we have to solve two Toeplitz systems $A_n x = e_1$ and $A_n y = e_n$ in (2.6). In this paper, we prefer the iterative methods with complexity $\mathcal{O}(n \log n)$ over the direct methods with complexity $\mathcal{O}(n \log^2 n)$ [2, 5, 6]. For example, one can choose the conjugate gradient normal equation method [14] for solving nonsymmetric Toeplitz systems like (2.6), with T. Chan's circulant preconditioner. The T. Chan's circulant preconditioner $c(A_n)$ is defined to be the minimizer of

$$\|A_n - C_n\|_F$$

over all $n \times n$ circulant matrices C_n. Here $\|\cdot\|_F$ denotes the Frobenius norm. The matrix $c(A_n)$ also is known as the optimal circulant preconditioner of A_n [5, 6]. It is shown that $c(A_n)_k$, the k-th diagonal of $c(A_n)$, is equivalent to

$$c(A_n)_k = \begin{cases} \dfrac{(n-k)a_k + k a_{k-n}}{n}, & 0 \le k \le n-1, \\ \\ c(A_n)_{n+k}, & 1-n \le k < 0. \end{cases}$$

The T. Chan's circulant preconditioner suits a wide class of Toeplitz matrices; see [5, 6] for more discussions.

In [7], Chan and Yeung provided some studies on solving nonsymmetric Toeplitz systems by the conjugate gradient normal equation method with T. Chan's preconditioner. Suppose $A_n = \mathcal{T}_n[f]$. For a generating function $f \in \mathcal{C}_{2\pi}$ with no zeros on $[-\pi, \pi]$, the spectra of the iteration matrices $(c(A_n)^{-1}A_n)^*(c(A_n)^{-1}A_n)$ are clustered around 1. Furthermore, if A_n is well-conditioned, then the total complexity for solving the Toeplitz systems is of $\mathcal{O}(n \log n)$; see the details in [7]. We note that if there is a straight line in the complex plane $\mathbb{C}$ such that the origin is not on the line and $\Omega(f)$ lies completely on the originless side of it, then the smallest singular value of A_n is positive independent of n. In this case, A_n is well-conditioned [30].

Alternatively, the GMRES method [29] with T. Chan's preconditioner is another choice for solving (2.6). In many applications, practitioners have shown that the GMRES method may converge amazingly fast.

3. Shift-invert Arnoldi method. Krylov subspace methods for computing the matrix exponential have been widely investigated over the years [4, 8, 9, 10, 11, 12, 17, 18, 22, 23, 24, 26, 28, 31]. The main concept of such methods is to approximately project the exponential of a large matrix onto a small Krylov subspace. In this section, we will go through the Arnoldi process and then adopt a shift-invert technique to it.

3.1. Arnoldi method. First we briefly introduce the standard Arnoldi method for approximating the matrix exponential $w(t) = \exp(-tA_n)v$; see [28] for details. In the beginning, the idea of using Krylov subspace methods sprang from approximating the exponential function with a polynomial p_{m-1} of degree $m-1$:

$$\exp(A_n)v \approx p_{m-1}(A_n)v,$$

and the fact that this approximation belongs to an m-th dimension Krylov subspace

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

As usual, we have to generate an orthonormal basis of this Krylov subspace $\mathcal{K}_m$. The renowned Arnoldi process is used in our case. We first summarize the Arnoldi process as the following algorithm with $v_1 = v/\|v\|_2$ being an initial vector.

ALGORITHM 1: ARNOLDI PROCESS

1. Initialize: Compute $v_1 = v/\|v\|_2$
2. Iterate: Do $j = 1, \ldots, m$
   (a) Compute $u := A_n v_j$
   (b) Do $k = 1, \ldots, j$
      i. Compute $h_{k,j} := (u, v_k)$
      ii. Compute $u := u - h_{k,j} v_k$
   (c) Compute $h_{j+1,j} := \|u\|_2$ and $v_{j+1} := u/h_{j+1,j}$

We follow the Arnoldi algorithm step by step and will eventually reach the relation:

$$A_n V_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^{\intercal}, \tag{3.1}$$

where $V_m = [v_1, \ldots, v_m]$ is the resulting $n \times m$ matrix containing the orthonormal basis, H_m is an $m \times m$ upper Hessenberg matrix with $H_m = V_m^* A_n V_m$, and e_j denotes the j-th column of the identity matrix. More specifically, H_m is the projection of the linear transformation A_n onto the subspace $\mathcal{K}_m$. The formulation (3.1) thus leads to an approximation:

$$\exp(-tA_n)v \approx \beta V_m \exp(-tH_m)e_1, \tag{3.2}$$

where $\beta = \|v\|_2$. The approximation (3.2) indicates that the large matrix exponential $\exp(-tA_n)$ is replaced by a small matrix exponential $\exp(-tH_m)$ [10, 24, 28].

3.2. Arnoldi method with shift-invert technique. Note that a small m would be greatly preferable, or the approximation (3.2) barely means anything after all. However, it is shown in [17] that m is close to $\mathcal{O}(\|tA_n\|_2)$. That means the standard Arnoldi method could be unsatisfactory if $\|tA_n\|_2$ is large. To untangle this knot, one can exploit a potential advantage of Krylov subspace methods; i.e., they incline to locate well-separated eigenvalues faster [10]. For instance, Moret and Novati [24] put this advantage into practice by filling in a shift-invert technique. Such a maneuver can be found in numerical methods for eigenvalue problems [3].

Let I be the identity matrix. The shift-invert technique is to apply the Arnoldi process to a shifted and inverted matrix $(I + \gamma A_n)^{-1}$, which stresses the required eigenvalues, with a shift parameter $\gamma > 0$.

