Numerische Mathematik (2019) 141:319–351
https://doi.org/10.1007/s00211-018-1006-y

Numerische Mathematik

On the exponential of semi-infinite quasi-Toeplitz matrices

Dario A. Bini[1] · **Beatrice Meini**[1]

Received: 24 September 2017 / Revised: 28 September 2018 / Published online: 31 October 2018

Abstract
Let $a(z) = \sum_{i\in\mathbb{Z}} a_i z^i$ be a complex valued function defined for $|z| = 1$, such that $\sum_{i\in\mathbb{Z}} |a_i| < \infty$; define $T(a) = (t_{i,j})_{i,j\in\mathbb{Z}^+}$, $t_{i,j} = a_{j-i}$ for $i, j \in \mathbb{Z}^+$, the semi-infinite Toeplitz matrix associated with the symbol $a(z)$; let $E = (e_{i,j})_{i,j\in\mathbb{Z}^+}$ be a compact operator in ℓ^p, with $1 \le p \le \infty$. A semi-infinite matrix of the kind $A = T(a) + E$ is said quasi-Toeplitz (QT). The problem of the computation of $\exp(A)$ or $\exp(A)v$, with A quasi-Toeplitz and v a vector, arises in many applications. We prove that the exponential of a QT-matrix A is QT, that is, $\exp(A) = T(\exp(a)) + F$ where F is a compact operator in ℓ^p. This property allows the design of an algorithm for computing $\exp(A)$ and $\exp(A)v$ up to any precision. The case of families of $n \times n$ matrices obtained by truncating infinite QT-matrices to finite size is also considered. Numerical experiments show the effectiveness of this approach.

Mathematics Subject Classification 65F60 · 15A16 · 15B05 · 47B35

1 Introduction

Let $\mathbb{T} = \{z \in \mathbb{C} : |z| = 1\}$ be the complex unit circle and denote by $\mathcal{W}$ the *Wiener algebra*, that is, the set of all functions $a(z) : \mathbb{T} \to \mathbb{C}$ of the form $a(z) = \sum_{i\in\mathbb{Z}} a_i z^i$ with $\sum_{i\in\mathbb{Z}} |a_i| < \infty$, endowed with the norm $\|a\|_{\mathcal{W}} = \sum_{i\in\mathbb{Z}} |a_i|$.

Given $a(z) \in \mathcal{W}$, let $T(a) = (t_{i,j})_{i,j\in\mathbb{Z}^+}$ be the semi-infinite Toeplitz matrix, associated with the symbol $a(z)$, such that $t_{i,j} = a_{j-i}$ for i, j in the set $\mathbb{Z}^+$ of positive integers. Denote by $T_n(a)$ the $n \times n$ leading principal submatrix of $T(a)$. Let

The research was carried out with the support of GNCS of INdAM.

✉ Dario A. Bini
dario.bini@unipi.it

Beatrice Meini
beatrice.meini@unipi.it

[1] Dipartimento di Matematica, Università di Pisa, Largo Bruno Pontecorvo 5, 56127 Pisa, Italy

$E = (e_{i,j})_{i,j\in\mathbb{Z}^+}$ be a compact operator acting on sequences $(x_i) \in \ell^p$ for $1 \le p \le \infty$, that is $\sum_{i\in\mathbb{Z}^+} |x_i|^p < \infty$ for $p < \infty$ and $\sup_{i\in\mathbb{Z}^+} |x_i| < \infty$ for $p = \infty$.

In this paper, we analyze the problem of computing the matrix exponential of semi-infinite matrices of the form $A = T(a) + E$ and of families of finite matrices $A_n = T_n(a) + E_n$, where E_n is the leading principal submatrix of E. We refer to this class of matrices as quasi-Toeplitz, in short, QT-matrices [4].

The computation of $\exp(T_n(a))$ has been recently considered by Kressner and Luce [19] for a finite Toeplitz matrix, motivated by several applications where this problem is encountered, in particular, the numerical solution of parabolic equations, and the Merton problem encountered in financial models. Other applications of the matrix exponential of finite (block) Toeplitz matrices which concern the Erlangian approximation of Markovian fluid queues are analyzed in [2,9].

Here we are also interested in the case of semi-infinite QT-matrices since they model a wide range of queuing problems involving a denumerable set of states where the computation of important quantities of interest, like the transient distribution and the occupation period duration, is reduced to computing $\exp(A)v$ for a given vector v and an infinite QT-matrix $A = T(a) + E$. For instance, in the context of Markov chains, if Q is the generator matrix of a continuous-time Markov process, then the probability distribution $\pi(t)$ at time $t \ge 0$ is given by $\pi(t)^T = \pi(0)^T \exp(Qt)$, where $\pi(0)$ is the initial distribution [28, Chapter 2]. Similarly, the distribution of the occupation period duration is obtained by the vector $\exp(Q_B t)e$, where e is the vector of all ones and Q_B is a matrix obtained by removing a suitable number of leading rows and columns of Q [28, Chapter 5]. In many problems of interest like random walks on the integer numbers or on the quarter plane [10,30], or in Birth-and-Death processes [21,24], and in tandem Jackson queues [23,27], the generator Q is a QT-matrix.

The problem of computing $f(A)v$, where A is a linear operator has been investigated in [13] for a generic function f, in [17] for the exponential function relying on Krylov subspaces, and in [29] relying on the finite section technique. The case where $f(z)$ is analytic and A is a Toeplitz matrix is considered in [3] relying on contour integration.

Our aim is to provide effective numerical algorithms for computing the matrix exponential of a QT-matrix A which exploit the specific structure of A. In particular, if $A = T(a) + E$, we show that $\exp(A)$ can be represented in the form

$$\exp(A) = T(\exp(a)) + F,$$

where $F = (f_{i,j})$ is a compact operator in ℓ^p. Moreover, if $\sum_{i\in\mathbb{Z}} |ia_i| < \infty$ and $\sum_{i,j\in\mathbb{Z}^+} |e_{i,j}| < \infty$, then also $\sum_{i,j\in\mathbb{Z}^+} |f_{i,j}| < \infty$. We prove this by showing that the set of QT-matrices is a Banach algebra either with respect to the norm $\|A\| = \alpha\|a\|_{\mathcal{W}} + \|E\|_p$, for $\alpha = \frac{1}{2}(1+\sqrt{5})$, or with the norm $\|A\| = \|a\|_{\mathcal{W}} + \|a'\|_{\mathcal{W}} + \sum_{i,j\in\mathbb{Z}^+} |e_{i,j}|$. This property allows us to represent a QT-matrix by storing the Toeplitz part and the correction part and to operate with them separately. In fact, the decay properties of the coefficients a_i and of the entries $e_{i,j}$, if $1 < p < \infty$, enable us to approximate a QT-matrix to arbitrary precision by means of a finite number of parameters.

We provide an algorithm for computing both the Toeplitz part $T(\exp(a))$ and the compact correction F of $\exp(A)$. The algorithm relies on the power series representation of the matrix exponential $\exp(A) = \sum_{k=0}^{\infty} \frac{1}{k!} A^k$ and on a suitable recurrence

equation which relates E_{i+1} to E_i where $A^i = T(a^i) + E_i$. For approximating the matrix exponential we used the Taylor expansion instead of Padé approximation, since an approach based on rational approximation involves matrix inversion that, for QT-matrices would have a higher computational cost. In order to reduce the number of terms in the Taylor expansion we used the classical technique of scaling and squaring [16].

The decay to zero of the coefficients f_i of $f(z) = \sum_{i\in\mathbb{Z}} f_i z^i = \exp(a(z))$ is analyzed and upper bounds for the numerical bandwidth b of $T(\exp(a))$, that is, the minimum integer b such that $|f_i| \leq \epsilon \max_j |f_j|$ for $i > b$, are given, where ϵ is a given positive number, say the machine precision. *A priori* upper bounds for the numerical rank r, to the numerical size n and to the norm of the correction F are also provided. Here, for numerical rank of a matrix A we mean the minimum rank of B such that $\|A - B\| < \epsilon$ for a given norm $\|\cdot\|$. By numerical size of F we mean the minimum value of n such that $|f_{i,j}| < \epsilon \max_{p,q} |f_{p,q}|$ for $i, j > n$.

The complexity of our algorithm for infinite matrices, in terms of arithmetic operations, is proportional to $r(n+b)\log(n+b) + r^2 n$, times the overall number of terms in the Taylor expansion and the number of squaring steps. The same algorithm can be adjusted to the case of finite $N \times N$ matrices. Clearly, if $N > n + b$, the complexity is independent of the size N. This property makes the algorithm very effective in large part of the cases and more convenient than the approach of [19], which relies on the displacement representation of Toeplitz-like matrices and of [22,26,32], where the complexity depends on the size N of the matrix. It is important to point out that, once the representation $\exp(A) = T(\exp(a)) + F$ has been computed, the product $y = \exp(T(a))v$ can be written as $y = T(\exp(a))v + Fv$. The first term is a convolution which can be easily computed by means of FFT, and the second term is reduced to computing the product of a finite matrix of size n, of rank r, and a vector. The overall complexity of this multiplication is $O(M \log M + rn)$ operations where M is the maximum of the numerical size of the vector v and the number of computed components of y.

The algorithms for computing $\exp(A)$ and $\exp(A)v$, where A is a semi-infinite or finite QT-matrix, have been implemented in Matlab and tested with several problems. These tests include infinite generators from queueing models typically having a Hessenberg structure, the $n \times n$ finite differences second derivative matrix in the form $\frac{\Delta_t}{\Delta_x^2}\mathrm{trid}_n(1, -2, 1)$ encountered in the discretization of the heat equation, and the Merton problem. Comparisons have been performed with the currently most advanced toolbox of [19] for computing the exponential of a finite Toeplitz matrix, and with the toolbox of Al-Mohy and Higham [1] for computing the product $\exp(A)v$ for a sparse matrix A, available from https://github.com/higham/expmv. From the numerical experiments that we have performed, it turns out that the algorithms based on QT-matrices are much faster than the algorithms of [1,19], and in all cases they are numerically very reliable. When applied to finite matrices, the effectiveness of our algorithm is evident in the cases where the size N of the matrix is larger than the numerical bandwidth b of $T(\exp(a))$ and the numerical size n of F. However, our algorithm still provides fast computations and accurate results if the size N is smaller than b and n, like in the case of the Merton problem.

The paper is organized as follows. In Sect. 2 we recall some preliminary results concerning semi-infinite Toeplitz matrices. In Sect. 3 we analyze the properties of $\exp(T(a))$ and show that $\exp(T(a)) = T(\exp(a)) + F$, where F is compact. In Sect. 4 we perform the decay analysis of the coefficients of the function $\exp(a)$ and give bounds for the numerical size and numerical rank of the correction F. Section 5 extends the results of Sect. 3 to $\exp(A)$ where $A = T(a)+E$, with $E \neq 0$. In Sect. 6 we describe in detail the algorithm for computing the exponential of a Toeplitz matrix and we outline the case of a general QT-matrix. Section 7 reports the results of extensive numerical experimentation.

2 Preliminaries

We recall the basic definition of the matrix exponential of an $n \times n$ matrix and the main properties of semi-infinite Toeplitz matrices which will be used in our analysis.

For an $n \times n$ matrix A, it is well known that the series $\exp(A) := \sum_{i=0}^{+\infty} \frac{1}{i!} A^i$ is convergent and defines the matrix exponential of A. We refer to the book by N. Higham [16] for the concept of matrix function and for more details on the matrix exponential. Indeed, defining the partial sum

$$S_k = \sum_{i=0}^{k} \frac{1}{i!} A^i, \tag{1}$$

and the remainder R_k of the series as $R_k = \sum_{i=k+1}^{\infty} \frac{1}{i!} A^i$, for any matrix norm $\|\cdot\|$ such that $\|A^2\| \leq \|A\|^2$ it follows that

$$\|R_k\| = \left\| \sum_{i=k+1}^{\infty} \frac{1}{i!} A^i \right\| \leq \sum_{i=k+1}^{\infty} \frac{1}{i!} \|A\|^i \tag{2}$$

so that $\lim_{k\to\infty} \|R_k\| = 0$ which implies the convergence of the sequence S_k.

This property is still valid if $A = (a_{i,j})_{i,j\in\mathbb{Z}^+}$ is a semi-infinite matrix provided that A belongs to a Banach algebra $\mathcal{A}$, that is an algebra endowed with a sub-multiplicative norm $\|\cdot\|$, such that $\|AB\| \leq \|A\| \cdot \|B\|$ for any $A, B \in \mathcal{A}$, which makes it a Banach space. Indeed, for $A \in \mathcal{A}$, consider the sequence $\{S_k\}_k$ defined in (1). For $i > j$ we have

$$\|S_i - S_j\| \leq \sum_{h=j+1}^{i} \frac{1}{h!} \|A^h\| \leq \sum_{h=j+1}^{i} \frac{1}{h!} \|A\|^h.$$

From this bound it follows that for any $\epsilon > 0$ there exists $k > 0$ such that $\|S_i - S_j\| \leq \epsilon$ for any $i > j \geq k$. That is, $\{S_k\}_k$ is a Cauchy sequence. Since by definition of Banach space, the Cauchy sequences in $\mathcal{A}$ have a limit in $\mathcal{A}$, there exists a matrix $L \in \mathcal{A}$ such that $\lim_{k\to\infty} \|S_k - L\| = 0$. We let $L = \exp(A)$.

We now recall some results concerning infinite Toeplitz matrices. For more details on this topic we refer the reader to the book by Böttcher and Grudsky [7].

