Noname manuscript No.
(will be inserted by the editor)

Estimating the error in matrix function approximations

Nasim Eshghi · Lothar Reichel

the date of receipt and acceptance should be inserted later

Abstract The need to compute matrix functions of the form $f(A)v$, where $A \in \mathbb{R}^{N\times N}$ is a large symmetric matrix, f is a function such that $f(A)$ is well defined, and $v \neq 0$ is a vector, arises in many applications. This paper is concerned with the situation when A is so large that the evaluation of $f(A)$ is prohibitively expensive. Then an approximation of $f(A)v$ often is computed by applying a few, say $1 \leq n \ll N$, steps of the symmetric Lanczos process to A with initial vector v to determine a symmetric tridiagonal matrix $T_n \in \mathbb{R}^{n\times n}$ and a matrix $V_n \in \mathbb{R}^{N\times n}$, whose orthonormal columns span a Krylov subspace. The expression $V_n f(T_n)e_1\|v\|$ furnishes an approximation of $f(A)v$. The evaluation of $f(T_n)$ is inexpensive, because the matrix T_n is small. It is important to be able to estimate the error in the computed approximation. This paper describes a novel approach that is based on a technique proposed by Spalević for estimating the error in Gauss quadrature rules.

Keywords matrix function, symmetric Lanczos process, error estimation

1 Introduction

Many problems in science and engineering require the evaluation of expressions of the form

$$f(A)v, \tag{1}$$

where $A \in \mathbb{R}^{N\times N}$ is a large symmetric matrix, $v \in \mathbb{R}^N\backslash\{0\}$ is a vector, and f is a function such that $f(A)$ is well defined and continuous on the convex hull of the spectrum of A. Applications arise, for instance, in the solution of partial differential equations, ill-posed problems, and network analysis; see, e.g., [2–4, 6, 12].

N. Eshghi
Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA. E-mail: neshghi@kent.edu

L. Reichel
Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA. E-mail: reichel@math.kent.edu

One way to define $f(A)$ is via the spectral factorization

$$A = U\Lambda U^T, \qquad \Lambda = \mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_N], \tag{2}$$

where the λ_j are eigenvalues of A, the columns of the matrix $U \in \mathbb{R}^{N\times N}$ are suitably normalized orthogonal eigenvectors (see below for details about the normalization), and the superscript ^{T} denotes transposition. Then

$$f(A) = Uf(\Lambda)U^T, \qquad f(\Lambda) = \mathrm{diag}[f(\lambda_1), f(\lambda_2), \ldots, f(\lambda_N)]; \tag{3}$$

see, e.g., Golub and Van Loan [22] or Higham [23] for several ways to define matrix functions.

When the matrix A is of small to moderate size, we can easily compute the spectral factorization (2) and evaluate $f(A)$ according to (3). This and many other techniques for evaluating functions of small to moderately sized matrices are described by Higham [23]. When $f(A)$ is known, it is straightforward to evaluate the expression (1).

However, when the matrix A is large, the computation of the spectral factorization (2) may be too expensive to be practical. Also other techniques that require factorization of A to compute $f(A)$ may be too expensive when A is large without an exploitable structure. In this situation, the expression (1) commonly is approximated by using a Lanczos decomposition. Application of a few, say $1 \leq n \ll N$, steps of the symmetric Lanczos process to A with initial vector v gives the Lanczos decomposition

$$AV_n = V_nT_n + \beta_n v_{n+1}e_n^T, \tag{4}$$

where the columns of the matrix $V_n = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^{N\times n}$ form an orthonormal basis for the Krylov subspace

$$\mathcal{K}_n(A, v) := \mathrm{span}\{v, Av, \ldots, A^{n-1}v\}$$

with $v_1 = v/\|v\|$, the unit vector $v_{n+1} \in \mathbb{R}^N$ is such that $V_n^T v_{n+1} = 0$, and $\beta_n \geq 0$. Throughout this paper e_n denotes the n^{th} column of an identity matrix of suitable order, $\|\cdot\|$ stands for the Euclidean vector norm, and the symmetric tridiagonal matrix

$$T_n = \begin{bmatrix} \alpha_0 & \beta_1 & & & 0 \\ \beta_1 & \alpha_1 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{n-2} & \alpha_{n-2} & \beta_{n-1} \\ 0 & & & \beta_{n-1} & \alpha_{n-1} \end{bmatrix} \in \mathbb{R}^{n\times n} \tag{5}$$

is an orthogonal section of A. We assume that the number of steps, n, of the Lanczos process is small enough so that the decomposition (4) with the stated properties exists; see, e.g., Golub and Van Loan [22], Golub and Meurant [21], or Saad [31] for detailed discussions on the symmetric Lanczos process, which is described by Algorithm 1.