ALGORITHM 2: ARNOLDI PROCESS WITH SHIFT-INVERT TECHNIQUE

1. Initialize: Compute $v_1 = v/\|v\|_2$
2. Iterate: Do $j = 1, \ldots, m$
   (a) Compute $u := (I + \gamma A_n)^{-1} v_j$
   (b) Do $k = 1, \ldots, j$
      i. Compute $h_{k,j} := (u, v_k)$
      ii. Compute $u := u - h_{k,j} v_k$
   (c) Compute $h_{j+1,j} := \|u\|_2$ and $v_{j+1} := u/h_{j+1,j}$

By using $v_1 = v/\|v\|_2$ as an initial vector, we come similarly to the following formulation:

$$(I + \gamma A_n)^{-1} V_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^{\intercal}; \tag{3.3}$$

i.e., the matrix $(I+\gamma A_n)^{-1}$ is projected onto a Krylov subspace. Then the renewed formulation creates another approximation for $\exp(-tA_n)v$ [24]:

$$\exp(-tA_n)v \approx \beta V_m \exp(-\tau(H_m^{-1}-I))e_1 \equiv \beta V_m g(H_m)e_1, \tag{3.4}$$

where $\tau = t/\gamma$, $\beta = \|v\|_2$, and

$$g(z) = \exp(-\tau(z^{-1}-1)). \tag{3.5}$$

After the shift-invert Arnoldi process, the smaller matrix exponential of size $m \times m$ takes over, and we let $w_m(t)$ denote the approximation in (3.4):

$$w_m(t) = \beta V_m g(H_m)e_1. \tag{3.6}$$

Then this algorithm is called the *shift-invert Arnoldi method*. Here we remark that this treatment also has been presented independently by [10] for symmetric matrices.

Moret and Novati [24] estimated the error between the approximation $w_m(t)$ and the vector $w(t) = \exp(-tA_n)v$ from the viewpoint of sectorial operator and numerical range. For convenience, we simply assume $\beta = \|v\|_2 = 1$ in this part. Following [24], we first introduce several relevant concepts.

Let $\Sigma_{\alpha,\vartheta}$ be the following set:

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

i.e., $\Sigma_{\alpha,\vartheta}$ is an unbounded sector in the right-half plane with semiangle $\vartheta < \pi/2$ and vertex lying on the nonnegative real axis. Note that $\Sigma_{\alpha,\vartheta}$ is symmetric about the real axis. In addition, we define a bounded sector with vertex $(0,0)$ as

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

Let Π_s denote the set of all algebraic polynomials of degree less than s. Proposition 2.1 in [24] suggests an estimation for $g(z)$ in (3.5) on a given sector.

THEOREM 3.1 (see [24, Proposition 2.1]). *Let $\tau = t/\gamma$ be fixed. Given any sector $S_{\rho,\vartheta}$, for every integer $k \geq 0$ and for every $\varepsilon > 0$, there is $s > 0$ and a polynomial $p_s \in \Pi_s$ such that*

$$\left| \frac{g(z) - p_s(z)}{z^k} \right| < \varepsilon$$

for every $z \in \overline{S_{\rho,\vartheta}}$.

We let Z_n be the shifted and inverted matrix

$$Z_n = (I + \gamma A_n)^{-1}.$$

Proposition 3.2 in [24] gives the following estimate for the approximation $w_m(t)$ in (3.6) under the condition of $W(Z_n) \subset \overline{S_{\rho,\vartheta}}$.

THEOREM 3.2 (see [24, Proposition 3.2]). *Let $W(Z_n) \subset \overline{S_{\rho,\vartheta}}$. Let Γ^* be the contour of any sector S_{ρ^*,ϑ^*} with $\rho^* \geq \rho(1-\sin(\vartheta^*-\vartheta))^{-1}$ and $\vartheta < \vartheta^* < \pi/2$. Then*

$$\|w(t) - w_m(t)\| \leq \frac{1}{\pi \sin(\vartheta^* - \vartheta)} \min_{p_{m-1} \in \Pi_{m-1}} \int_{\Gamma^*} \left| \frac{g(z) - p_{m-1}(z)}{z} \right| |dz|.$$

Without loss of generality, let ϑ^* be the middle value between ϑ and $\pi/2$; i.e.,

$$\vartheta^* = \frac{2\vartheta + \pi}{4}.$$

Meanwhile ρ^* can be chosen as

$$\rho^* = \rho(1 - \sin(\vartheta^* - \vartheta))^{-1} < \infty.$$

Suppose τ is already fixed as required in Theorem 3.1. By Theorem 3.1, the integrand in Theorem 3.2 is arbitrarily small for $z \in \overline{S_{\rho^*,\vartheta^*}}$ as long as m is large enough. Moreover, the contour Γ^* is a rectifiable curve since ρ^* is finite. Therefore, Theorem 3.2 together with Theorem 3.1 implies that

$$w_m(t) \to w(t), \quad m \to \infty.$$

Aside from convergence, we can tell from Theorem 3.2 that $\|tA_n\|_2$ is not connected with the error bound. Obviously, the error bound in Theorem 3.2 does not contain the norm $\|A_n\|_2$, which has been the major setback in matrix exponential approximation [28]. On the other hand, the integrand consists of the function $g(z)$ in (3.5), which is related to $\tau = t/\gamma$. In practice, we can adjust the value of the shift parameter γ in order to eliminate t. Thus t has nothing to do with the error bound once γ is appropriately selected. After all, the error bound of the shift-invert Arnoldi approximation $w_m(t)$ does not rely on $\|tA_n\|_2$. In general, a proper choice of γ results also in a fixed τ. Therefore, Theorem 3.2 itself concludes that $w_m(t)$ converges to $w(t)$, and at the same time, the error bound is independent of $\|tA_n\|_2$.

Theorem 3.2 also hints that a ϑ not close to $\pi/2$ is more favored. It is because a ϑ close to $\pi/2$ leads to a small $\sin(\vartheta^* - \vartheta)$, and hence turns the factor $(\pi \sin(\vartheta^* - \vartheta))^{-1}$ towards infinity. Particularly when $\vartheta = 0$, it is reduced to the symmetric case, and related error estimations can be found in [10].