Let $\mathcal{W} = \{a(z) = \sum_{i\in\mathbb{Z}} a_i z^i : \sum_{i\in\mathbb{Z}} |a_i| < +\infty\}$ denote the Wiener algebra formed by Laurent power series, defined on the unit circle $\mathbb{T} = \{z \in \mathbb{C} : |z| = 1\}$, such that the sum of the moduli of their coefficients is finite. It is well known that $\mathcal{W}$ endowed with the norm $\|a\|_{\mathcal{W}} = \sum_{i\in\mathbb{Z}} |a_i|$ is a Banach algebra. For $a(z) \in \mathcal{W}$ let $T(a)$ denote the semi-infinite Toeplitz matrix whose entries $t_{i,j}$ are such that $t_{i,j} = a_{j-i}$ for $i, j \in \mathbb{Z}^+$, where $\mathbb{Z}^+$ denotes the set of positive integers. Also let $a_+(z)$ and $a_-(z)$ be the power series defined by $a_+(z) = \sum_{i\in\mathbb{Z}^+} a_i z^i$ and $a_-(z) = \sum_{i\in\mathbb{Z}^+} a_{-i} z^i$ so that $a(z) = a_0 + a_+(z) + a_-(z^{-1})$. Finally, given the power series $b(z) = \sum_{i\in\mathbb{Z}^+} b_i z^i$ define $H(b) = (h_{i,j})$ the Hankel matrix such that $h_{i,j} = b_{i+j-1}$, for $i, j \in \mathbb{Z}^+$.

Any semi-infinite matrix $S = (s_{i,j})_{i,j\in\mathbb{Z}^+}$ can be viewed as a linear operator, acting on semi-infinite vectors $v = (v_i)_{i\in\mathbb{Z}^+}$, which maps the vector v onto the vector u such that $u_i = \sum_{j\in\mathbb{Z}^+} s_{i,j} v_j$, provided that the summations are finite. For any $p \geq 1$, included $p = \infty$, we may define the Banach space ℓ^p formed by all the semi-infinite vectors $v = (v_i)_{i\in\mathbb{Z}^+}$ such that $\|v\|_p = (\sum_{i\in\mathbb{Z}^+} |v_i|^p)^{\frac{1}{p}} < \infty$, where for $p = \infty$ we have $\|v\|_\infty = \sup_{i\in\mathbb{Z}^+} |v_i|$. It is well known that these norms induce the corresponding operator norms $\|S\|_p = \sup_{\|v\|_p=1} \|Sv\|_p$ which are sub-multiplicative, i.e., $\|AB\|_p \leq \|A\|_p \|B\|_p$ for any semi-infinite matrices A, B having finite norm, and that the linear space formed by the latter semi-infinite matrices forms a Banach algebra. We denote by L^p the set of linear operators with finite operator norm induced by the ℓ^p norm.

We may wonder if the matrices $T(a)$, $H(a_+)$ and $H(a_-)$ define bounded linear operators acting on the Banach space ℓ^p. The answer to this question is given by the following result of [7] which relates the matrix $T(a)T(b)$ to $T(ab)$, $H(a_-)$ and $H(a_+)$, see Propositions 1.2 and 1.3 in [7].

Theorem 1 *For $a(z), b(z) \in \mathcal{W}$ let $c(z) = a(z)b(z)$. Then we have*

$$T(a)T(b) = T(c) - H(a_-)H(b_+).$$

Moreover, for any $p \geq 1$, including $p = \infty$, we have

$$\|T(a)\|_p \leq \|a\|_{\mathcal{W}}, \quad \|H(a_-)\|_p \leq \|a_-\|_{\mathcal{W}}, \quad \|H(b_+)\|_p \leq \|b_+\|_{\mathcal{W}},$$

and the matrices $H(a_-)$ and $H(b_+)$ define compact operators in ℓ^p.

The above result implies that the product of two Toeplitz matrices can be written as a Toeplitz matrix plus a correction whose ℓ^p operator norm is bounded by $\|a_-\|_{\mathcal{W}}\|b_+\|_{\mathcal{W}} \leq \|a\|_{\mathcal{W}}\|b\|_{\mathcal{W}}$. It is observed in [8] that the smallest closed subalgebra of L^2 which contains the set of semi-infinite Toeplitz matrices associated with a continuous symbol is formed by matrices of the kind $A = T(a) + E_a$ where $a(z)$ is continuous and E_a is a compact operator in L^2. In this algebra we have $\|A\|_2 \leq \|a\|_{\mathcal{W}} + \|E_a\|_2$.

We can endow the above set of matrices with different norms that are better suited for numerical computations. In particular, consider matrices of the kind $A = T(a)+E_a$

where $a(z) \in \mathcal{W}$ and E_a is a compact operator in L^p for some $1 \le p \le \infty$, and define $\|A\| := \alpha\|a\|_{\mathcal{W}} + \|E_a\|_p$. It is easy to show that if $\alpha \ge (1+\sqrt{5})/2$ then this norm is submultiplicative. In fact, in view of Theorem 1, if $A = T(a) + E_a$, $B = T(b) + E_b$, $C = AB$, $c(z) = a(z)b(z)$, we have $C = T(c) + E_c$, $E_c = -H(a_-)H(b_+) + T(a)E_b + E_aT(b) + E_aE_b$ so that E_c is compact since the sum of products of operators of which at least a factor is compact, see [20, Theorem 8.3-2], and

$$\begin{aligned}\|AB\| &= \|T(c) + E_c\| = \alpha\|c\|_{\mathcal{W}} + \|E_c\|_p \\ &\le \alpha\|a\|_{\mathcal{W}}\|b\|_{\mathcal{W}} + \|a\|_{\mathcal{W}}\|b\|_{\mathcal{W}} + \|a\|_{\mathcal{W}}\|E_b\|_p + \|E_a\|_p\|b\|_{\mathcal{W}} + \|E_a\|_p\|E_b\|_p \\ &\le (\alpha\|a\|_{\mathcal{W}} + \|E_a\|_p)(\alpha\|b\|_{\mathcal{W}} + \|E_b\|_p),\end{aligned}$$

where the last inequality follows since $\alpha \ge (1+\sqrt{5})/2$.

Thus, the set formed by matrices of the kind $T(a) + E_a$, where $a(z) \in \mathcal{W}$ and E_a is a compact bounded operator in ℓ^p, is a Banach algebra with the norm $\|A\| := \alpha\|a\|_{\mathcal{W}} + \|E_a\|_p$. By choosing for instance $p = 1$ or $p = \infty$, the norm $\|A\|$ can be easily computed.

Under the assumption $a'(z) \in \mathcal{W}$, where $a'(z) = \sum_{i\in\mathbb{Z}} i a_i z^i$ is the first derivative of $a(z)$, we may define a more restrictive norm as $\|A\| := \|a\|_{\mathcal{W}} + \|a'\|_{\mathcal{W}} + \|E_a\|_{\mathcal{F}}$, $\|E_a\|_{\mathcal{F}} = \sum_{i,j\in\mathbb{Z}^+} |(E_a)_{i,j}|$. The set of matrices $A = T(a) + E_a$ where $a'(z) \in \mathcal{W}$ and $\|E_a\|_{\mathcal{F}} < +\infty$, endowed with the above norm, is a Banach algebra. For more details we refer the reader to [4].

Observe that the boundedness of $\|E_a\|_2$ or of $\|E_a\|_{\mathcal{F}}$ implies that for any $\epsilon > 0$ there exists k such that $|(E_a)_{i,j}| \le \epsilon$ for any $i, j > k$. This bound allows us to represent E_a, up to within any given error bound, by using a finite number of parameters. This property does not hold if, say, the 1-norm or the infinity norm is used. In fact setting e, the semi-infinite vectors with components equal to 1 and setting e_1 the semi-infinite vector with zero components except for the first which is equal to 1, it holds that $\|ee_1^T\|_\infty = 1$, $\|e_1e^T\|_1 = 1$.

3 Exponential of a semi-infinite Toeplitz matrix

In this section we study properties of the exponential of a semi-infinite Toeplitz matrix, by relating in particular $\exp(T(a))$ to $T(\exp(a))$.

Let $a \in \mathcal{W}$ and consider the associated semi-infinite Toeplitz matrix $T(a)$. From Theorem 1 and by the monotonicity of the function $\exp(z)$ we have

$$\|\exp(T(a))\|_p \le \sum_{i=0}^{\infty} \frac{1}{i!}\|T(a)\|_p^i = \exp(\|T(a)\|_p) \le \exp(\|a\|_{\mathcal{W}}).$$

Now we will take a closer look at $\exp(T(a))$ and relate it to $T(\exp(a))$. Since $\mathcal{W}$ is a Banach algebra, the exponential function is well defined over $\mathcal{W}$ and we have $\exp(a(z)) = \sum_{i=0}^{+\infty} \frac{1}{i!}a(z)^i$.

We first relate $T(a)^i$ to $T(a^i)$, for $i \geq 2$. From Theorem 1 we may write $T(a)^2 = T(a^2) + E_2$, where $E_2 = -H(a_-)H(a_+)$. For a general $i \geq 0$, define E_i as

$$E_i = T(a)^i - T(a^i), \tag{3}$$

where $E_0 = 0$, $E_1 = 0$. Then we have the following

Theorem 2 *Let $a \in \mathcal{W}$ and let $E_i = T(a)^i - T(a^i)$, for $i \geq 1$. Then*

$$\begin{aligned} &E_i = T(a)E_{i-1} - H(a_-)H((a^{i-1})_+), \quad i \geq 2, \\ &E_1 = 0. \end{aligned} \tag{4}$$

Moreover, for any $i \geq 2$ and any integer $p \geq 1$, including $p = \infty$,

$$\|E_i\|_p \leq 2\|a\|_{\mathcal{W}}^i. \tag{5}$$

Proof From the equation $T(a)^i = T(a)T(a)^{i-1}$ and from Theorem 1 we obtain

$$\begin{aligned} T(a)^i &= T(a)T(a)^{i-1} = T(a)[T(a^{i-1}) + E_{i-1}] \\ &= T(a^i) - H(a_-)H((a^{i-1})_+) + T(a)E_{i-1} \\ &= T(a^i) + E_i, \end{aligned}$$

with $E_i = -H(a_-)H((a^{i-1})_+) + T(a)E_{i-1}$. Whence we deduce recurrence (4). Moreover, taking the ℓ^p-norm in (3) we get the bound (5). □

Now define S_k, F_k and G_k as follows

$$\begin{aligned} S_k &= \sum_{i=0}^{k} \frac{1}{i!} T(a)^i = G_k + F_k, \\ G_k &= \sum_{i=0}^{k} \frac{1}{i!} T(a^i), \quad F_k = \sum_{i=0}^{k} \frac{1}{i!} E_i. \end{aligned}$$

The following result shows that the exponential of a QT-matrix is a QT-matrix:

Theorem 3 *Let $a \in \mathcal{W}$. Then*

$$\exp(T(a)) = T(\exp(a)) + F$$

where $F = \sum_{i=0}^{\infty} \frac{1}{i!} E_i$ is a compact operator in L^p such that

$$\|F\|_p \leq \|\exp(T(a))\|_p + \|T(\exp(a))\|_p \leq 2\exp(\|a\|_{\mathcal{W}}). \tag{6}$$

Proof Observe that $G_k = \sum_{i=0}^{k} \frac{1}{i!} T(a^i) = T(\sum_{i=0}^{k} \frac{1}{i!} a^i)$ is such that $\lim_{k\to\infty} G_k = T(\exp(a))$. Since $\lim_{k\to\infty} S_k = \exp(T(a))$, there exists the limit

$$\lim_{k\to\infty} F_k = \exp(T(a)) - T(\exp(a)) =: F. \tag{7}$$

The bound for the norm holds since $\|\exp(T(a))\|_p \le \exp(\|T(a)\|_p)$, $\|T(a)\|_p \le \|a\|_{\mathcal{W}}$, and $\|\exp(a)\|_{\mathcal{W}} \le \exp(\|a\|_{\mathcal{W}})$. Moreover, since $a(z) \in \mathcal{W}$, $H(a_-)$ and $H(a_+)$ are compact so that also E_k and F_k are compact. The correction F is compact in L^p since it is the limit of compact operators [20, Theorem 8.1-5]. □

If $a'(z) \in \mathcal{W}$, we may prove a similar result, by first providing a bound for $\|E_i\|_{\mathcal{F}}$. To this end, observe that the subset of $\mathcal{W}$ formed by functions $a(z)$ such that $a'(z) \in \mathcal{W}$ is a Banach algebra. This property enables us to prove the following

Lemma 1 *Let $a(z) \in \mathcal{W}$ and $E \in \mathcal{F}$ then $\|T(a)E\|_{\mathcal{F}} \le \|a\|_{\mathcal{W}} \|E\|_{\mathcal{F}}$. Moreover, if $a'(z) \in \mathcal{W}$, then for any $k \ge 1$ we have*

$$\|H(a_-)H((a^{k-1})_+)\|_{\mathcal{F}} \le (k-1)\|a\|_{\mathcal{W}}^{k-2} \|a'\|_{\mathcal{W}}^2.$$

Proof For the first part, let $V = T(a)E$ so that $v_{i,j} = \sum_{r\in\mathbb{Z}^+} a_{r-i} e_{r,j}$. Observe that for any $j, r \in \mathbb{Z}^+$, one has $\sum_{i\in\mathbb{Z}^+} |a_{r-i} e_{r,j}| \le \sum_{k\in\mathbb{Z}} |a_k| \cdot |e_{r,j}| = \|a\|_{\mathcal{W}} |e_{r,j}|$. From this inequality we find that

$$\begin{aligned} \|V\|_{\mathcal{F}} &= \sum_{i,j\in\mathbb{Z}^+} |v_{i,j}| \le \sum_{i,j\in\mathbb{Z}^+} \sum_{r\in\mathbb{Z}^+} |a_{r-i} e_{r,j}| = \sum_{r,j\in\mathbb{Z}^+} \sum_{i\in\mathbb{Z}^+} |a_{r-i} e_{r,j}| \\ &\le \|a\|_{\mathcal{W}} \sum_{r,j\in\mathbb{Z}^+} |e_{r,j}| = \|a\|_{\mathcal{W}} \|E\|_{\mathcal{F}}. \end{aligned}$$