The Lanczos decomposition (4) is used to approximate the expressions (1) by

$$V_n f(T_n) e_1 \|v\|; \tag{6}$$

see, e.g., [2, 18, 24] for discussions and error bounds. Hence, the difficult problem of evaluating $f(A)$ for a large matrix A is replaced by the much simpler task of

Algorithm 1 The symmetric Lanczos process.

Input: symmetric matrix $A \in \mathbb{R}^{n \times n}$, initial unit vector $v \in \mathbb{R}^n$,
number of steps m.
$v_0 := 0 \in \mathbb{R}^n$, $\beta_0 := 0$, $v_1 := v$
for $j = 1$ **to** m
  $w := Av_j - v_{j-1}\beta_{j-1}$
  $\alpha_j := v_j^T w$
  $w := w - v_j \alpha_j$
  $\beta_j := \|w\|$; $v_{j+1} := w/\beta_j$
end for
Output: Entries $\alpha_1, \alpha_2, \ldots, \alpha_m$ and $\beta_1, \beta_2, \ldots, \beta_m$ of the matrix (5).

first applying n steps of the Lanczos process to A and then computing $f(T_n)$. One way to evaluate the latter expression is to compute the spectral factorization of T_n and use a formula analogous to (3). The existence of $f(T_n)$ is secured when f is continuous on the convex hull of the spectrum of A.

We are interested in determining an easily computable estimate of the norm of the approximation error

$$E_n(f) := \|f(A)v - V_n f(T_n) e_1 \|v\|\|. \tag{7}$$

An elegant recent paper by Frommer and Schweitzer [18] provides easily computable error bounds when f is a Stieltjes function. However, for general functions f error bounds for matrix function approximations are not available or impractical to compute; see, e.g., [2,12,17,24] for discussions of a variety of error bounds.

It is the purpose of the present paper to describe a novel approach to estimate the error norm (7). Our approach is based on a technique for estimating the error in Gauss quadrature rules proposed by Spalević [32]. The symmetric tridiagonal matrix (5) can be associated with an n-node Gauss quadrature rule $\mathcal{G}_n$ with respect to a measure that is determined by the matrix A and vector v. Since quadrature rules are important in what follows, we outline the connection. A thorough discussion is provided by Golub and Meurant [21]. The representation (3) yields

$$v^T f(A)v = v^T U f(\Lambda) U^T v = \sum_{j=1}^{N} f(\lambda_j)\mu_j^2 = \int f(t) d\mu(t), \tag{8}$$

where $U^T v = [\mu_1, \mu_2, \ldots, \mu_N]^T$. The above sum can be interpreted as a Stieltjes integral associated with a measure $d\mu$ shown in the right-hand side of (8); the distribution function affiliated with this measure can be chosen to be nondecreasing and piece-wise constant with jumps μ_j^2 at the eigenvalues λ_j. The Gauss rule $\mathcal{G}_n$ mentioned above is associated with the measure $d\mu$. Its nodes are the eigenvalues of the matrix (5) and its weights are the square of the first components of suitably normalized eigenvectors of this matrix. We recall that the eigenvectors of the matrix (5) are orthogonal; they should be normalized so that their Euclidean norm is $\|v\|$. Since the matrix (5) is an orthogonal section of A, its eigenvalues are contained in the convex hull of the eigenvalues of A. We refer to the difference

$$\mathcal{E}_n(f) := v^T f(A)v - \mathcal{G}_n(f) \tag{9}$$

as the quadrature error in $\mathcal{G}_n(f)$. It vanishes for $f \in \mathbb{P}_{2n-1}$, where $\mathbb{P}_{2n-1}$ denotes the set of all polynomials of degree at most $2n-1$. This property makes $\mathcal{G}_n$ a Gaussian quadrature rule; see Golub and Meurant [21] for details.