Theorem 3.2 provides a sufficient condition for error estimate in terms of the numerical range $W(Z_n)$, where Z_n is the shifted and inverted matrix. In fact, we also can consider the numerical range of the original matrix A_n. Suppose A_n is a *sectorial operator*; i.e., [24]

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

Then $Z_n = (I + \gamma A_n)^{-1}$ is obtained through the transformation $\varphi(z) = (1 + \gamma z)^{-1}$ in a matrix case. Note that the function φ maps an unbounded sector $\Sigma_{\alpha,\vartheta}$ into

$$\Sigma_{0,\vartheta} \cap D_{(1+\gamma\alpha)^{-1}/2},$$

where $D_{(1+\gamma\alpha)^{-1}/2}$ is a disk of center and radius $(1 + \gamma\alpha)^{-1}/2$. More specifically, φ does not change the value of ϑ during the transformation. Furthermore, there exists a bounded sector $S_{(1+\gamma\alpha)^{-1},\vartheta}$ such that

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

which meets the condition of Theorem 3.2. Thus we acquire an alternative condition for Theorem 3.2. If A_n is a sectorial operator, then Theorem 3.2 holds true and the error bound does not depend on $\|tA_n\|_2$; see [24] for more details.

4. Implementation and error estimation. In this section, we go into further details of approximating the real TME by the shift-invert Arnoldi method. Recall the assumptions that the real Toeplitz matrix $A_n = \mathcal{T}_n[f]$ is generated by $f \in \mathcal{C}_{2\pi}$, and f satisfies the assumptions in (2.4):

$$Re(f) \geq 0 \quad \text{and} \quad \|Im(f)/Re(f)\|_\infty = \mathcal{O}(1).$$

We remark that $Re(f) \geq 0$ is not a necessary condition since the other possible cases can easily be handled with certain shifting treatment [10, 24]. We first clarify how the shift-invert Arnoldi method practically works and then investigate the error estimation by using generating functions.

4.1. Implementation of shift-invert Arnoldi method. In the standard Arnoldi algorithm, the matrix-vector product $A_n v_j$ for $j = 1, \ldots, m$ is evaluated at each iteration step. If A_n is a Toeplitz matrix, then these multiplications can be carried out by (2.2) in $\mathcal{O}(n \log n)$ operations. Once we have included the shift-invert technique, the required matrix-vector multiplication becomes $(I + \gamma A_n)^{-1} v_j$ at each iteration step. Suppose the inverse of $I + \gamma A_n$ can be found beforehand. Then all $(I + \gamma A_n)^{-1} v_j$ are obtained by matrix-vector products instead of solving systems. Since A_n is a real Toeplitz matrix, the shifted matrix $I + \gamma A_n$ also is a real Toeplitz matrix for a fixed γ. Recall that the GSF (2.5) provides an explicit representation of the inverse of a Toeplitz matrix, therefore this celebrated formula would come in handy in our case.

We first gather the shift-invert Arnoldi method for real TME as the algorithm below:

Algorithm 3: Shift-invert Arnoldi method for real TME

1. Solve $(I + \gamma A_n)x = e_1$ and $(I + \gamma A_n)y = e_n$ by the method in section 2.4
2. Perform the Arnoldi process in which each multiplication $(I + \gamma A_n)^{-1} v_j$ is calculated through (2.8) by FFTs
3. Evaluate the approximation $w_m(t) = \beta V_m g(H_m) e_1$

In order to apply the GSF (2.5), we first need to verify that $x_1 \neq 0$. Since $I + \gamma A_n$ is real and nonsingular, x should be a real vector not equal to zero. From the equality

$$(I + \gamma A_n)x = e_1,$$

we left-multiply $x^\intercal$ and get

$$x^\intercal (I + \gamma A_n)x = x_1.$$

Note that $\gamma > 0$ and $Re(f) \geq 0$; it follows that:

$$x_1 = x^\intercal x + \gamma x^\intercal \mathcal{T}_n(Re(f))x > 0;$$

i.e., x_1 is not equal to zero, and hence the GSF (2.5) is feasible.

To seek the inverse of the Toeplitz matrix $I + \gamma A_n$, we have to start with finding the first and last columns x and y of $(I + \gamma A_n)^{-1}$ by solving the following two systems:

$$(I + \gamma A_n)x = e_1 \quad \text{and} \quad (I + \gamma A_n)y = e_n, \tag{4.1}$$

where the coefficient matrix $I + \gamma A_n$ is real and Toeplitz. Recall that the generating function f of A_n is assumed to have a nonnegative real part $Re(f) \geq 0$ in (2.4). Thus

the generating function of $I + \gamma A_n$ has a strictly positive real part

$$Re(1 + \gamma f) \geq 1 > 0;$$

i.e., the range of $1 + \gamma f$ lies on the right-hand side of a straight line in the complex plane

$$\{z \in \mathbb{C} : \ Re(z) = 1\}.$$

In such case $I + \gamma A_n$ is nonsingular and well-conditioned, the two Toeplitz systems (4.1) can be solved rapidly and accurately by fast Toeplitz solvers with $\mathcal{O}(n \log n)$ complexity; see section 2.4.

After collecting the two columns x, y and making sure $x_1 \neq 0$, we proceed to find the spectral decomposition of $(I + \gamma A_n)^{-1}$ in (2.8). Then we can compute all the matrix-vector products $(I+\gamma A_n)^{-1}v_j$ exactly through FFTs in $\mathcal{O}(n \log n)$ operations [5, 6]. After performing the shift-invert Arnoldi process, we derive the matrix formulation (3.3). Finally, the resulting small matrix exponential in (3.6) can be evaluated by the scaling and squaring method [16] or other classic methods [20, 21], provided that m is small enough.

Note that steps 1 and 2 carry most of the workloads in the whole algorithm. For a Toeplitz matrix, we manage to reduce them all to $\mathcal{O}(n \log n)$ operations by making use of the Toeplitz properties. Thus the computational cost of the shift-invert Arnoldi method for approximating the real TME is of $\mathcal{O}(n \log n)$.