For the second part, we have $\|H(a_-)\|_{\mathcal{F}} = \sum_{i,j,\in\mathbb{Z}^+} |a_{1-i-j}| = \sum_{h\in\mathbb{Z}^+} h|a_{-h}| \le \|a'\|_{\mathcal{W}}$ which is finite since $a'(z) \in \mathcal{W}$. Similarly, $\|H((a^{k-1})_+)\|_{\mathcal{F}} \le \|(a^{k-1})'\|_{\mathcal{W}} < \infty$ since both the functions $a^{k-1}(z)$ and $(a^{k-1}(z))'$ belong to $\mathcal{W}$. Thus, for the matrix product $L_k = H(a_-)H((a^{k-1})_+)$ we find that

$$\|L_k\|_{\mathcal{F}} \le \|H(a_-)\|_{\mathcal{F}} \cdot \|H((a^{k-1})_+\|_{\mathcal{F}} \le \|a'\|_{\mathcal{W}} \|(a^{k-1})'\|_{\mathcal{W}}.$$

Now, since $(a^{k-1}(z))' = (k-1)a^{k-2}(z)a'(z)$, we have

$$\|(a^{k-1})'\|_{\mathcal{W}} \le (k-1)\|a\|_{\mathcal{W}}^{k-2} \|a'\|_{\mathcal{W}}.$$

Thus we get $\|L_k\|_{\mathcal{F}} \le (k-1)\|a\|_{\mathcal{W}}^{k-2} \|a'\|_{\mathcal{W}}^2$. □

From the above result we deduce the following

Theorem 4 *If $a'(z) \in \mathcal{W}$, then*

$$\|E_i\|_{\mathcal{F}} \le \frac{i(i-1)}{2} \|a'\|_{\mathcal{W}}^2 \|a\|_{\mathcal{W}}^{i-2}, \quad i \ge 2.$$

Moreover, $\exp(T(a)) = T(\exp(a)) + F$, *with* F *such that*

$$\|F\|_{\mathcal{F}} \leq \frac{1}{2}\|a'\|_{\mathcal{W}}^2 \exp(\|a\|_{\mathcal{W}}).$$

Proof Taking norms in (4), by Lemma 1 we have

$$\|E_i\|_{\mathcal{F}} \leq \|T(a)E_{i-1}\|_{\mathcal{F}} + (i-1)\|a\|_{\mathcal{W}}^{i-2}\|a'\|_{\mathcal{W}}^2, \quad i \geq 2,$$

where $E_0 = E_1 = 0$. By applying again Lemma 1 we deduce that

$$\|E_i\|_{\mathcal{F}} \leq \|a\|_{\mathcal{W}}\|E_{i-1}\|_{\mathcal{F}} + (i-1)\|a\|_{\mathcal{W}}^{i-2}\|a'\|_{\mathcal{W}}^2.$$

Therefore, by using an induction argument we arrive at

$$\|E_i\|_{\mathcal{F}} \leq \frac{i(i-1)}{2}\|a'\|_{\mathcal{W}}^2\|a\|_{\mathcal{W}}^{i-2}, \quad i \geq 2,$$

which proves the first bound. This implies that

$$\|F\|_{\mathcal{F}} \leq \sum_{i=0}^{\infty}\frac{1}{i!}\|E_i\|_{\mathcal{F}} \leq \frac{1}{2}\|a'\|_{\mathcal{W}}^2 \exp(\|a\|_{\mathcal{W}}),$$

which completes the proof. □

4 Decay properties and analysis of the correction

The aim of this section is to study decay properties of the entries of $\exp(T(a)) = T(\exp(a)) + F$. These properties will enable us to approximate the series $f(z) = \sum_{i=-\infty}^{+\infty} f_i z^i = \exp(a(z))$ by a Laurent polynomial $\widetilde{f}(z) = \sum_{i=-k_-}^{k_+} f_i z^i$, such that $|f_i| \leq \epsilon \max_j |f_j|$ for $|i| > b = \max\{k_-, k_+\}$, where $\epsilon > 0$ is a given tolerance, and to approximate the correction F by a matrix $\widetilde{F}$ such that $\|F - \widetilde{F}\| \leq \epsilon$, and $\widetilde{f}_{i,j} = 0$ if $i > n$ or $j > n$. We refer to b as the numerical band-width of $T(\exp(a))$, while $r = \operatorname{rank}(\widetilde{F})$, and n are referred to as the numerical rank and the numerical size of the correction F. *A priori* upper bounds for the numerical band-width, numerical rank r, and numerical size n are provided.

More specifically, in the case where $\exp(a(z))$ is analytic in an annulus containing the unit circle, we give upper bounds to $|f_i|$ which provide estimates of their convergence to zero and show that for the function $a(z) = \theta(z^{-1} - 2 + z)$ these bounds are strict for any value of θ. Moreover, we show that the numerical rank and the numerical size of F are strictly related to the decay of $|f_i|$.

We recall that not all the functions in the Wiener class are analytic in an open annulus containing the unit circle, consider for instance $a(z) = \sum_{i\in\mathbb{Z}\setminus\{0\}} \frac{1}{i^2} z^i$. On the other hand, Laurent polynomials are analytic in a neighbourhood of the unit circle so that our analysis covers the class of finite representable QT matrices.

4.1 Decay properties

The analysis of the decay relies on the Cauchy estimate of the coefficients of an analytic function [15, Theorem 4.4c]:

Theorem 5 *Let $h(z) = \sum_{i=-\infty}^{+\infty} h_i z^i$ be a Laurent series convergent in an annulus $r < |z| < R$. Then, for any ρ with $r < \rho < R$, the coefficients satisfy*

$$|h_i| \le \max_{|z|=\rho} |h(z)| \cdot \rho^{-i}, \quad i \in \mathbb{Z}.$$

If $a(z)$ is analytic in an annulus $r < |z| < R$, then $\exp(a(z))$ is analytic as well in the annulus $r < |z| < R$. Therefore, from Theorem 5, the coefficients of $f(z) = \sum_{i=-\infty}^{+\infty} f_i z^i = \exp(a(z))$, which define the Toeplitz part of $\exp(T(a))$, have an exponential decay. Indeed, for any $r < \rho < R$ and for any $k \in \mathbb{Z}$, one has

$$|f_k| \le \max_{|z|=\rho} |\exp(a(z))| \rho^{-k}. \tag{8}$$

In particular, the larger is $R > 1$, the faster is the decay of the coefficients f_k with $k < 0$, while the smaller is $r < 1$, the faster is the decay of the coefficients f_k with $k > 0$.

We may give a more strict bound for the decay with a suitable choice of $\rho \in (r, R)$ by minimizing the right-hand side of (8). A similar argument has been applied in [18], to estimate the decay of the exponential of banded matrices. In this latter case, since $a(z)$ is a Laurent polynomial, $R > r > 0$ can take any value.

Let $z = \rho(\cos t + \underline{i} \sin t)$, where $\underline{i}$ is the complex unit such that $\underline{i}^2 = -1$. For simplicity, consider the case where $a_i \in \mathbb{R}$ and observe that

$$\exp(a(z)) = \exp\left(\sum_{j\in\mathbb{Z}} a_j \rho^j \cos jt\right) \exp\left(\underline{i} \sum_{j\in\mathbb{Z}} a_j \rho^j \sin jt\right).$$

Since the second factor has modulus 1, we get $|\exp(a(z))| = \exp(\sum_{j\in\mathbb{Z}} a_j \rho^j \cos jt)$ $\le \exp(a_0 + \sum_{j\ne 0} |a_j| \rho^j)$. Let us define $v(\rho) = \sum_{j\ne 0} |a_j| \rho^j$, so that we may write

$$\rho^{-k} \max_{|z|=\rho} |\exp(a(z))| \le \rho^{-k} \exp(a_0) \exp(v(\rho)).$$

Therefore, we may minimize the bound by computing the zeros of the derivative of $\gamma(\rho) = \rho^{-k} \exp(v(\rho))$. From the condition $\gamma'(\rho) = 0$ we get the equation $\rho v'(\rho) = k$, which allows us to provide bounds for the decay of the coefficients which are better than the simple exponential obtained directly by (8).

We show this by means of a simple example which will be considered again in Sect. 7 on numerical experiments.

Consider the matrix $A = T(a)$, where $a(z) = \theta(z^{-1} - 2 + z)$. For $\theta = (n+1)^2$, the $n \times n$ section of A coincides with the finite difference approximation of the second

Table 1 Upper bounds for the numerical bandwidth of the infinite matrix $\exp(A)$ for $A = \theta T(a)$, $a(z) = z^{-1} - 2 + z$, obtained by means of the analysis based on the Cauchy theorem. These values coincide with the actual values reported in Table 9 obtained by means of the numerical computation

θ	513	1025	2049	4097	8193	16,385	32,768
Bandwidth	273	385	544	769	1088	1538	2174

derivative over a grid of width $\Delta_x = 1/(n+1)$. We have $v'(\rho) = \theta(1 - 1/\rho^2)$, so that the optimal value of ρ solves the equation $\rho^2 - \frac{k}{\theta}\rho - 1 = 0$, having the positive solution $\rho_1(k) = \frac{k}{2\theta} + \sqrt{\left(\frac{k}{2\theta}\right)^2 + 1}$. Observe that $\rho_1(k) + 1/\rho_1(k) = 2\sqrt{1 + \left(\frac{k}{2\theta}\right)^2}$ so that we get the bound

$$|f_k| \leq \rho_1(k)^{-k} \exp\left(2\theta\left(\sqrt{1 + \left(\frac{k}{2\theta}\right)^2} - 1\right)\right) \tag{9}$$

which allows us to provide an a-priori estimate of the numerical bandwidth of the Toeplitz part of $\exp(T(a))$.

In fact, computing the logarithm of the above bound for a sufficiently large number of positive integers k and comparing it to the logarithm of ϵ, with the help of Matlab, we get upper bounds for the numerical bandwidth of $\exp(T(a))$ displayed in Table 1. It is interesting to observe that in this specific case, the theoretical upper bounds exactly coincide with the actual values obtained in the algorithmic treatment performed in Sect. 7 reported in Table 9.

A similar analysis can be performed for matrices arising in Markov chains, where $a_0 < 0$, $a_i \geq 0$ for $i \neq 0$ and $\sum_i a_i \leq 0$. Observe that the nonnegativity of a_i for $i \neq 0$ implies that $\max_{|z|=\rho} |\exp(a(z))| = \exp(a_0 + \sum_{j\neq 0} \rho^j a_j)$ so that the bounds that one obtains are sharp.

4.2 Numerical size of the correction

Here we estimate the numerical size of the correction $F = \exp(T(a)) - T(\exp(a))$, where we assume that $a(z) = \sum_{i=-n_-}^{n_+} a_i z^i$ is a Laurent polynomial. We recall the following identity [16, Section 10.2], which is valid also for operators S and H [12]:

$$\exp(S + H) - \exp(S) = \int_0^1 \exp(S(1-s))H\exp((S+H)s)ds. \tag{10}$$

Consider the bi-infinite Toeplitz matrix $T_{\pm\infty}(a) = (t_{i,j})_{i,j\in\mathbb{Z}}$, $t_{i,j} = a_{j-i}$. It is well known [6] that if $a(z) \in \mathcal{W}$ then $T_{\pm\infty}(a)$ defines a bounded operator in $\ell^2(\mathbb{Z})$ such that $\|T_{\pm\infty}(a)\|_2 \leq \|a\|_{\mathcal{W}}$. Partition it into a 2×2 block matrix

$$T_{\pm\infty}(a) = \left[\begin{array}{c|c} \hat{T}(a) & Y_1 \\ \hline Y_2 & T(a) \end{array}\right] \tag{11}$$

where $\hat{T}(a) = (a_{j-i})_{i,j\le 0}$ and Y_1, Y_2 derive from the partitioning. Split $T_{\pm\infty}(a)$ as

$$T_{\pm\infty}(a) = S + H, \quad S = \left[\begin{array}{c|c} \hat{T}(a) & Y_1 \\ \hline 0 & T(a) \end{array}\right], \quad H = \left[\begin{array}{c|c} 0 & 0 \\ \hline Y_2 & 0 \end{array}\right]. \tag{12}$$

Observe that $\exp(T_{\pm\infty}(sa)) = T_{\pm\infty}(\exp(sa))$ so that, by partitioning $T_{\pm\infty}(\exp(sa))$ into a 2×2 block matrix we get

$$T_{\pm\infty}(\exp(sa)) = \left[\begin{array}{c|c} * & Z(s) \\ \hline * & T(\exp(sa)) \end{array}\right],$$

where $Z(s) = (z_{i,j}(s))_{i\le 0, j>0}$, $z_{i,j}(s) = f_{j-i}(s)$, $\exp(a(z)s) = \sum_{i\in\mathbb{Z}} f_i(s)z^i$, and the $*$'s denote suitable submatrices. Since, for the block triangular structure of S, one has

$$\exp(sS) = \left[\begin{array}{c|c} * & * \\ \hline 0 & \exp(sT(a)) \end{array}\right],$$

it turns out that the correction F is given by the $(2, 2)$ block of $\exp(S) - \exp(S + H)$. In view of (10), this matrix is expressed by the following integral

$$F = -\int_0^1 \exp(T(a)(1-s))Y_2 Z(s)ds. \tag{13}$$

In view of the results of Sect. 4.1, the coefficients $f_k(s)$ have an exponential decay as $k \to \infty$, more specifically $|f_k(s)| \le \max_{|z|=\rho} |\exp(a(z)s)|\rho^{-k} \le \sigma(\rho)\rho^{-k}$ where $\sigma(\rho) = \max_{s\in[0,1]} \max_{|z|=\rho} |\exp(a(z)s)|$.