We also will be interested in the $(n-1)$-node Gauss rule, $\mathcal{G}_{n-1}$, affiliated with the measure $d\mu$. Its nodes are the eigenvalues of the leading $(n-1)\times(n-1)$ principal submatrix T_{n-1} of (5), and its weights are the square of the first components of eigenvectors of T_{n-1}, that are normalized to have Euclidean norm $\|v\|$.

Spalević [32] described a $(2n-1)$-node quadrature rule, which we denote by $\mathcal{S}_{2n-1}$, for the estimation of the quadrature error $\mathcal{E}_{n-1}(f)$. We will refer to $\mathcal{S}_{2n-1}$ as a $(2n-1)$-node Spalević rule. This rule is associated with a symmetric tridiagonal matrix $S_{2n-1} \in \mathbb{R}^{(2n-1)\times(2n-1)}$, similarly as the Gauss rule $\mathcal{G}_n$ is associated with the matrix (5). It is straightforward to determine the matrix S_{2n-1} from the matrix (5) and the coefficient β_n in (4). We describe in Section 2 how a leading principal submatrix of S_{2n-1} can be applied to provide an estimate of the error (7). A few computed examples are presented in Section 3 and concluding remarks can be found in Section 4.

2 A new error estimation method

Assume for the moment that we apply $n+q$ steps of the symmetric Lanczos process to A with initial vector v for some $q > 1$. This gives the decomposition

$$AV_{n+q} = V_{n+q}T_{n+q} + \beta_{n+q}v_{n+q+1}e_{n+q}^T, \tag{10}$$

which is analogous to (4). In particular, the matrix $V_{n+q} = [v_1, v_2, \ldots, v_{n+q}] \in \mathbb{R}^{N\times(n+q)}$ has orthonormal columns with $v_1 = v/\|v\|$. Its first n columns agree with those of the matrix V_n in (4). The unit vector v_{n+q+1} is such that $V_{n+q}^T v_{n+q+1} = 0$, and $\beta_{n+q} \geq 0$. The matrix T_{n+q} is symmetric and tridiagonal with the $n\times n$ leading principal submatrix (5). We assume the decomposition (10) to exist and we can use it to evaluate the expression

$$V_{n+q}f(T_{n+q})e_1\|v\|, \tag{11}$$

which is analogous to (6).

We will need the matrix

$$\begin{bmatrix} f(T_n) & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n+q)\times(n+q)},$$

which is obtained by zero-padding of $f(T_n) \in \mathbb{R}^{n\times n}$. Typically, the expression (11) is a more accurate approximation of the matrix function (1) than (6). This suggests that it may be possible to estimate the error norm (7) as

$$\begin{aligned} &\|V_{n+q}f(T_{n+q})e_1\|v\| - V_nf(T_n)e_1\|v\|\| \\ = &\left\| V_{n+q}f(T_{n+q})e_1\|v\| - V_{n+q}\begin{bmatrix} f(T_n) & 0 \\ 0 & 0 \end{bmatrix} e_1\|v\| \right\|, \end{aligned} \tag{12}$$

which simplifies to

$$\widetilde{E}_{n+q,q}(f) := \left\| f(T_{n+q})e_1 - \begin{bmatrix} f(T_n) & 0 \\ 0 & 0 \end{bmatrix} e_1 \right\| \|v\|, \tag{13}$$

where we note that this expression does not require knowledge of the columns of the matrix V_{n+q} in (10). However, the need to evaluate $f(T_{n+q})$ makes it necessary to carry out $n+q > n$ steps of the Lanczos process to determine the entries of the matrix T_{n+q}.

Estimates of the quadrature error (9) that are analogous to the error estimate (13) have been applied by Golub and Meurant [20,26]. Specifically, they estimate the error (9) by $\mathcal{G}_{n+q}(f)-\mathcal{G}_n(f)$, and use this estimate to assess the error in iterates determined by the conjugate gradient method. To determine an accurate estimate of the error (9), the parameter q has to be sufficiently large. In computed examples reported in [20], this parameter is chosen to be $2 \leq q \leq 20$, while Meurant [26] uses $q = 5$. It is known that $q = 1$ is not large enough; see Clenshaw and Curtis [7] for a discussion on the estimation of the error (9) by $\mathcal{G}_{n+1}(f) - \mathcal{G}_n(f)$.