We then clarify the difference in computational costs of the shift-invert Arnoldi method and the standard Arnoldi method. For the standard one, only one Toeplitz matrix-vector product $A_n v_j$ is involved at each iteration step. That is to say, about four FFTs of size n are carried out. For the shift-invert Arnoldi method, we first solve two additional Toeplitz systems (4.1) by the iterative Toeplitz solver introduced in section 2.4. Then in each iteration of the shift-invert Arnoldi process, about six FFTs of size n in (2.8) are utilized to compute $(I+\gamma A_n)^{-1}v_j$. If we focus on each iteration, then the calculation of $(I+\gamma A_n)^{-1}v_j$ is nearly one-and-a-half times the costs of $A_n v_j$.

In any respect, the shift-invert Arnoldi method is slightly more expensive to implement than the standard Arnoldi method. However, the iteration number of the shift-invert Arnoldi method is usually far smaller because of the shift-invert technique. Therefore, the increased computational costs would actually pay off, especially when $\|tA_n\|_2$ is large. Later we will illustrate this argument by numerical experiments in section 5.

4.2. Error estimation by generating functions. In [24], Moret and Novati diagnosed the error bound of the shift-invert Arnoldi approximation. The premise is that A_n is a sectorial operator. Therefore, the next thing we do is sort out such characteristics of Toeplitz matrices by using their generating functions.

LEMMA 4.1. *Let $f \in \mathcal{C}_{2\pi}$ and $\vartheta = \arctan \|Im(f)/Re(f)\|_\infty$. If f satisfies the two assumptions in* (2.4)*, then we have $\vartheta < \pi/2$ and*

$$\overline{\mathrm{conv}(\Omega(f))} \subseteq \overline{\Sigma_{0,\vartheta}},$$

where

$$\overline{\Sigma_{0,\vartheta}} = \left\{ z \in \mathbb{C} : \ |\arg z| \leq \vartheta, \ 0 < \vartheta < \frac{\pi}{2} \right\}.$$

Proof. It is known from the assumption that $\|Im(f)/Re(f)\|_\infty \leq M < \infty$. Thus

$$\vartheta = \arctan \|Im(f)/Re(f)\|_\infty \leq \arctan M < \pi/2.$$

Since $Re(f) \geq 0$, we have for any $z = f(\theta) \in \Omega(f)$ that

$$|\arg z| = \arctan |Im(f(\theta))/Re(f(\theta))| \leq \arctan \|Im(f)/Re(f)\|_\infty = \vartheta.$$

It follows that $z \in \overline{\Sigma_{0,\vartheta}}$, which implies

$$\Omega(f) \subseteq \overline{\Sigma_{0,\vartheta}}.$$

Apparently $\overline{\Sigma_{0,\vartheta}}$ is a closed convex set; the proof is completed. ∎

THEOREM 4.2. *Suppose $A_n = \mathcal{T}_n[f] \in \mathbb{R}^{n\times n}$ with $f \in \mathcal{C}_{2\pi}$, and $Z_n = (I+\gamma A_n)^{-1}$ with $\gamma > 0$. If f satisfies the two assumptions in* (2.4)*, then A_n is a sectorial operator, and furthermore,*

$$W(Z_n) \subset \overline{S_{1,\vartheta}},$$

where $\vartheta = \arctan \|Im(f)/Re(f)\|_\infty$.

Proof. By Lemma 4.1, we have $\vartheta < \pi/2$ and

$$\overline{\text{conv}(\Omega(f))} \subseteq \overline{\Sigma_{0,\vartheta}}.$$

Moreover, Theorem 2.3 points out that

$$W(A_n) \subseteq \overline{\text{conv}(\Omega(f))},$$

which leads to

$$W(A_n) \subseteq \overline{\Sigma_{0,\vartheta}};$$

i.e., A_n is a sectorial operator. Recall that the transformation $\varphi(z) = (1+\gamma z)^{-1}$ maps $\overline{\Sigma_{0,\vartheta}}$ into a set which belongs to a bounded sector $\overline{S_{1,\vartheta}}$ with the same semi-angle. Therefore,

$$W(Z_n) \subset \overline{S_{1,\vartheta}}.$$

The proof is completed. ∎

Theorem 4.2 gives a sufficient condition of whether a real Toeplitz matrix A_n is a sectorial operator, in terms of its generating function f. Recall that Theorem 3.2 guarantees the absence of $\|tA_n\|_2$ in the error bound when A_n is a sectorial operator. In conclusion, if the generating function f of A_n satisfies the condition of Theorem 4.2, then the error bound of the shift-invert Arnoldi approximation does not depend on $\|tA_n\|_2$. In the next section, we will verify this conclusion by numerical experiments.

5. Numerical results. In the following numerical tests, we consider approximating the real TME (1.1), namely

$$w(t) = \exp(-tA_n)v,$$

by the shift-invert Arnoldi method and the standard Arnoldi method. All experiments are conducted in MATLAB. We regard the MATLAB command `expm` as the exact value for $w(t)$. For all tables, "n" denotes the matrix size, and "*tol*" stands for the tolerance of $\|w(t) - w_m(t)\|_2/\|w(t)\|_2 < tol$, where $w_m(t)$ is the numerical approximation to $w(t)$. The column "shft-inv" displays the iteration numbers of the shift-invert Arnoldi method, while "stdrd" shows those of the standard Arnoldi method. For any "-" showing up in the column, it means the number of iterations exceeds 250. The shift parameter is chosen as $\gamma = t/10$.

TABLE 5.1
The numbers of iterations of the shift-invert Arnoldi method and the standard Arnoldi method in Example 1 *and Example* 2.

t	Example 1				Example 2			
	$tol = 10^{-4}$		$tol = 10^{-7}$		$tol = 10^{-4}$		$tol = 10^{-7}$	
	shft-inv	stdrd	shft-inv	stdrd	shift-inv	stdrd	shift-inv	stdrd
1	11	31	31	41	7	10	11	15
10	10	147	22	183	18	43	28	54
100	9	-	18	-	59	148	84	193
1000	9	-	16	-	-	-	-	-

Three examples are given to demonstrate the shift-invert Arnoldi method and the standard Arnoldi method. The first two examples explain mainly how the assumption (2.4) makes a difference, where the vector v is chosen to be the vector of all ones. The third example is an application in computational finance, in which a real nonsymmetric TME is involved.