Since Y_2 is nonzero only in the top-right $n_- \times n_-$ submatrix $\hat{Y}_2$, the integrand of (13) can be written in the form $T_1(s)\hat{Y}_2 Z_1(s)$, where $T_1(s)$ is formed by the first n_- columns of $\exp(T(a)(1-s))$ and $Z_1(s)$ is formed by the last n_- rows of $Z(s)$. More specifically, the jth column of $Z_1(s)$ has entries $f_{j-n_--1}(s), \ldots, f_j(s)$. Therefore, if ℓ is the length of the meaningful part of the infinite vector $(f_i(s))_{i\ge 1}$, that is $|f_i(s)| < \epsilon\|f(s)\|_\infty$ for $i > \ell$, then the columns of $Z(s)$ of index $j > \ell$ have entries of modulus smaller than $\epsilon\|f(s)\|_\infty$ so that they are relatively negligible and can be replaced by zero. Therefore, the value of ℓ provides the number of significant columns of the correction F. This value can be estimated in terms of the specific function $a(z)$ by following the analysis of Sect. 4.1. For instance, in the case $a(z) = \theta(z^{-1} - 2 + z)$ considered in the previous section, we get the same bound (9) for ℓ where θ is replaced by θs, and the maximum over $s \in [0, 1]$ must be taken.

A similar analysis can be carried out for estimating the significant number of rows of the correction. It turns out that the number of non-negligible columns depends on the decay of $f_i(s)$ for $i > 0$, while the number of non-negligible rows depends on the decay of $f_i(s)$ for $i < 0$.

4.3 Rank of the correction

The a-priori estimation of the numerical rank of the correction F is a difficult task. A similar problem is examined in the paper [19] by Kressner and Luce, where the *displacement rank* of the exponential of a finite Toeplitz matrix is considered. The displacement rank of a matrix A is defined as $\text{rank}\Delta(A)$ where $\Delta(A) = ZA - AZ$ and Z is the Toeplitz matrix having ones in the subdiagonal and zeros elsewhere. The numerical rank of F and the numerical displacement rank of $\exp(T(a))$ are closely related. In fact, if $\exp(T(a)) = T(\exp(a)) + F$, then $\Delta(\exp(A)) = \Delta(T(\exp(a)) + \Delta(F)$. This way, if $\text{rank}(F) = k$ then $\text{rank}\Delta(\exp(T(a))) \leq 2k + 2$ since the displacement rank of a Toeplitz matrix is at most 2. That is, a bound for the rank of the correction F provides a bound for the displacement rank. Conversely, a bound for the displacement rank of $\exp(T(a))$ provides a bound for the displacement rank of the correction F.

Indeed, the numerical size of the correction analyzed in the previous section is an upper bound for the numerical rank of F. It is possible to give a-priori bounds for the numerical rank of F depending only on the norm of $T(a)$, by using tools of numerical integration analyzed in [31] by Trefethen. Since these bounds do not use other information except the norm of $T(a)$, they are very general and thus not very sharp. However, they can be made sharper if information about the function $a(z)$ is available which allow us to choose integration contours well suited for the specific features of $a(z)$.

Using the Dunford–Cauchy formula one can write [11]

$$\exp(A) = \frac{1}{2\pi i} \int_{\Gamma} e^z (zI - A)^{-1} dz$$

where Γ is a Jordan curve containing all the eigenvalues of the operator A, say a circle of center 0 and radius $R > \|A\|$ for a give operator norm $\|\cdot\|$. Applying this formula to the matrices S and $S + H$ of Eq. (12) and with Γ being the circle of center 0 and radius $R > \max(\|S\|, \|S + H\|)$, we get

$$\exp(S + H) - \exp(S) = \frac{1}{2\pi i} \int_{|z|=R} e^z (zI - S)^{-1} H (zI - S - H)^{-1} dz. \quad (14)$$

Since $S + H = T_{\pm\infty}(a)$ then $\|S + H\|_2 \leq \|a\|_{\mathcal{W}}$. Moreover, from (12) we have $|S + H| = |S| + |H| \geq |S|$ so that, by the monotonicity of the 2-norm, it follows that $\|S\|_2 \leq \|S + H\|_2$. This way, any R such that $R > \|a\|_{\mathcal{W}}$ guarantees the condition $R > \max(\|S\|_2, \|S + H\|_2)$. By using the same arguments of the previous section, based on the 2×2 block partitioning of S and $S + H$, we obtain that F is the (2,2) block of the integral (14).

Observe that the matrix H has finite rank n_- so that, by applying a quadrature formula with N knots to approximate the integral of this submatrix, we find that F is approximated by the sum of N matrices of rank at most n_-, i.e., by a matrix of rank at most Nn_-. This way, the problem turns into finding the minimum number of knots which provides a numerical approximation of this integral with error at most ϵ. This

problem is investigated for an analytic function over the annulus $\{z \in \mathbb{C}, \quad r^{-1} < |z| < r\}$ by Trefethen and Weideman in [31] where the following result is proved:

Theorem 6 *Let $u(z)$ be an analytic function in the annulus $\{z \in \mathbb{C} : r^{-1} < |z| < r\}$ for $r > 1$, which takes values in $\mathbb{C}$. Let $M > 0$ be such that $M \geq |u(z)|$ for $r^{-1} < |z| < r$. Then,*

$$|\mathcal{I} - \mathcal{I}_N| \leq \frac{4\pi M}{r^N - 1},$$

where $\mathcal{I} = \int_{|z|=1} u(z)dz$, and $\mathcal{I}_N$ is the approximation obtained with the trapezoidal rule with N knots.

In order to apply this result to our case, we have to scale the problem in such a way that the integral is taken on the unit circle. First observe that the trapezoidal rule applied to the right-hand side of (14) is the same as applying the trapezoidal rule separately to both terms in the left-hand side. Therefore it is enough to consider an integral of the form $\mathcal{I} = \frac{1}{2\pi i} \int_{|z|=R} e^z (zI - A)^{-1} dz$, where A is such that $\|A\|_2 \leq \|a\|_{\mathcal{W}}$ and $R > \|a\|_{\mathcal{W}}$. In order to rescale the latter integral we change the variable $z = Ry$ and get

$$\mathcal{I} = \frac{1}{2\pi i} \int_{|y|=1} e^{Ry} \left(yI - \frac{1}{R} A \right)^{-1} dy. \tag{15}$$

Observe that the entries of the integrand matrix are analytic functions for $|y| > \|A\|_2/R$ so that we may apply Theorem 6 to each entry of the matrix with $r^{-1} > \|A\|_2/R$ and arrive at the following

Theorem 7 *Let $\mathcal{I}$ be the integral in (15), with A such that $\|A\|_2 \leq \|a\|_{\mathcal{W}}$ and $R > \|a\|_{\mathcal{W}}$. Denote $\mathcal{I}_N$ its approximation with the trapezoidal rule with N knots. Then we have*

$$\|\mathcal{I} - \mathcal{I}_N\|_2 \leq 2 \min_{(r,R)\in\Omega} \left\{ \frac{R}{(r^N - 1)(R - r\|a\|_{\mathcal{W}})} \max \left\{ e^{R/r} r, e^{Rr}/r \right\} \right\}$$

where $\Omega = \{(r, R) : R > \|a\|_{\mathcal{W}}, \quad 1 < r < R/\|a\|_{\mathcal{W}}\}$.

Proof It is sufficient to estimate an upper bound $m_{i,j}$ for the value of the modulus of the (i, j) entry of the integrand function in (15) for $r^{-1} < |y| < r$. This way, the matrix $M = (m_{i,j})_{i,j}$ is a componentwise upper bound for the matrix integral, so that we may apply Theorem 6 to get the component-wise bound $|\mathcal{I} - \mathcal{I}_N| \leq \frac{2}{r^N - 1} |M|$. Since the 2-norm is monotonic, from the latter bound we get $\|\mathcal{I} - \mathcal{I}_N\|_2 \leq \frac{2}{r^N - 1} \|M\|_2$. In order to estimate M we rely on the entry-wise inequalities

$$\left| e^{Ry} \left(yI - \frac{1}{R} A \right)^{-1} \right| = \left| e^{Ry} y^{-1} \left(I - \frac{1}{Ry} A \right)^{-1} \right| \leq \frac{e^{R|y|}}{|y|} \left(I - \frac{1}{|Ry|} |A| \right)^{-1}$$
$$\leq \frac{e^{R|y|}}{|y|} \left(I - \frac{1}{|R/r|} |A| \right)^{-1} =: M(|y|),$$

where we have used the property that $|(I-B)^{-1}| \le (I-|B|)^{-1}$, if $\|B\|_2 < 1$. Thus taking the maximum of each entry of $M(|y|)$ over $r^{-1} < |y| < r$, we get the matrix M which provides the sought entry-wise bound for the modulus of the integrand:

$$M = \sup_{r^{-1}<|y|<r} M(|y|) = \max_{r^{-1}\le|y|\le r} \frac{e^{R|y|}}{|y|} \cdot \left(I - \frac{r}{R}|A|\right)^{-1}.$$

Since the function $e^{R|y|}/|y|$ is convex, the maximum over the interval $[r^{-1}, r]$ is taken at one end of the interval so that

$$M = \max\left\{e^{R/r}r, e^{Rr}/r\right\} \cdot \left(I - \frac{r}{R}|A|\right)^{-1}.$$

Taking the norm yields

$$\|M\|_2 \le \max\left\{e^{R/r}r, e^{Rr}/r\right\} \cdot \frac{R}{R - r\|A\|_2}.$$

Thus, since $\|A\|_2 \le \|a\|_{\mathcal{W}}$, we get

$$\|\mathcal{I} - \mathcal{I}_N\|_2 \le \frac{2R}{(r^N-1)(R - r\|a\|_{\mathcal{W}})} \max\left\{e^{R/r}r, e^{Rr}/r\right\}$$

and the proof is complete. □

Observe that if $Rr \le \|a\|_{\mathcal{W}} + 1$ the maximum value is taken by the first function. If $R/r \ge \|a\|_{\mathcal{W}} + 1$ the maximum is taken by the second function. In these two cases the above problem turns into a constrained minimization problem which could be analyzed by applying the Kuhn–Tucker conditions.

An easier way to find out how large or small the bounds for the numerical rank obtained by applying the above theorem are, is to numerically evaluate this upper bound for (r, R) ranging in a grid discretization of Ω and taking the minimum. Then find the values of N which makes this minimum less than or equal to $\exp(\|a\|_{\mathcal{W}})\epsilon/2$. Indeed, the value obtained this way multiplied by $\min(n_-, n_+)$ is an upper bound for the numerical rank of the correction F.

Table 2 reports the values of the number N of knots as function of $\|a\|_{\mathcal{W}}$ computed this way. Indeed, these values are over estimates of the real values encountered in practice as shown in the numerical experiments. Actually, having more information about the function $a(z)$ helps to find a better contour line containing the eigenvalue of $T(a)$. This may provide better bounds for the number of required knots as shown in [11].

A different analysis can be based on the properties of the exponential series. Consider the case of a Laurent polynomial $a(z) = \sum_{i=-n_-}^{n_+} a_i z^i$ and observe that $\operatorname{rank} H(a_-) \le n_-$. Thus, from (4) we have $\operatorname{rank}(E_i) \le \operatorname{rank}(E_{i-1}) + n_-$. Moreover, since $E_i = E_{i-1}T(a) - H((a^{i-1})_-)H(a_+)$, we may also write $\operatorname{rank}(E_i) \le \operatorname{rank}(E_{i-1}) + n_+$. Whence we get

Table 2 Values of the number N of knots such that $N\min(n_-, n_+)$ is an upper bound for the numerical rank of $F = \exp(T(a)) - T(\exp(a))$ for different values of $\|a\|_{\mathcal{W}}$

$\|a\|_{\mathcal{W}}$	1	2	4	8	16	32	64	128
N	41	52	69	98	148	241	420	775

Table 3 Values of the number $k+1$ such that $(k+1)\min(n_-, n_+)$ is an upper bound for the numerical rank of $F = \exp(T(a)) - T(\exp(a))$ for different values of $\|a\|_{\mathcal{W}}$

$\|a\|_{\mathcal{W}}$	1	2	4	8	16	32	64	128
$k+1$	19	24	32	46	71	117	205	380

$$\operatorname{rank}(E_i) \le \operatorname{rank}(E_{i-1}) + n \le (i-1)n, \quad n = \min(n_-, n_+).$$

Now we are ready to estimate the numerical rank of the correction matrix $F = \sum_{i=0}^{\infty} \frac{1}{i!} E_i$. In fact, we may write $F = F_k + S_k$ where $F_k = \sum_{i=0}^{k} \frac{1}{i!} E_i$, $S_k = \sum_{i=k+1}^{\infty} \frac{1}{i!} E_i$. We have $\operatorname{rank}(F_k) \le n(k+1)$ while, in view of (5) we may write

$$\|S_k\|_p \le 2 \sum_{i=k+1}^{\infty} \frac{1}{i!} \|a\|_{\mathcal{W}}^i \le \frac{2\|a\|_{\mathcal{W}}^{k+1}}{(k+1)!} e^{\|a\|_{\mathcal{W}}}.$$

This property provides a tool for estimating the numerical rank of F. In fact, we may proceed this way. Given $\epsilon > 0$, say the machine precision of the floating point arithmetic, find k such that $\|S_k\|_p \le \epsilon \exp(\|a\|_{\mathcal{W}})$. This can be obtained by imposing the condition $\frac{2}{(k+1)!} \|a\|_{\mathcal{W}}^{k+1} e^{\|a\|_{\mathcal{W}}} \le \epsilon \exp(\|a\|_{\mathcal{W}})$. Then, determine the bound for the numerical rank of F as $(k+1)n$.

Table 3 provides upper bounds for the numerical rank of F in the case where $n = 1$, obtained by means of this analysis.

5 A generalization

Here we deal with the case of a matrix $A = T(a) + E$ where $a(z) \in \mathcal{W}$ and E is compact in L^p, or $a'(z) \in \mathcal{W}$ and $\|E\|_{\mathcal{F}} < +\infty$. Matrices of the kind $T(a) + E$ are typically encountered in applications related to the analysis of certain stochastic processes. The argument used in Sect. 3 can be applied to provide an expression to $\exp(A)$. In fact, we may write $\exp(A) = S_k + R_k$, where $S_k = \sum_{i=0}^{k} \frac{1}{i!} A^i$ and $R_k = \sum_{i=k+1}^{\infty} \frac{1}{i!} A^i$, so that $S_k = G_k + F_k$ with $G_k = \sum_{i=0}^{k} \frac{1}{i!} T(a^i)$, $\quad F_k = \sum_{i=0}^{k} \frac{1}{i!} D_i$, where

$$D_i = A^i - T(a^i), \quad i \ge 0. \tag{16}$$

Observe that $D_0 = 0$, $D_1 = E$. As in the previous section, there exists

$$F = \lim_{k\to\infty} F_k = \lim_{k\to\infty} S_k - \lim_{k\to\infty} G_k = \exp(A) - T(\exp(a)).$$

Theorem 8 *Let $A = T(a) + E$ where $a(z) \in \mathcal{W}$ and E is compact in L^p. Then $\exp(A) = T(\exp(a)) + F$, with F a compact operator in L_p such that*

$$\|F\|_p \le \exp(\|T(a) + E\|_p) + \exp(\|a\|_{\mathcal{W}}) \le \|a\|_{\mathcal{W}}(1 + \exp(\|E\|_p)).$$

Proof Taking p-norms in both sides of $F = \exp(T(a) + E) - T(\exp(a))$, we get the sought bound. Concerning the compactness of F, we observe that $H(a_-)$ and E are compact and bounded in the p norm. Therefore from (16) and in view of [20, Lemma 8.3-2], it follows that D_i is compact for any i. This implies that F_k is compact for any k as well, so that F is compact as a limit of compact operators [20, Theorem 8.1-5]. □

If $a'(z) \in \mathcal{W}$ and $\|E\|_{\mathcal{F}} < +\infty$ we may prove a similar result.
Concerning $\|F\|_{\mathcal{F}}$, from (16) we deduce that, for $i \ge 1$,

$$\begin{aligned} A^i &= (T(a) + E)(T(a^{i-1}) + D_{i-1}) \\ &= T(a)T(a^{i-1}) + ET(a^{i-1}) + AD_{i-1} \end{aligned}$$

and, in view of Lemma 1, it follows that

$$A^i = T(a^i) - H(a_-)H((a^{i-1})_+) + ET(a^{i-1}) + AD_{i-1}.$$

Hence, we obtain

$$D_i = AD_{i-1} - H(a_-)H((a^{i-1})_+) + ET(a^{i-1}). \tag{17}$$

Since from Lemma 1 we have $\|T(a)D_{i-1}\|_{\mathcal{F}} \le \|a\|_{\mathcal{W}}\|D_{i-1}\|_{\mathcal{F}}$, we may write

$$\|AD_{i-1}\|_{\mathcal{F}} = \|T(a)D_{i-1} + ED_{i-1}\|_{\mathcal{F}} \le (\|a\|_{\mathcal{W}} + \|E\|_{\mathcal{F}})\|D_{i-1}\|_{\mathcal{F}}.$$

Therefore, from the above inequality and from Lemma 1, in view of (17), we obtain

$$\begin{aligned} \|D_i\|_{\mathcal{F}} &\le (\|a\|_{\mathcal{W}} + \|E\|_{\mathcal{F}})\|D_{i-1}\|_{\mathcal{F}} + (i-1)\|a\|_{\mathcal{W}}^{i-2}\|a'\|_{\mathcal{W}}^2 + \|a\|_{\mathcal{W}}^{i-1}\|E\|_{\mathcal{F}}, \\ &\le \xi\|D_{i-1}\|_{\mathcal{F}} + \gamma_i, \quad i \ge 1, \end{aligned}$$

where

$$\xi = \|a\|_{\mathcal{W}} + \|E\|_{\mathcal{F}}, \quad \gamma_i = (i-1)\|a\|_{\mathcal{W}}^{i-2}\|a'\|_{\mathcal{W}}^2 + \|a\|_{\mathcal{W}}^{i-1}\|E\|_{\mathcal{F}}, \quad i \ge 1, \tag{18}$$

and $\|D_0\|_{\mathcal{F}} = 0$, $\|D_1\|_{\mathcal{F}} = \|E\|_{\mathcal{F}}$. Thus we may bound $\|D_i\|_{\mathcal{F}}$ with the value that the polynomial $p(z) = \sum_{j=0}^{i-1} z^j \gamma_{i-j}$ takes at $\xi = \|a\|_{\mathcal{W}} + \|E\|_{\mathcal{F}}$, i.e,

$$\|D_i\|_{\mathcal{F}} \le \sum_{j=0}^{i-1} \xi^j \gamma_{i-j}, \quad \xi = \|a\|_{\mathcal{W}} + \|E\|_{\mathcal{F}}. \tag{19}$$

Thus, concerning the sequence F_k, from (19) we obtain

$$\|F_k\|_{\mathcal{F}} \le \sum_{i=1}^{k} \frac{1}{i!} \|D_i\| \le \sum_{i=1}^{k} \sum_{j=0}^{i-1} \frac{1}{i!} \xi^j \gamma_{i-j}.$$

For notational simplicity set $\alpha = \|a\|_{\mathcal{W}}$, $\beta = \|E\|_{\mathcal{F}}$ so that $\xi = \alpha + \beta$ and $\gamma_k = (k-1)\alpha^{k-2}\|a'\|_{\mathcal{W}}^2 + \alpha^{k-1}\beta$. Then, since $\alpha \le \xi$, we have $\gamma_k \le (k-1)\xi^{k-2}\|a'\|_{\mathcal{W}}^2 + \xi^{k-1}\beta$. Whence we deduce that

$$\begin{aligned}
\|F_k\|_{\mathcal{F}} &\le \sum_{i=1}^{k} \sum_{j=0}^{i-1} \frac{1}{i!} \left[\|a'\|_{\mathcal{W}}^2 (i-j-1)\xi^{i-2} + \beta \xi^{i-1} \right] \\
&= \|a'\|_{\mathcal{W}}^2 \sum_{i=1}^{k} \frac{1}{2} \frac{i(i-1)}{i!} \xi^{i-2} + \beta \sum_{i=1}^{k} \frac{i}{i!} \xi^{i-1} \\
&\le \frac{1}{2} \|a'\|_{\mathcal{W}}^2 \exp(\xi) + \beta \exp(\xi).
\end{aligned}$$

Thus we may conclude with the following

Theorem 9 *Let $A = T(a) + E$. If $a'(z) \in \mathcal{W}$ and $\|E\|_{\mathcal{F}} < \infty$, then for the matrices D_i of (16) we have*

$$\|D_i\|_{\mathcal{F}} \le \sum_{j=0}^{i-1} (\|a\|_{\mathcal{W}} + \|E\|_{\mathcal{F}})^j \gamma_{i-j},$$

where the constants γ_i, $i \ge 1$, are defined in (18). Moreover, $\exp(A) = T(\exp(a)) + F$, with $F = \lim_{k\to\infty} F_k$, $F_k = \sum_{i=0}^{k} \frac{1}{i!} D_i$ and

$$\|F\|_{\mathcal{F}} \le \left(\frac{1}{2} \|a'\|_{\mathcal{W}}^2 + \|E\|_{\mathcal{F}} \right) \exp(\|a\|_{\mathcal{W}} + \|E\|_{\mathcal{F}}).$$

The above theorem states that, also in the case of $A = T(a) + E$, the matrix $F = \exp(A) - T(\exp(a))$ is such that $\sum_{i,j\in\mathbb{Z}^+} |f_{i,j}| < +\infty$. Moreover, if $E = 0$, the bounds given in the above theorem reduce to the bounds given in Theorem 4.

With some formal manipulations, it is possible to provide the following bound for $\|D_i\|_{\mathcal{F}}$ expressed in closed form

$$\|D_i\|_{\mathcal{F}} \leq \frac{1}{\|E\|_{\mathcal{F}}} \left(\varphi \frac{(\|a\|_{\mathcal{W}} + \|E\|_{\mathcal{F}})^i - \|a\|_{\mathcal{W}}^i}{\|E\|_{\mathcal{F}}} - \psi i \|a\|_{\mathcal{W}}^{i-1} \right),$$
$$\varphi = \|a'\|_{\mathcal{W}}^2 + \|E\|_{\mathcal{F}}^2, \quad \psi = \|a'\|_{\mathcal{W}}^2$$

which, taking the limit for $\|E\|_{\mathcal{F}} \to 0$, coincides with the bound of Theorem 4.

6 Algorithms

In this section we provide an algorithm for computing the exponential function of a QT-matrix $A = T(a) + E$. Since $a(z) \in \mathcal{W}$, the coefficients a_i of $a(z)$ decay to zero as $i \to \pm\infty$, and the boundedness of $\|E\|_p$ for $p \neq 1, \infty$ implies that the entries $e_{i,j}$ of the matrix E decay to zero as $i \to \infty$ or $j \to \infty$. Thus, we may represent $a(z)$ in an approximate way just by considering a finite number of coefficients, i.e, $a(z) = \sum_{i=-n_-}^{n_+} a_i z^i + r(z)$, with $n_-, n_+ \geq 0$, where we assume that the remainder $r(z)$ is such that $\|r\|_{\mathcal{W}} \leq \epsilon$, for a given error bound ϵ. Similarly, also the matrix E can be represented in an approximate way by storing only a finite number of nonzero entries. Observe that for the decay of the coefficients a_i, also the matrix $H(a_-)$ can be represented, up to an error ϵ, by means of a semi-infinite matrix which is zero everywhere except in its $n_- \times n_-$ leading principal submatrix, which coincides with the Hankel matrix associated with $a_{-1}, \ldots, a_{-n_-}$.

For computational reasons, it is convenient to represent $H(a_-)$ as the product UV^T where U and V have infinitely many rows and n_- columns. Moreover, due to the truncation of the series, the matrices U and V have null entries for a sufficiently large row index.

Define $p_i(z) = \frac{1}{i!} a(z)^i$, $i \geq 0$, so that, for a sufficiently large i, $\exp(a)$ can be approximated by $s_i(z)$, defined by means of the recursion

$$\begin{aligned} p_i(z) &= \frac{1}{i} a(z) p_{i-1}(z), \\ s_i(z) &= s_{i-1}(z) + p_i(z), \quad i \geq 1, \end{aligned} \tag{20}$$

with $s_0(z) = 1$.

We consider separately, in the following two sections, the Toeplitz case, i.e., $A = T(a)$, and the general case where $A = T(a) + E$ with $E \neq 0$.

6.1 The Toeplitz case

Consider the case where A is Toeplitz, i.e., $A = T(a)$ and $E = 0$. According to the results of the previous section, the matrix $\exp(A)$ is approximated by $T(s_k) + F_k$, for a suitable $k \geq 1$. In order to compute $s_k(z)$ we rely on formula (20), while for computing $F_k = \sum_{i=0}^{k} \frac{1}{i!} E_i$ we rely on recursion (4).

We define $\widehat{E}_i = \frac{1}{i!}E_i$, so that we may rewrite Eq. (4) in the following form

$$\widehat{E}_i = \frac{1}{i}T(a)\widehat{E}_{i-1} - \frac{1}{i}H(a_-)H((p_{i-1})_+), \quad i \geq 1,$$

so that $F_k = \sum_{i=0}^{k} \widehat{E}_i$.

In order to reduce the complexity of the computation, we will represent also the matrices $\widehat{E}_i$, F_i and the matrix $H(a_-)$ in the form

$$\widehat{E}_i = U_i V_i^T, \quad F_i = W_i Y_i^T, \quad H(a_-) = UV^T, \tag{21}$$

where U_i, V_i, W_i, Y_i, U and V are matrices with infinitely many rows and with a finite number of columns. Moreover, due to the finite representation, these matrices have null entries if the row index is sufficiently large. Therefore they can be represented, up to an arbitrarily small error, with a finite number of parameters.

Using the decompositions (21) we may write

$$U_i V_i^T = \frac{1}{i}T(a)U_{i-1}V_{i-1}^T - \frac{1}{i}UV^T H((p_{i-1})_+),$$

whence

$$U_i = \left[T(a)U_{i-1} \mid U\right], \quad V_i = \left[\tfrac{1}{i}V_{i-1} \mid -\tfrac{1}{i}H((p_{i-1})_+)V\right]. \tag{22}$$

Moreover, from the relation $F_k = F_{k-1} + U_k V_k^T$ we obtain

$$W_k = [W_{k-1} \mid U_k], \quad Y_k = [Y_{k-1} \mid V_k]. \tag{23}$$

By using these decompositions, the implementation of Eq. (4), together with the computation of $F_k = \sum_{i=1}^{k} \widehat{E}_i$ and of the function $s_k(z) = \sum_{i=0}^{k} p_i(z)$, will proceed as described in the following algorithm, where matrices are formed by a finite number of rows containing the nonzero part of the corresponding infinite matrices.

Algorithm 1 kth step of Taylor expansion

Input: Integer $k \geq 1$, the coefficient vectors of the functions $a(z)$, $p_{k-1}(z)$, $s_{k-1}(z)$ and the matrices U, V, U_{k-1}, V_{k-1}, W_{k-1}, Y_{k-1}, such that (21) holds for $i = k-1$

Output: The coefficient vectors of the functions $p_k(z)$, $s_k(z)$ and the matrices U_k, V_k, W_k, Y_k, such that (21) holds for $i = k$

Computation:

1: compute $P_1 = H((p_{k-1})_+)V/k$, set $Q_1 = [\frac{1}{k}V_{k-1} \mid -P_1]$
2: compute $P_2 = T(a)U_{k-1}$, set $Q_2 = [P_2 \mid U]$
3: compress the pair Q_1, Q_2 and get a new pair V_k, U_k
4: set $S_1 = [W_{k-1} \mid U_k]$ and $S_2 = [Y_{k-1} \mid V_k]$
5: compress the pair S_1, S_2 and get the new pair W_k, Y_k
6: compute $p_k(z) = \frac{1}{k}a(z)p_{k-1}(z)$ and set $s_k(z) = s_{k-1}(z) + p_k(z)$
7: truncate $s_k(z)$ and $p_k(z)$

In the above description we have used a compression operation in stages 3 and 5 acting on a pair of matrices, together with the operation of truncating a Laurent polynomial at stage 7. We will describe these operations in Sect. 6.3. Observe also that even if the involved matrices have infinitely many rows, only a finite number of them are nonzero. A detailed implementation of the above algorithm keeps track of the number of nonzero rows of each matrix.