Example 1. We consider a Toeplitz matrix A_n which is generated by the function

$$f(\theta) = \theta^2 + i \cdot \theta^3, \qquad \theta \in [-\pi, \pi].$$

Note that $Re(f) = \theta^2 \geq 0$ is an even function, and $Im(f) = \theta^3$ is an odd function. According to section 2.1, A_n is a real Toeplitz matrix. It is obvious that the generating function satisfies $\|Im(f)/Re(f)\|_\infty = \mathcal{O}(1)$ and leads to a semi-angle

$$\vartheta = \arctan \|\theta^3/\theta^2\|_\infty = \arctan \pi < \pi/2.$$

By Theorems 4.2 and 3.2, the error bound of the shift-invert Arnoldi method should be independent of $\|tA_n\|_2$. Note that $\|A_n\|_2$ does not depend on the matrix size n in this example, hence we set $n = 512$ and try out different values of t. Numerical results in Table 5.1 show that the iteration numbers of the shift-invert Arnoldi method are indeed independent of $\|tA_n\|_2$, or t in this case. Oppositely, the standard Arnoldi method needs more iterations as t increases.

Example 2. We consider a Toeplitz matrix A_n which is generated by the function

$$f(\theta) = \theta^2 + i \cdot \operatorname{sgn}(\theta), \qquad \theta \in [-\pi, \pi],$$

where $\operatorname{sgn}(\theta)$ is the sign function defined as

$$\operatorname{sgn}(\theta) = \begin{cases} 1, & 0 < \theta \leq \pi, \\ 0, & \theta = 0, \\ -1, & -\pi \leq \theta < 0. \end{cases}$$

Note that $Re(f) = \theta^2 \geq 0$ is an even function and $Im(f) = \operatorname{sgn}(\theta)$ is an odd function. According to section 2.1, A_n is a real Toeplitz matrix. It is easy to see that the quotient $|\operatorname{sgn}(\theta)/\theta^2|$ is unbounded when $\theta \to 0$. Therefore, f does not satisfy the condition (2.4).

As in Example 1, $\|A_n\|_2$ does not rely on n, hence the matrix size is fixed at $n = 512$, and different values of t should be put to the test. In Table 5.1, we see that the shift-invert Arnoldi method is inferior to the previous example, and the number of iterations gradually increases in accordance with $\|tA_n\|_2$, or simply t in this example. It is due to the incapability of meeting the condition $\|Im(f)/Re(f)\|_\infty = \mathcal{O}(1)$ in Theorem 4.2. For the standard Arnoldi method, the iteration numbers still fail to stay steady, just like their counterparts in Example 1.

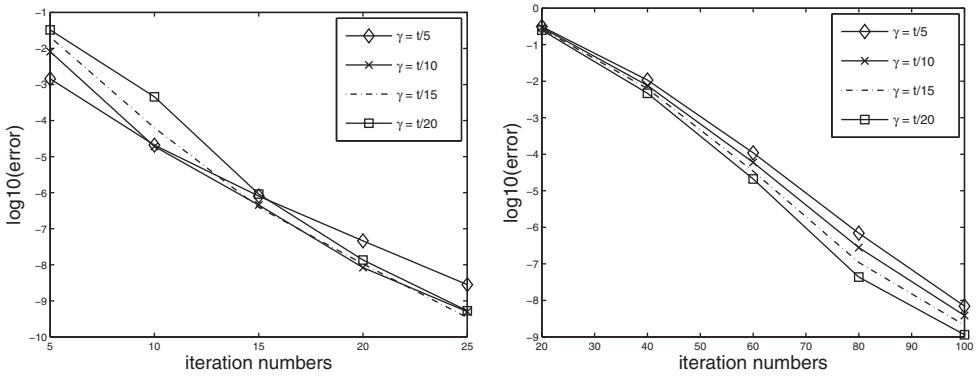

FIG. 5.1. *Iteration numbers of the shift-invert Arnoldi method versus error with* $t = 100$ *for different choices of* γ. *Left picture: Example* 1; *right picture: Example* 2.

In the implementation of the shift-invert Arnoldi method, the choice of γ is an intricate issue. In the symmetric matrix case, there are detailed studies on the optimal choice of γ [10, 26], though an exact value $\gamma = t/10$ is used throughout the numerical experiments therein. However, for the general case, how to pick an appropriate γ remains a puzzle. In [24], which studies the nonsymmetric matrix case, the parameter γ is chosen as t divided by a number varying in the range $[1, 10]$. In our case, the shift parameter similarly is selected as $\gamma = t/10$. We then show that the choice of parameter is not sensitive to convergence after all.

To check the influence of γ, we test four different parameters $\gamma = t/5$, $\gamma = t/10$, $\gamma = t/15$, and $\gamma = t/20$. We can see from Figure 5.1 that γ is not sensitive to convergence in Example 1, as four different choices of γ all lead to fast convergence. In addition, we notice that these γ's perform diversely at different stages. For instance, $\gamma = t/5$ starts out perfectly but fails to keep its pace afterward. From Figure 5.1, we also can observe that γ is not sensitive to convergence in Example 2 as well. All choices of γ have slow convergence; it is due to the fact that Example 2 does not meet the condition (2.4).