6.2 The general case

Consider the case where $A = T(a) + E$, with $E \neq 0$. According to the results of Sect. 5, the matrix $\exp(A)$ is approximated by $T(s_k) + F_k$, for a suitable $k \geq 1$, where $F_k = \sum_{i=0}^{k} \frac{1}{i!} D_i$ and the matrices D_i are defined in (16).

As in the previous section, define $\widehat{D}_i = \frac{1}{i!} D_i$, so that, in view of (17), we have

$$\widehat{D}_i = \frac{1}{i} A \widehat{D}_{i-1} - \frac{1}{i} H(a_-) H((p_{i-1})_+) + \frac{1}{i} E T(p_{i-1}). \tag{24}$$

Let us represent the matrices E, $H(a_-)$ and D_i in the form $E = WY^T$, $H(a_-) = UV^T$ and $\widehat{D}_i = U_i V_i^T$, where W, Y, U, V, U_i and V_i have a finite number of columns. We may rewrite Eq. (24) in the form

$$U_i V_i^T = \frac{1}{i}(T(a) + WY^T) U_{i-1} V_{i-1}^T - \frac{1}{i} UV^T H((p_{i-1})_+) + \frac{1}{i} WY^T T(p_{i-1}).$$

Whence we deduce that

$$\begin{aligned} U_i &= [(T(a)U_{i-1} + W(Y^T U_{i-1}) \;\vdots\; U \;\vdots\; W], \\ V_i &= \left[\frac{1}{i} V_{i-1} \;\vdots\; -\frac{1}{i} H((p_{i-1})_+) V \;\vdots\; \frac{1}{i} T(p_{i-1})^T Y\right]. \end{aligned}$$

Moreover, by representing F_k as $F_k = W_k Y_k^T$, from the relation $F_k = F_{k-1} + U_k V_k^T$ we obtain

$$W_k = [W_{k-1} \;\vdots\; U_k], \quad Y_k = [Y_{k-1} \;\vdots\; V_k].$$

It is immediate to write an algorithm that implements the above equations.

6.3 Compression and truncation

Given the matrix E in the form $E = FG^T$ where F and G are matrices of size $m \times k$ and $n \times k$, respectively, we aim to reduce the size k and to approximate E by means of $\widetilde{E} = \widetilde{F}\widetilde{G}^T$ where $\widetilde{F}$ and $\widetilde{G}$ are matrices of size $m \times \tilde{k}$ and $n \times \tilde{k}$, respectively, with $\tilde{k} \leq k$.

We use the following procedure which, for simplicity, we describe in the case of real matrices. Compute the QR factorizations $F = Q_f R_f$, $G = Q_g R_g$, where Q_f

and Q_g are orthogonal and R_f, R_g are upper triangular. Then, in the factorization $FG^T = Q_f(R_f R_g^T)Q_g^T$, compute the SVD of the matrix in the middle $R_f R_g^T = U\Sigma V^T$ where the singular values σ_i satisfying the condition $\sigma_i < \epsilon\sigma_1$ are removed together with the corresponding columns of U and V, where ϵ is a given tolerance, say the machine precision. In output, the matrices $\widetilde{F} = Q_f U \Sigma^{1/2}$, $\widetilde{G} = Q_g V \Sigma^{1/2}$ are delivered.

This procedure is described with more details in Algorithm 2. The overall cost of this algorithm is $O(k^2(m+n))$.

Algorithm 2 Compression

Input: Matrices F and G of size $m \times k$ and $n \times k$, respectively, a real $\epsilon > 0$

Output: matrices $\widetilde{F}$ and $\widetilde{G}$ of size $m \times \widetilde{k}$ and $n \times \widetilde{k}$, respectively, such that $\widetilde{k} \le k$ and $\|FG^T - \widetilde{F}\widetilde{G}^T\|_2 \le \epsilon\|F\|_2\|G\|_2$

Computation:

1: Compute the QR factorizations $F = Q_f R_f$, $G = Q_g R_g$;

2: compute the SVD of $R_f R_g^T$, i.e., $R_f R_g^T = U\Sigma V^T$;

3: select the integer ℓ such that $\sigma_i < \epsilon\sigma_1$ for $i > \ell$ where σ_i are the singular values, and set $\widehat{U}$, $\widehat{V}$ the submatrices formed by the first ℓ columns of U and V, respectively; set $\widehat{\Sigma} = \mathrm{diag}(\sigma_1, \ldots, \sigma_\ell)$ so that $\|U\Sigma V^T - \widehat{U}\widehat{\Sigma}\widehat{V}^T\|_2 \le \sigma_{\ell+1}$;

4: output $\widetilde{F} = Q_f\widehat{U}\widehat{\Sigma}^{\frac{1}{2}}$, $\widetilde{G} = Q_g\widehat{V}\widehat{\Sigma}^{\frac{1}{2}}$.

6.4 Scaling and squaring

The scaling and squaring technique, described in the book [16], is a standard way for accelerating convergence of the exponential series. It consists of replacing the matrix A with $B = \frac{1}{2^q}A$, computing the exponential $\exp(B)$ and recover $\exp(A)$ as $\exp(B)^{2^q}$. The advantage is that, with a suitable choice of q, the length of the Taylor expansion is substantially reduced, at the cost of performing a small number of repeated squarings.

This technique is easily implementable in our framework. In particular, in the case where $A = T(a)$, we determine the least integer q such that $\|a\|_{\mathcal{W}}/2^q < 1$, then we set $\widehat{a}(z) = a(z)/2^q$ so that

$$\exp(T(a)) = \exp(T(\widehat{a}))^{2^q}$$

and we may compute $\exp(T(a))$ by first computing $\exp(T(\widehat{a}))$, which requires a shorter power series expansion, and then computing $\exp(T(\widehat{a}))^{2^q}$ by means of q steps of repeated squarings applied to $\exp(T(\widehat{a}))$.

The square of a matrix of the kind $T(a) + E$ is computed by means of the equation

$$(T(a)+E)^2 = T(a^2) - H(a_-)H(a_+) + T(a)E + ET(a) + E^2 =: T(a^2) + \widehat{E}$$

where $\widehat{E} = -H(a_-)H(a_+) + T(a)E + ET(a) + E^2$.

Assuming that E is factored in the form $E = WY^T$, for W and Y being slim matrices, and that $H(a_-)H(a_+)$ is factored as UV^T, then $\widehat{E}$ is factored as $\widehat{E} = \widehat{U}\widehat{V}^T$

where

$$\widehat{U} = [-U \mid T(a)W \mid W], \quad \widehat{V} = [V \mid Y \mid T(a)^T Y + Y(W^T Y)]. \tag{25}$$

This formula requires to compute a rank revealing factorization of the Hankel product $H(a_-)H(a_+)$. Two algorithms for this computation are described in [5] and are based on a Lanczos-type method, in the form of the Golub–Kahan bidiagonalization procedure [25], or on a random sampling approach [14]. The cost of these approaches, which exploit the Hankel structure, is $O(rn \log n + r^2 n)$ where r is the value of the numerical rank of the product. A compression step, performed according to Algorithm 2 can be applied to reduce the rank of $\widehat{U}$ and $\widehat{V}$. Algorithm 3 describes the squaring of QT-matrix.

Algorithm 3 Square of a QT-matrix

Input: $a(z)$, W and Y defining the QT-matrix $A = T(a) + WY^T$
Output: $b(z)$, $\widehat{U}$ and $\widehat{V}$ such that $A^2 = T(b) + \widehat{E}$, with $\|\widehat{E} - \widehat{U}\widehat{V}^T\|_2 \leq \epsilon \|\widehat{U}\|_2 \|\widehat{V}^T\|_2$
Computation:

1: compute $b(z) = a(z)^2$
2: compute a rank revealing factorization $H(a_-)H(a_+) = UV^T$
3: set $S_1 = [-U \mid T(a)W \mid W]$, $S_2 = [V \mid Y \mid T(a)^T Y + Y(W^T Y)]$
4: compress the pair S_1, S_2 to get the new pair $\widehat{U}, \widehat{V}$

6.5 Cost analysis

We may perform a complexity analysis of the algorithms designed in the previous sections. We consider only the case where $A = T(a)$ and divide the problem into the different sub-problems of evaluating the recurrence (4) by means of Eqs. (22) and (23), performing the compression according to Algorithm 2, and computing the repeated squaring of a QT-matrix.

Concerning (22), we have to compute the product $T(a)U_{i-1}$, where $T(a)$ is an infinite Toeplitz matrix having bandwidth $n_- + n_+$, and U_{i-1} has infinitely many rows and a finite number, say r_{i-1}, of columns. Denoting by m_{i-1} the number of numerically nonzero rows of U_{i-1}, the problem is reduced to multiplying an $(m_{i-1} + n_-) \times m_{i-1}$ Toeplitz matrix and an $m_{i-1} \times r_{i-1}$ matrix. By using fast algorithms for Toeplitz-vector matrix multiplication we have a cost of $O(r_{i-1}(m_{i-1} + n_-)\log(m_{i-1} + n_-))$ arithmetic operations (ops). Similarly, the computation of the product $H((p_{i-1})_+)V$ is reduced to multiplying an $n_{i-1} \times q_i$ Hankel matrix times a matrix of size $q_i \times s$, where n_{i-1} is the degree of the polynomial $(p_{i-1}(z))_+$, s is the number of columns of V, $q_i = \min(n_{i-1}, n)$, with n the number of numerically nonzero rows of V. Thus, even this computation has a cost of $O(sn_{i-1} \log n_{i-1}))$. In fact, the product of a Hankel matrix and a vector, up to permutation, is the same as the product of a Toeplitz matrix and a vector where FFT-based algorithms can be used.

The cost of compression in the steps 3 and 5 of Algorithm 1, performed with Algorithm 2, which relies on QR and SVD, is proportional to the square of the rank and to the maximum dimension. Finally, the cost analysis of Algorithm 3 is the same

as that of Algorithm 1, except for the computation of the rank-revealing factorization $H(a_-)H(a_+) = UV^T$.

An upper bound for the overall cost of the computation of the exponential can be given in terms of the overall number q of terms in the exponential series and the number of squaring steps. To this purpose, denote by b, n and r upper bounds for the numerical bandwidth, size and rank of the correction, respectively of all the QT-matrices involved in the Taylor series computation and in the squaring stage. Then we may write the overall cost in the form $O(q((b+n)r\log(b+n) + nr^2))$. In the numerical experimentation performed so far, the values of b, n and r coincide with those of the matrix $\exp(T(a))$. Recall also that, according to our analysis of Sect. 4.1, the values of the numerical bandwidth b and of the numerical size n have the same behaviour. Thus, the most expensive part of this complexity bound is the term $O(r^2 n)$ due to compression. In the cases where the numerical rank of F is much smaller than the numerical size the algorithm performs very efficiently.

7 Numerical experiments

We have provided a Matlab implementation of the algorithm for the computation of $\exp(T(a)) = T(\exp(a)) + F$, based on Taylor expansion with scaling and squaring, valid for semi-infinite matrices. We denote this algorithm with the symbol `infQT`. This algorithm can be also applied to finite $n \times n$ matrices under the assumption that n is larger than the numerical bandwidth of $T(\exp(a))$ and the numerical size of F. In fact, if this condition is satisfied, then it can be verified that, numerically, $\exp(T_n(a)) = T_n(\exp(a)) + F_n + J_n F_n J_n$, where J_n is the $n \times n$ permutation matrix with ones in the anti-diagonal and F_n is the $n \times n$ leading principal submatrix of F.

A second algorithm has been implemented for finite matrices, which works also in the general case where the bandwidth of $T(\exp(a))$ or the numerical size of F is larger than n. This version relies on the CQT-matrix arithmetic, valid for both finite and infinite QT-matrices, available in the package CQT-Toolbox of [5], and substantially coincides with the `expm` function invoked with the option `'taylor'` available in the CQT-Toolbox. The computation of the exponential relies once again on the truncated Taylor series with scaling and squaring, but acting on *finite QT-matrices*. We will refer to this version with the symbol `finQT`. The software is provided upon request by the authors.

We have compared our implementation with the function `expmt` for computing matrix exponential of finite Toeplitz matrices recently introduced by Kressner and Luce [19]. This version is based on the properties of Displacement Rank operators. We will refer to this algorithm with the symbol `DR`. We have also considered the Matlab function `expmv` of [1], available from https://github.com/higham/expmv, for computing the product $y = \exp(A)x$, where A is a sparse matrix which relies only on Matrix-Vector multiplications. We denote this software with the symbol `MV`, while we use the symbol `M` to denote the algorithm `expm` of Matlab.

It must be said that the algorithm `DR`, based on displacement rank, is not optimized for banded matrices and does not exploit any decay of coefficients of $\exp(A)$. In

principle, it would actually be possible to incorporate decay into DR by truncating generators. On the other hand, both infQT and finQT exploit decay properties.

Some numerical experiments have been performed with several test problems by analyzing the CPU time and the approximation error. All the experiments have been run under the Linux system on an I7 processor with Matlab R2018a. For each experiment we report the CPU time computed with the commands tic and toc, and the normwise relative error. In the case where we compute the vector $y = \exp(A)x$, the error is given by $\|y - \widehat{y}\|_\infty / \|y\|_\infty$, where y is computed by means of Matlab and $\widehat{y}$ denotes the vector computed with the algorithms infQT, finQT, DR, MV. In computing $B = \exp(A)$, the error has been evaluated as $\max_{i,j} |b_{i,j} - \widehat{b}_{i,j}| / \max_{i,j} |b_{i,j}|$, where $b_{i,j}$ are the values of the matrix exponential computed by the Matlab function expm and $\widehat{b}_{i,j}$ are the values computed by the algorithms infQT, finQT, DR. The CPU time does not include the cost of reconstructing the exponential matrix as a full matrix for the algorithm DR. In some cases, we report also the graph with the distribution of the errors obtained with the command mesh(log10(abs(Err))) where Err is the matrix $(B - \widehat{B}) / \max_{i,j} |b_{i,j} - \widehat{b}_{i,j}|$.