Example 3. We consider pricing options for a single underlying asset in Merton's jump-diffusion model [19] as an application of the shift-invert Arnoldi method. In Merton's model, jumps are normally distributed with mean μ and variation σ. The option value $\omega(\xi, t)$ with logarithmic price ξ and backward time t satisfies a forward PIDE on $(-\infty, +\infty) \times [0, T]$:

$$\omega_t = \frac{\nu^2}{2}\omega_{\xi\xi} + \left(r - \lambda\kappa - \frac{\nu^2}{2}\right)\omega_\xi - (r + \lambda)\omega + \lambda \int_{-\infty}^{\infty} \omega(\xi + \eta, t)\phi(\eta)d\eta, \tag{5.1}$$

where T is the maturity time, ν is the stock return volatility, r is the risk-free interest rate, λ is the arrival intensity of a Poisson process, $\kappa = e^{(\mu+\sigma^2/2)} - 1$ is the expectation of the impulse function, and ϕ is the Gaussian distribution given by

$$\phi(\eta) = \frac{e^{-(\eta-\mu)^2/2\sigma^2}}{\sqrt{2\pi}\sigma}. \tag{5.2}$$

For a European call option, the initial condition is

$$\omega(\xi, 0) = \max(Ke^{\xi} - K, 0), \tag{5.3}$$

where K is the strike price [19]. We first truncate the infinite ξ-domain $(-\infty, \infty)$ to $[\xi_{\min}, \xi_{\max}]$ and then divide $[\xi_{\min}, \xi_{\max}]$ into $n+1$ subintervals with a uniform mesh size Δ_ξ. By approximating the differential part of (5.1) by central difference discretization, we obtain an $n \times n$ tridiagonal Toeplitz matrix

$$\mathcal{D}_n = \text{tridiag}\left[\frac{\nu^2}{2\Delta_\xi^2} - \frac{2r - 2\lambda\kappa - \nu^2}{4\Delta_\xi},\ -\frac{\nu^2}{\Delta_\xi^2} - r - \lambda,\ \frac{\nu^2}{2\Delta_\xi^2} + \frac{2r - 2\lambda\kappa - \nu^2}{4\Delta_\xi}\right].$$

For the integral term in (5.1), the localized part can be expressed in discrete form by using the rectangle rule. The corresponding operator is an $n \times n$ Toeplitz matrix

$$\mathcal{I}_n = \Delta_\xi \begin{bmatrix} \phi(0) & \phi(\Delta_\xi) & \cdots & \phi((n-2)\Delta_\xi) & \phi((n-1)\Delta_\xi) \\ \phi(-\Delta_\xi) & \phi(0) & \phi(\Delta_\xi) & \cdots & \phi((n-2)\Delta_\xi) \\ \vdots & \phi(-\Delta_\xi) & \phi(0) & \ddots & \vdots \\ \phi((2-n)\Delta_\xi) & \cdots & \ddots & \ddots & \phi(\Delta_\xi) \\ \phi((1-n)\Delta_\xi) & \phi((2-n)\Delta_\xi) & \cdots & \phi(-\Delta_\xi) & \phi(0) \end{bmatrix}.$$

Let $A_n = \mathcal{D}_n + \lambda\mathcal{I}_n$ be the real nonsymmetric Toeplitz matrix. Then A_n is the coefficient matrix of the semidiscretized system with regard to t [32]. The option price at $t = T$ requires evaluating the exponential term $\exp(TA_n)\omega_0$, where ω_0 is the discretized form of the initial value in (5.3); see [32] for details.

Here we first prove that Example 3 also satisfies (2.4) under the condition of $r > 0$. We consider the consistent problem $\exp(-T(-A_n))\omega_0$ and the corresponding generating function of $-A_n = -\mathcal{D}_n - \lambda\mathcal{I}_n$. For simplicity, we let

$$a = \nu^2 \quad \text{and} \quad b = r - \lambda\kappa - \frac{\nu^2}{2}.$$

It is easy to find that the generating function of the differential operator $\mathcal{D}_n$ is

$$f_{diff} = -\frac{a}{\Delta_\xi^2}(1 - \cos\theta) + i\frac{b}{\Delta_\xi}\sin\theta - r - \lambda.$$

Let f_{int} denote the generating function of the integral part $\mathcal{I}_n$. Thus we know that the generating function of $-A_n$ is

$$f = -f_{diff} - \lambda f_{int} = \frac{a}{\Delta_\xi^2}(1 - \cos\theta) - i\frac{b}{\Delta_\xi}\sin\theta + r + \lambda - \lambda f_{int}.$$

Note that the density function $\phi(\eta)$ (5.2) is a Gaussian distribution; we have

$$\phi(\eta) \geq 0 \quad \text{and} \quad \int_{-\infty}^{\infty} \phi(\eta)d\eta = 1.$$

Then we can directly deduce that for a large n, the real and imaginary parts of f_{int} satisfy

$$|Re(f_{int})| \leq 1 \quad \text{and} \quad |Im(f_{int})| \leq 1.$$

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

$$Re(f) = \frac{a}{\Delta_\xi^2}(1 - \cos\theta) + r + \lambda[1 - Re(f_{int})] \geq 0,$$

TABLE 5.2
The numbers of iterations of the shift-invert Arnoldi method and the standard Arnoldi method in Example 3.

n	$T = 0.5$				$T = 1$			
	$tol = 10^{-4}$		$tol = 10^{-7}$		$tol = 10^{-4}$		$tol = 10^{-7}$	
	shft-inv	stdrd	shft-inv	stdrd	shft-inv	stdrd	shft-inv	stdrd
256	9	44	17	62	10	65	17	88
512	10	88	17	122	10	128	18	175
1024	10	174	17	242	10	-	18	-
2048	10	-	17	-	10	-	18	-

and with $r > 0$,

$$\left|\frac{Im(f)}{Re(f)}\right| = \frac{|-b\Delta_\xi \sin\theta - \lambda\Delta_\xi^2 Im(f_{int})|}{a(1-\cos\theta) + r\Delta_\xi^2 + \lambda\Delta_\xi^2[1 - Re(f_{int})]}$$
$$\leq \frac{b^2\sin^2\theta + (1+2\lambda)\Delta_\xi^2}{4a\sin^2\frac{\theta}{2} + 2r\Delta_\xi^2} = \mathcal{O}(1).$$

Therefore, Example 3 also meets the condition (2.4), and the numerical results will show that $\|tA_n\|_2$ does not interfere with the iteration numbers.

The input parameters are $\xi_{\min} = -2$, $\xi_{\max} = 2$, $K = 100$, $\nu = 0.25$, $r = 0.05$, $\lambda = 0.1$, $\mu = -0.9$, and $\sigma = 0.45$. The shift parameter is selected as $\gamma = T/10$ just like before, and in fact γ also is not sensitive to convergence in this example. Note that $\|A_n\|_2$ increases with n as we refine grid nodes in the spatial direction. Therefore, we use various matrix size n to tell the effectiveness of the shift-invert Arnoldi method. In Table 5.2, numerical results show that the shift-invert Arnoldi method outperforms the standard one, and the error bound is independent of $\|A_n\|_2$.

Apart from showing the iteration behavior of the two methods, we now continue to cover some other numerical aspects of approximating the matrix exponential. In the implementation of the shift-invert Arnoldi method, it is common to first find the inverse $(I + \gamma A_n)^{-1}$ before going into the iterative process. For instance, an LU decomposition would be a natural choice for a general matrix [24]. Then every term $(I + \gamma A_n)^{-1}v_j$ is computed in each iteration by solving triangular systems. However, the factorization of a dense matrix costs $\mathcal{O}(n^3)$ operations, and also, those triangular systems require $\mathcal{O}(n^2)$ operations to solve. For the TME case, the GSF (2.5) takes the upper hand in finding the inverse of a Toeplitz matrix, and it needs to be done only for once and for all. The two Toeplitz systems in (4.1) can be solved by the GMRES method with T. Chan's preconditioner. In the iterative process, $(I + \gamma A_n)^{-1}v_j$ is computed by FFTs with $\mathcal{O}(n \log n)$ complexity.

Table 5.3 contains the numerical results of Example 3 with $T = 1$. This time we report the CPU times (in seconds) of the standard Arnoldi method and the shift-invert Arnoldi method with GSF or LU decomposition to reach the final accuracy of 10^{-4} and 10^{-7}. It is easy to see that the shift-invert Arnoldi method with GSF is less time-consuming. The standard Arnoldi method is plagued by the heavy iteration numbers and happens to be the worst among them, even worse than the costly shift-invert Arnoldi method with LU decomposition. The difference in CPU times is more obvious when the matrix size n grows larger.

Previously we have clarified the different computational costs of the shift-invert Arnoldi method and the standard Arnoldi method. The conclusion is that we have to pay a higher price to run the shift-invert Arnoldi method, but hopefully its smaller

TABLE 5.3
CPU times (in seconds) of the standard Arnoldi method and the shift-invert Arnoldi method with GSF or LU decomposition in Example 3 with $T = 1$.

n	$tol = 10^{-4}$			$tol = 10^{-7}$		
	stdrd	shft-inv		stdrd	shft-inv	
		LU	GSF		LU	GSF
256	0.0238	0.0191	0.0126	0.0380	0.0265	0.0182
512	0.0970	0.0804	0.0187	0.1718	0.0985	0.0263
1024	0.6446	0.4258	0.0331	3.0102	0.4932	0.0466
2048	14.4769	1.9983	0.0821	33.8326	2.2678	0.1016

TABLE 5.4
CPU times (in seconds) of the standard Arnoldi method and the shift-invert Arnoldi method using time subdivision approach in Example 3 with $T = 1$.

n	Δ_t	$tol = 10^{-4}$		$tol = 10^{-7}$	
		stdrd	shft-inv	stdrd	shft-inv
512	0.05	0.0784	0.0649	0.1371	0.0755
	0.1	0.0772	0.0414	0.1227	0.0541
	0.5	0.0946	0.0280	0.1588	0.0358
	1	0.0970	0.0261	0.1718	0.0347
1024	0.05	0.3504	0.1124	0.5398	0.1398
	0.1	0.3475	0.0714	0.5370	0.1044
	0.5	0.4186	0.0541	0.6934	0.0670
	1	0.6446	0.0523	3.0102	0.0659
2048	0.05	1.2313	0.2552	2.2561	0.2987
	0.1	1.4411	0.1417	2.4366	0.2125
	0.5	4.1286	0.0904	14.2003	0.1117
	1	14.4769	0.0821	33.8326	0.1016

iteration number would help unload the weight on its back. The results in Table 5.3 indeed verify this conclusion, and the standard Arnoldi method seems to fall far behind. In fact, Popolizio and Simoncini [26] showed that the standard Lanczos method for approximating a symmetric matrix exponential can be improved significantly by chopping up the time direction:

$$\exp(TA_n)\omega_0 = \overbrace{\exp(\Delta_t A_n)\cdots\exp(\Delta_t A_n)}^{k}\,\omega_0, \qquad \Delta_t = T/k.$$

This time subdivision approach is also applicable in the standard Arnoldi method and would very likely enhance the method. Therefore, we take this into account in the following experiments to make sure the standard Arnoldi method and the shift-invert Arnoldi method are compared in their best shapes. Once again we evaluate their overall performance in terms of CPU time used, with time subdivision this time. We will try four kinds of time subdivision, which is analogous to [26], for the standard Arnoldi method as well as for the shift-invert Arnoldi method. The step sizes are chosen as $\Delta_t = T = 1$, $\Delta_t = 0.5$, $\Delta_t = 0.1$, and $\Delta_t = 0.05$ to approximate the exponential at $T = 1$ in Example 3.

In Table 5.4, we display the CPU times used by the two methods in order to reach the tolerance of 10^{-4} and 10^{-7}. The numerical results show that the standard Arnoldi method is better with a time subdivision. When the matrix size is as large as $n = 2048$, the standard Arnoldi method with $\Delta_t = 0.05$ is approximately 15 times faster than the original approach without any subdivision. On the contrary, the shift-

invert Arnoldi method apparently is not suitable to apply the time subdivision, and in fact it is more competent with larger time steps. Nevertheless, we can observe that even the best record of the time-subdivided standard Arnoldi method still is topped by the shift-invert Arnoldi method. That means all the hustle and bustle of implementing the shift-invert Arnoldi method is worthwhile after all.