In all the experiments, in the truncation and compression stages, we set the parameter ϵ equal to the machine precision 2.22e−16. Moreover, in the evaluation of the error, in the case of finite matrices we compared the output of the different algorithms to the output of the Matlab function expm. In the case of an infinite matrix A, in order to make comparisons with the algorithms valid for finite matrices for computing $y = Ax$, we have truncated A to the matrix A_N of finite size N, where N is larger than the numerical bandwidth and the numerical size of the correction of $\exp(A)$. We have computed the vector $w := \texttt{expm}(A_N)x_N$ where x_N is the truncation of x to finite size N. The comparison of the different algorithms is restricted to the first M components of the computed vectors, for a suitable $M < N$. The values of M and N are reported in the related tables. This comparison provides a heuristic approach which may give some error estimate of the algorithms.

The tests concern: two problems with infinite matrices coming from the transient analysis of M/M/1 queues modeled by Markov chains in continuous time where A is a *generator* [28]; the case of the finite-differences second derivative in finite dimension $A = (\Delta_t/\Delta_x^2)\text{trid}_n(1, -2, 1)$, coming from the numerical treatment of the heat equation where $\Delta_t = \Delta_x = 1/(n+1)$; a matrix used as test in [19] which models the Merton problem.

7.1 Infinite matrices: the transient analysis of an M/M/1 queue

The M/M/1 queue (see [28, Section 5.1]) is a queueing model where customers arrive according to a Poisson process with rate $\lambda > 0$, i.e., interarrival times are independent and identically exponentially distributed with rate λ. The service times are independent and identically exponentially distributed with rate $\mu > 0$. There is one server, the service discipline is FIFO and the queue capacity is infinite. This queue is modelled by means of a Markov process, where X_t is the number of customers in the queue at time t. The probability distribution of X_t, i.e., the number of customers in the queue at time t, is given by the vector $\pi(t)^T = \pi(0)^T \exp(Qt)$ where Q is the *generator matrix*

of the form $Q = T(a) + E$ where $a(z) = z^{-1}\mu - (\lambda + \mu) + z\lambda$ and $E = \mu e_1 e_1^T$, with $e_1 = (1, 0, 0, \ldots)^T$. The distribution of the busy period duration is obtained by means of the vector $y = \exp(T(a)t)e$, where e is the vector of all ones (see [28, Section 5.1.2]).

A generalization of the above model gives rise to the following generator matrix, that represents an M/G/1 type Markov process [24],

$$Q = \begin{pmatrix} b_0 & b_1 & b_2 & b_3 & \ldots \\ \mu & -(\lambda+\mu) & k_0 & k_1 & \ldots \\ & \mu & -(\lambda+\mu) & k_0 & \ddots \\ & & \ddots & \ddots & \ddots \end{pmatrix}.$$

where $b_i \geq 0$ for $i > 0, b_0 = -\sum_{i=1}^{\infty} b_i, \mu > 0$ and $k_i, i \geq 0$, are nonnegative numbers such that $\sum_{k=0}^{\infty} k_i = \lambda$.

7.1.1 Banded matrix with large bandwidth

We have considered the matrix Q defined by $\mu = 230, \lambda = 1, k_i = 1/\lambda, i = 0, \ldots, 200, k_i = 0$ for $i > 200$. With these values, the queue turns out to be positive recurrent and the range of the components y_i of the busy period duration vector $y = \exp(A)e$ which are in the range $[\epsilon, 1 - \epsilon]$ is meaningfully large, where $\epsilon = 2.2204 \cdot 10^{-16}$. In fact, we have restricted the computation to the first m components of y having values in the above interval.

The computation has been performed in the following way: for $t = 2^i, i = 0, 1, \ldots, 6$, we have considered the matrix $A = tQ$ and computed $\exp(A)$ together with $y = \exp(A)e$ by means of the `infQT` algorithm. In this computation, we used the information about the numerical bandwidth of $\exp(A)$ and the size of the correction, to determine how many terms to sum up in the computation of y together with the value of meaningful components y_i. The former value is used to truncate the size of the infinite matrix to a finite value N in order to apply the algorithms `finQT, DR, MV, M`.

Figure 1 displays the graphs with the timings of the different algorithms, while Table 4 reports the errors in the approximation of y. Table 5 reports the values of the lower and upper bandwidth k_-, k_+ in the Toeplitz part of $\exp(tQ)$, the values of the numerical size (m, n) and the rank r of the correction F, together with the truncation size N where the infinite matrix is truncated in order to apply the other algorithms with truncation error less than the machine precision.

It is interesting to observe that the cost of `infQT` is negligible with respect to the other algorithms. In fact, since the numerical bandwidth of the Toeplitz part and the size of the correction are quite small, the complexity of computing $\exp(A)$ is small as well and is independent of the truncation size N, while the other algorithms have a cost which necessarily depends on N. Concerning the approximation errors, Table 4 shows that algorithm `infQT` performs better than the other algorithms while `DR` has the largest errors and seems to loose half the digits.

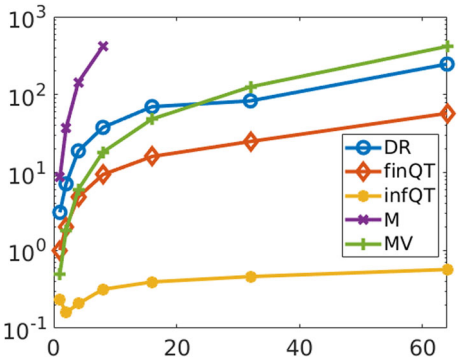

Fig. 1 CPU times in computing $y = \exp(tQ)e$ for the semi-infinite generator matrix Q for $t = 2^i$, $i = 0, 1, \ldots, 6$

Table 4 Error estimates in the computation of $y = \exp(A)e$ for the infinite generator matrix $A = tQ$ and the vector e with components 1. The corresponding truncation values N and M are reported in Table 5

t	1	2	4	8
DR	7.1e−07	6.5e−06	1.6e−05	2.7e−06
finQT	1.3e−12	2.6e−12	5.2e−12	1.0e−11
infQT	5.8e−14	1.2e−13	2.3e−13	4.6e−13
expmv	2.3e−13	1.3e−12	1.4e−12	1.3e−12

Table 5 Lower and upper bandwidth k_-, k_+ of the Toeplitz part, numerical sizes m, n and rank r of the correction F in $\exp(A) = T(\exp(a)) + F$, where $A = tQ$ for the infinite generator matrix Q. In the last two lines, N is the size of the truncated matrix to which the other algorithms are applied while M is the number of meaningful components of y which are computed

t	1	2	4	8	16	32	64
k_-	390	680	1219	2238	4181	8361	16,721
k_+	2634	3076	3625	4268	4982	9963	19,925
m	364	645	1282	2197	4129	7815	14,474
n	2315	5008	8076	11,818	17,657	20,935	30,885
r	42	50	49	35	19	12	10
N	3024	5008	8076	11,818	17,657	20,935	36,646
M	390	680	1219	2238	4181	8361	16,721

7.1.2 Dense matrix with exponential decay

In this test we considered a dense semi-infinite generator matrix $Q = T(a)$, defined by the function $a(z) = t(a_0 + \sum_{i=1}^{\infty}(0.9^i z^{-i} + (i+1)0.7^i z^i))$, where $a_0 = -\sum_{i=1}^{\infty}\left(0.9^i + (i+1)0.7^i\right)$, for $t = 2^j$, $j = 0, 1, \ldots, 6$. We have computed $\exp(T(a))$ with the algorithms infQT, finQT, DR and M and we have multiplied the matrix exponential by the vector e of all ones. The size of e and of the truncation of $T(a)$, for applying algorithms for finite matrices, is determined according to the numerical bandwidth and to the size of the correction in $\exp(T(a))$. We have not applied the algorithm MV since it is taylored for sparse matrices while $T(a)$ is a dense matrix.

Figure 2 reports the plot with the CPU time of the algorithms, while Table 6 reports the values of the relative errors in computing $y = \exp(T(a))e$, where e is the vector of all ones, for each value of t. Figure 3 graphically shows the distribution of the errors, as function of (i, j), in the matrix exponential for the algorithms `infQT` and `DR`. Table 7 reports the sizes of the bandwidth, the size and the rank of the correction together with the truncation size of the infinite matrix performed to apply algorithms `finQT` and `DR`.

Even in this case the CPU time of algorithm `infQT` is negligible and almost independent of t, while for the other algorithms it grows significantly. Concerning

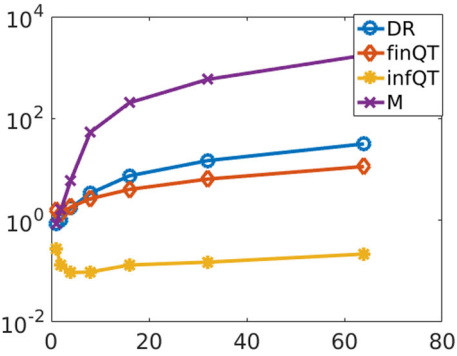

Fig. 2 CPU times in the approximation of $y = \exp(A)e$ for $A = tQ$ where Q is a dense semi-infinite generator having exponential decay and $t = 2^i, i = 0, \ldots, 6$

Table 6 Error estimates in the computation of $y = \exp(A)e$ for $A = tQ$, where Q is a dense semi-infinite generator having exponential decay, and the vector e has components 1. The corresponding values of the truncation parameter N are reported in Table 7

t	1	2	4	8	16	32	64
`DR`	4.8e−13	1.4e−12	3.0e−13	5.7e−13	1.7e−12	2.9e−11	7.2e−10
`finQT`	3.5e−13	6.9e−13	1.4e−12	2.8e−12	5.5e−12	1.1e−11	2.2e−11
`infQT`	2.4e−14	4.7e−14	8.2e−14	1.6e−13	3.2e−13	6.3e−13	1.2e−12

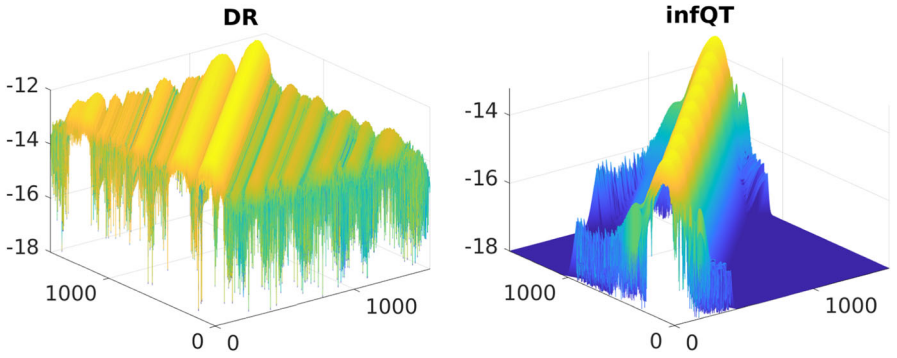

Fig. 3 Errors in the approximation of $\exp(A)$ for $A = tQ$, where Q is a dense semi-infinite generator having exponential decay, where $t = 2$. Left: algorithm based on displacement rank; right: algorithm based on quasi-Toeplitz matrices

Table 7 Lower and upper bandwidth k_-, k_+ of the Toeplitz part, numerical sizes m, n and rank r of the correction F in $\exp(A) = T(\exp(a)) + F$, where $A = tQ$ and Q is an infinite matrix with exponential decay

t	1	2	4	8	16	32	64
k_-	861	1076	1403	1914	2728	4063	6313
k_+	371	470	607	792	1023	1283	1538
m	755	959	1201	1674	2439	3695	5832
n	316	404	540	1144	1938	2961	4245
r	10	12	13	13	13	12	10
N	1232	1546	2010	2706	3751	5346	7851

In the last line, N is the size of the truncated matrix to which the other algorithms are applied

the approximation errors, from Table 6 it turns out that the algorithm `infQT` is the one which performs better than the other algorithms. In particular `infQT` has a more uniform distribution of the errors with respect to the parameter t while `DR` slightly deteriorates if t takes large values. The better performance of `infQT` is also illustrated by Fig. 3 which shows that the larger errors generated by `DR` are more uniformly distributed in all the matrix, while for `infQT` not only the errors are smaller, but the most part of them is much below the machine precision, and the largest errors are more concentrated along the main diagonal.

7.2 Second derivative

A second test concerns the $n \times n$ matrix $A_n = \theta \text{trid}_n(1, -2, 1)$ which, for $\theta = (n+1)^2$ provides the finite differences discretization of the second derivative of a sufficiently regular function. In the semi-discretization solution of the heat equation $u_{xx}(x,t) - \gamma u_t(x,t) = 0$, $\gamma > 0$, with initial conditions $u(x,0) = b(x)$ and boundary conditions $u(0,t) = u(1,t) = 0$, the vector $v^{(t)} = (v_i^{(t)})$ defined by the recurrence $v^{(t+\Delta_t)} = \exp(\theta \text{trid}_n(1,-2,1))v^{(t)}$, $\theta = \frac{\Delta_t}{\gamma \Delta_x^2}$, where $v^{(0)} = (v_i^{(0)})$, $v_i^{(0)} = b(i\Delta_x)$, provides approximation of the solution $u(i\Delta_x, t + \Delta_t)$. Choosing Δ_t of the order of $\gamma \Delta_x$, leads to computing $\exp((n+1)\text{trid}_n(1,-2,1))$. Moreover, updating the vector $v^{(t+\Delta_t)}$ involves the multiplication of the matrix exponential and a vector.

In the experiments we have chosen $\Delta_t = \gamma \Delta_x$ so that the matrix to exponentiate is $A_n = (n+1)\text{trid}_n(1,-2,1)$. We computed $\exp(A_n)$ for $n = 2^i, i = 9, \ldots, 15$, by using the algorithms `infQT, finQT, DR, M`. We multiplied a random vector v by these matrices, moreover, we computed the product $\exp(A_n)v$ by means of the algorithm `MV`. In the case of algorithm `M`, for the memory problems due to the full storage of the matrix, we have performed the experiments for $n \leq 2^{13}$.

Figure 4 displays the cpu times of the different algorithms for computing $\exp(A_n)$ and the total time for computing the product $\exp(A_n)v$, for several values of n. Table 8 reports the errors in approximating the vector $y = \exp(A_n)v$. The errors in the entries of $\exp(A_n)$ are distributed differently for algorithms `DR` and `QT`. Figure 5 reports the plot of these errors for $n = 512$.

Also for this test, algorithm `infQT` is much faster than the other algorithms. In fact, for the matrix $\exp(A_n)$ the bandwidth, and the size of the correction F have a

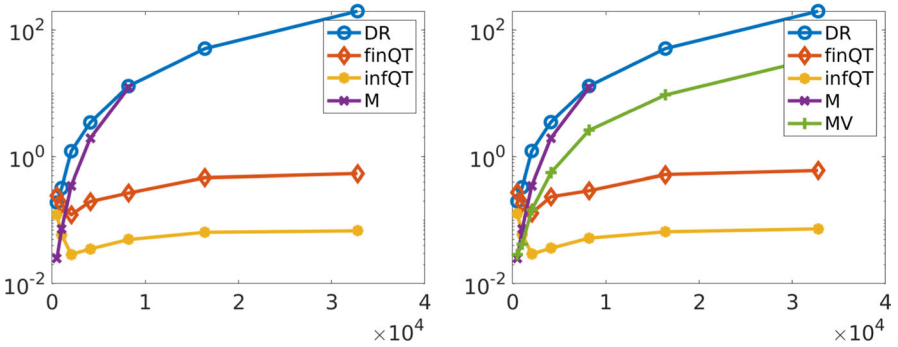

Fig. 4 CPU times in the approximation of $\exp(A_n)$ (left) and of $y = \exp(A_n)v$ (right) for the $n \times n$ matrix $A_n = (n+1)\text{trid}_n(1,-2,1)$, and a random vector v for $n = 2^i$, $i = 9, \ldots, 16$

Table 8 Errors in computing $y = \exp(A_n)v$ for $A_n = (n+1)\text{trid}_n(1,-2,1)$

n	512	1024	2048	4096	8192
DR	1.6e−12	2.0e−12	2.5e−12	8.7e−12	1.2e−11
finQT	5.0e−12	7.6e−12	1.6e−11	2.6e−11	4.8e−11
infexp	6.1e−13	1.1e−12	5.3e−12	6.5e−12	1.6e−11
MV	1.5e−13	2.1e−13	2.5e−12	2.4e−12	6.6e−12

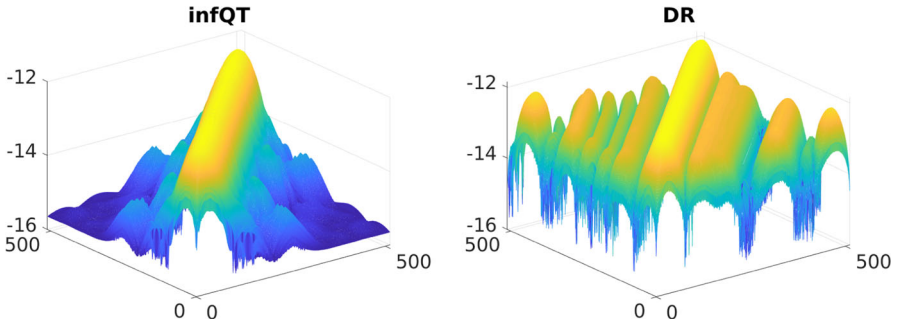

Fig. 5 Errors in the approximation of $\exp(A_n)$ for the $n \times n$ matrix $A_n = (n+1)\text{trid}_n(1,-2,1)$, $n = 512$, computed with the algorithms infQT and DR

slow growth with respect to n while the rank of F seems independent of n as shown in Table 9. The dependence on n of the cpu time for infQT is due to the cost of multiplying the QT-matrix $\exp(A_n)$ and the vector v. This computation, performed with FFT, has a cost which grows as $O(n \log n)$. Observe also that actual values of the bandwidth are in accordance with the values obtained with the analysis performed in Sect. 4.1 reported in Table 1. From the point of view of errors, the different algorithms perform similarly with a slight better performance of infQT and MV.

Table 9 Numerical bandwidth, size and rank of the correction in the infinite matrix $\exp((n+1)\mathrm{trid}(1,-2,1) = T(\exp(a)) + E$, for $a(z) = (n+1)(z^{-1}-2+z)$, where the threshold $\epsilon = 2.2204\mathrm{e}{-16}$ is used

n	512	1024	2048	4096	8192	16,384	32,768
Bandwidth	273	385	544	769	1088	1538	2174
Size (E)	287	406	574	812	1148	1624	2296
Rank (E)	15	15	15	15	15	15	15

Table 10 CPU time in seconds and corresponding errors for the $n \times n$ Merton matrix

n	1000	2000	4000	8000	16,000	32,000	64,000
`M`	0.23	1.7	13.9	116	–	–	–
`DR`	0.3	0.8	3.4	12.9	48.2	185.9	*
`finQT`	0.57	1.1	2.4	7.0	17.3	40.0	91.4
`DR` err	2.9e−11	3.8e−11	1.8e−10	1.3e−9			
`finQT` err	2.3e−11	9.2e−11	2.9e−10	1.2e−9			

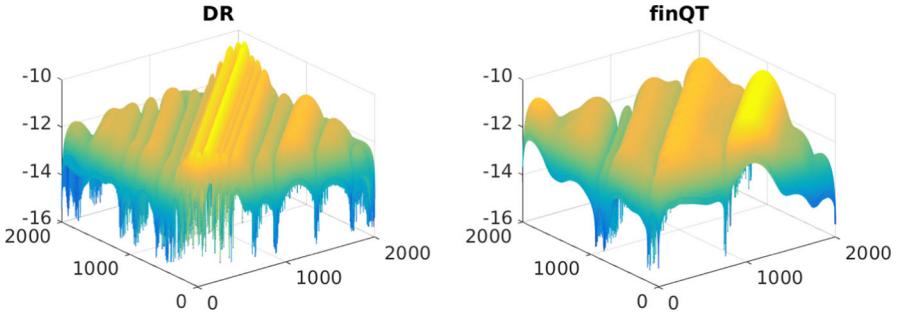

Fig. 6 Errors in the approximation of $\exp(A)$ for the Merton problem. Left: algorithm based on displacement rank; right: algorithm based on quasi-Toeplitz matrices

7.3 The Merton problem

The last test with finite matrices concerns the Merton problem used in [19] as benchmark. In this test, the matrix sequence T_n is not generated by a single symbol $a(z)$ but each matrix T_n is the truncation of an infinite matrix with its own symbol $a_n(z)$. The exponential of T_n has a correction whose size largely exceeds n. Therefore algorithm `infQT` does not apply, but `finQT` can still be used.

In Table 10 we report the CPU time needed by the algorithms `M`, `finQT` and `DR` together with the approximation error. A "*" denotes breakdown due to lack of memory. From this table it clearly turns out that the time of the Matlab function `expm` grows as $O(n^3)$, the time of the `DR` algorithm grows as $O(n^2)$ as pointed out in [19], while the time of `finQT` has a cost which grows slightly more than linearly with n. In terms of time, our algorithm outperforms the algorithm of [19] for moderately large values of n. Concerning the errors, computed up to size $n = 8000$ for memory

reasons, we can see that the two algorithms perform similarly. Finally, in Fig. 6 we report the plot of the errors obtained in the two cases. We may see that the errors show a similar distribution with respect to (i, j). The Matlab tests, as well as the evaluation of the error norm, have been halted for $n > 8000$ due to lack of memory. The function `expmt` has a breakdown for $n \geq 64{,}000$ for lack of memory.

Acknowledgements The authors wish to thank Robert Luce for providing the software for computing the matrix exponential of a finite Toeplitz matrix based on the displacement rank and the anonymous referees who provided useful suggestions and remarks which helped to improve the presentation of the paper.

References

1. Al-Mohy, A.H., Higham, N.J.: Computing the action of the matrix exponential, with an application to exponential integrators. SIAM J. Sci. Comput. **33**(2), 488–511 (2011). https://doi.org/10.1137/100788860
2. Bini, D., Dendievel, S., Latouche, G., Meini, B.: Computing the exponential of large block-triangular block-Toeplitz matrices encountered in fluid queues. Linear Algebra Appl. **502**, 387–419 (2016). https://doi.org/10.1016/j.laa.2015.03.035
3. Bini, D.A., Massei, S., Meini, B.: On functions of quasi-Toeplitz matrices. Mat. Sb. **208**(11), 56–74 (2017). https://doi.org/10.4213/sm8864
4. Bini, D.A., Massei, S., Meini, B.: Semi-infinite quasi-Toeplitz matrices with applications to QBD stochastic processes. Math. Comput. **87**(314), 2811–2830 (2018). https://doi.org/10.1090/mcom/3301
5. Bini, D.A., Massei, S., Robol, L.: Quasi-Toeplitz matrix arithmetic: a Matlab toolbox. Numer. Algorithms (2018). https://doi.org/10.1007/s11075-018-0571-6
6. Böttcher, A., Grudsky, S.M.: Toeplitz Matrices, Asymptotic Linear Algebra, and Functional Analysis. Birkhäuser Verlag, Basel (2000). https://doi.org/10.1007/978-3-0348-8395-5
7. Böttcher, A., Grusky, S.M.: Spectral Properties of Band Toeplitz Matrices. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2005)
8. Böttcher, A., Silbermann, B.: Introduction to Large Truncated Toeplitz Matrices. Springer, Berlin (2012)
9. Dendievel, S., Latouche, G.: Approximations for time-dependent distributions in Markovian fluid models. Methodol. Comput. Appl. Probab. **19**, 285–309 (2016). https://doi.org/10.1007/s11009-016-9480-0
10. Fayolle, G., Iasnogorodski, R., Malyshev, V.: Random Walks in the Quarter-Plane. Springer, Berlin (1999)
11. Gavrilyuk, I.P., Hackbusch, W., Khoromskij, B.N.: $\mathscr{H}$-matrix approximation for the operator exponential with applications. Numer. Math. **92**(1), 83–111 (2002). https://doi.org/10.1007/s002110100360
12. Gavrilyuk, I.P., Makarov, V.L.: Exponentially convergent algorithms for the operator exponential with applications to inhomogeneous problems in Banach spaces. SIAM J. Numer. Anal. **43**(5), 2144–2171 (2005). https://doi.org/10.1137/040611045
13. Grimm, V.: Resolvent Krylov subspace approximation to operator functions. BIT **52**(3), 639–659 (2012). https://doi.org/10.1007/s10543-011-0367-8
14. Halko, N., Martinsson, P.G., Tropp, J.A.: Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. SIAM Rev. **53**(2), 217–288 (2011)
15. Henrici, P.: Applied and Computational Complex Analysis, vol. 1. Wiley, New York (1974)
16. Higham, N.J.: Functions of Matrices: Theory and Computation. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2008)
17. Hochbruck, M., Ostermann, A.: Exponential integrators. Acta Numer. **19**, 209–286 (2010). https://doi.org/10.1017/S0962492910000048
18. Iserles, A.: How large is the exponential of a banded matrix? Dedicated to John Butcher. New Zealand J. Math. **29**(2), 177–192 (2000)
19. Kressner, D., Luce, R.: Fast computation of the matrix exponential for a Toeplitz matrix. SIAM J. Matrix Anal. Appl. **39**(1), 23–47 (2018). https://doi.org/10.1137/16M1083633

20. Kreyszig, E.: Introductory Functional Analysis with Applications. Wiley Classics Library. Wiley, New York (1989)
21. Latouche, G., Ramaswami, V.: Introduction to Matrix Analytic Methods in Stochastic Modeling. ASA-SIAM Series on Statistics and Applied Probability. SIAM, Philadelphia (1999)
22. Lee, S.T., Pang, H.K., Sun, H.W.: Shift-invert Arnoldi approximation to the Toeplitz matrix exponential. SIAM J. Sci. Comput. **32**(2), 774–792 (2010). https://doi.org/10.1137/090758064
23. Motyer, A.J., Taylor, P.G.: Decay rates for quasi-birth-and-death processes with countably many phases and tridiagonal block generators. Adv. Appl. Probab. **38**, 522–544 (2006)
24. Neuts, M.F.: Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach. Courier Corporation, North Chelmsford (1981)
25. Paige, C.C.: Bidiagonalization of matrices and solutions of the linear equations. SIAM J. Numer. Anal. **11**, 197–209 (1974). https://doi.org/10.1137/0711019
26. Pang, H.K., Sun, H.W.: Shift-invert Lanczos method for the symmetric positive semidefinite Toeplitz matrix exponential. Numer. Linear Algebra Appl. **18**(3), 603–614 (2011). https://doi.org/10.1002/nla.747
27. Sakuma, Y., Miyazawa, M.: On the effect of finite buffer truncation in a two-node Jackson network. Stoch. Models **12**, 143–164 (2005)
28. Sericola, B.: Markov Chains. Theory, Algorithms and Applications. Applied Stochastic Methods Series. ISTE, London; Wiley, Hoboken (2013). https://doi.org/10.1002/9781118731543
29. Shao, M.: On the finite section method for computing exponentials of doubly-infinite skew-Hermitian matrices. Linear Algebra Appl. **451**, 65–96 (2014)
30. Takahashi, Y., Fujimoto, K., Makimoto, N.: Geometric decay of the steady-state probabilities in a Quasi-Birth-Death process with a countable number of phases. Stoch. Models **14**, 368–391 (2001)
31. Trefethen, L.N., Weideman, J.A.C.: The exponentially convergent trapezoidal rule. SIAM Rev. **56**(3), 385–458 (2014). https://doi.org/10.1137/130932132
32. Wu, G., Feng, T.T., Wei, Y.: An inexact shift-and-invert Arnoldi algorithm for Toeplitz matrix exponential. Numer. Linear Algebra Appl. **22**(4), 777–792 (2015). https://doi.org/10.1002/nla.1992