6. Concluding remarks. In this paper, we have employed the shift-invert Arnoldi method to compute the real TME. We show that under the two assumptions in (2.4), the real Toeplitz matrix A_n is a sectorial operator, and hence the error bound of the shift-invert Arnoldi approximation is independent of $\|tA_n\|_2$. Moreover, we have reduced the computational costs to $\mathcal{O}(n \log n)$ by exploiting the Toeplitz structure. Several numerical examples, including an application in computational finance, illustrate that the shift-invert Arnoldi method needs far fewer iterations and is unaffected by the change of $\|tA_n\|_2$.

Finally we remark that if A_n is not an exact Toeplitz matrix, e.g., it is a block Toeplitz Toeplitz block matrix in the 2D case, there will not be any efficient inversion formula, just like the GSF for standard Toeplitz matrices. Accordingly in the shift-invert Arnoldi process, it is inevitable to solve a Toeplitz-like system at each iteration step. In future work, iterative methods would be studied for solving such Toeplitz-like systems instead of using direct representation from the matrix inversion formula.

Acknowledgments. The authors would like to thank Raymond H. Chan and Tao Wu for introducing this topic to us, Igor Moret for his useful comments, and Eugene Tyrtyshnikov for his helpful suggestions. The authors also are grateful to the anonymous referees for the reminder of Toeplitz-related references and pointing out several numerical aspects of the matrix exponential approximation.

REFERENCES

[1] M. Abdou and A. Badr, *On a method for solving an integral equation in the displacement contact problem*, Appl. Math. Comput., 127 (2002), pp. 65–78.
[2] G. Ammar and W. Gragg, *Superfast solution of real positive definite Toeplitz systems*, SIAM J. Matrix Anal. Appl., 9 (1988), pp. 61–76.
[3] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, 2000.
[4] L. Bergamaschi and M. Vianello, *Efficient computation of the exponential operator for large, sparse, symmetric matrices*, Numer. Linear Algebra Appl., 7 (2000), pp. 27–45.
[5] R. Chan and X. Jin, *An Introduction to Iterative Toeplitz Solvers*, SIAM, Philadelphia, 2007.
[6] R. Chan and M. Ng, *Conjugate gradient methods for Toeplitz systems*, SIAM Rev., 38 (1996), pp. 427–482.
[7] R. Chan and M. Yeung, *Circulant preconditioners for complex Toeplitz matrices*, SIAM J. Numer. Anal., 30 (1993), pp. 1193–1207.
[8] F. Diele, I. Moret, and S. Ragni, *Error estimates for polynomial Krylov approximations to matrix functions*, SIAM. J. Matrix Anal. Appl., 30 (2008), pp. 1546–1565.
[9] M. Eiermann and O. Ernst, *A restarted Krylov subspace method for the evaluation of matrix functions*, SIAM J. Numer. Anal., 44 (2006), pp. 2481–2504.
[10] J. van den Eshof and M. Hochbruck, *Preconditioning Lanczos approximations to the matrix exponential*, SIAM J. Sci. Comput., 27 (2006), pp. 1438–1457.
[11] A. Frommer and V. Simoncini, *Stopping criteria for rational matrix functions of Hermitian and symmetric matrices*, SIAM J. Sci. Comput., 30 (2008), pp. 1387–1412.
[12] E. Gallopoulos and Y. Saad, *Efficient solution of parabolic equations by Krylov approximation methods*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 1236–1264.
[13] I. Gohberg and A. Semencul, *On the inversion of finite Toeplitz matrices and their continuous analogs*, Mat. Issled., 2 (1972), pp. 201–233.
[14] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed., The John Hopkins University Press, Baltimore, 1996.

[15] G. Heinig and K. Rost, *Algebraic Methods for Toeplitz-Like Matrices and Operators*, Birkhäuser Verlag, Basel, 1984.
[16] N. Higham, *The scaling and squaring method for the matrix exponential revisited*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 1179–1193.
[17] M. Hochbruck and C. Lubich, *On Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 34 (1997), pp. 1911–1925.
[18] L. Lopez and V. Simoncini, *Analysis of projection methods for rational function approximation to the matrix exponential*, SIAM J. Numer. Anal., 44 (2006), pp. 613–635.
[19] R. Merton, *Option pricing when underlying stock returns are discontinuous*, J. Financ. Econ., 3 (1976), pp. 125–144.
[20] C. Moler and C. Van Loan, *Nineteen dubious ways to compute the exponential of a matrix*, SIAM Rev., 20 (1978), pp. 801–836.
[21] C. Moler and C. Van Loan, *Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later*, SIAM Rev., 45 (2003), pp. 3–49.
[22] I. Moret, *On RD-rational Krylov approximations to the core-functions of exponential integrators*, Numer. Linear Algebra Appl., 14 (2007), pp. 445–457.
[23] I. Moret and P. Novati, *An interpolatory approximation of the matrix exponential based on Faber polynomials*, J. Comput. Appl. Math., 131 (2001), pp. 361–380.
[24] I. Moret and P. Novati, *RD-rational approximations of the matrix exponential*, BIT, 44 (2004), pp. 595–615.
[25] M. Ng, H. Sun, and X. Jin, *Recursive-based PCG methods for Toeplitz systems with nonnegative generating functions*, SIAM J. Sci. Comput., 24 (2003), pp. 1507–1529.
[26] M. Popolizio and V. Simoncini, *Acceleration techniques for approximating the matrix exponential operator*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 657–683.
[27] R. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
[28] Y. Saad, *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 29 (1992), pp. 209–228.
[29] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.
[30] S. Serra Capizzano and P. Tilli, *Extreme singular values and eigenvalues of non-Hermitian block Toeplitz matrices*, J. Comput. Appl. Math., 108 (1999), pp. 113–130.
[31] V. Simoncini and D. Szyld, *Theory of inexact Krylov subspace methods and applications to scientific computing*, SIAM J. Sci. Comput., 25 (2003), pp. 454–477.
[32] D. Tangman, A. Gopaul, and M. Bhuruth, *Exponential time integration and Chebychev discretisation schemes for fast pricing of options*, Appl. Numer. Math., 58 (2008), pp. 1309–1319.
[33] P. Tilli, *Singular values and eigenvalues of non-Hermitian block Toeplitz matrices*, Linear Algebra Appl., 272 (1998), pp. 59–89.