We would like to determine an estimate of the expression (13) without having to carry out more than the n steps of the Lanczos process required to evaluate the expression (6). To describe our approach for achieving this, we first define the truncated Spalević rules. These rules have been discussed and analyzed in [8,9, 29]. They are truncated versions of the Spalević rule presented in [32].

Introduce the symmetric tridiagonal matrix

$$\overline{T}_q = \begin{bmatrix} \alpha_{n-2} & \beta_{n-2} & & & 0 \\ \beta_{n-2} & \alpha_{n-3} & \beta_{n-3} & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{n-q+1} & \alpha_{n-q} & \beta_{n-q} \\ 0 & & & \beta_{n-q} & \alpha_{n-q-1} \end{bmatrix} \in \mathbb{R}^{q \times q} \tag{14}$$

for some $1 \leq q < n$. This matrix is obtained by first reversing the rows and columns of the $(n-1)\times(n-1)$ leading principal submatrix of (5), and then retaining the first q rows and columns of the matrix so obtained. Define the symmetric tridiagonal matrix determined by concatenating the matrices (5) and (14), and including the coefficient β_n in (4),

$$\widetilde{T}_{n+q,q} = \begin{bmatrix} T_n & \beta_n e_n \\ \beta_n e_n^T & \overline{T}_q \end{bmatrix} \in \mathbb{R}^{(n+q)\times(n+q)}. \tag{15}$$

Note that all entries of this matrix can be determined from the Lanczos decomposition (4).

If $q = n-1$, then (15) is the symmetric tridiagonal matrix S_{2n-1} associated with the $(2n-1)$-node Spalević quadrature rule for estimating the error in the Gauss rule $\mathcal{G}_{n-1}$ commented on at the end of Section 1. The quadrature rule $\mathcal{G}_{n-1}$ integrates all polynomials in $\mathbb{P}_{2n-3}$ exactly; see, e.g., [21]. The associated $(2n-1)$-node Spalević rule is exact for all polynomials in $\mathbb{P}_{2n}$ and is described in [32–34]. When $1 \leq q < n-1$, the resulting matrix (15) is associated with an $(n+q)$-node truncated Spalević rule. These rules also are exact for all polynomials in $\mathbb{P}_{2n}$; see, e.g., [29].

Our error estimate for (1) is based on replacing the matrix T_{n+q} in (13) by (15). This yields the error estimate

$$E_{n,q}(f) := \left\| f(\widetilde{T}_{n+q,q})e_1 - \begin{bmatrix} f(T_n) & 0 \\ 0 & 0 \end{bmatrix} e_1 \right\| \|v\|. \tag{16}$$

We remark that other approaches to estimating the error in (1) based on quadrature rules that are commonly used for error estimation also can be developed. For instance, a classical approach to estimate the error in the Gauss rule $\mathcal{G}_{n-1}(f)$ is to use the associated $(2n-1)$-node Gauss–Kronrod rule. The latter rule corresponds to a real symmetric tridiagonal matrix of order $2n-1$ when the Gauss–Kronrod rule has real nodes and the $n-1$ Gauss nodes interlace the n non-Gauss nodes; see Notaris [28] for a recent discussion on Gauss–Kronrod quadrature. However, Gauss–Kronrod rules with nodes with this property are not guaranteed to exist. Moreover, they are more complicated to compute than the Spalević rules; see [1,5,25] for computational aspects.

The following results shed some light on the accuracy of the expressions in (16).

Theorem 1 *We have*

$$f(A)v = V_n f(T_n) e_1 \|v\| \qquad \forall f \in \mathbb{P}_{n-1}, \tag{17}$$

where the right-hand side is defined by (6)*, and*

$$f(A)v = V_{n+q} f(\widetilde{T}_{n+q,q}) e_1 \|v\| \qquad \forall f \in \mathbb{P}_n, \tag{18}$$

when $1 \leq q \leq n-1$*, where the matrix* $\widetilde{T}_{n+q,q}$ *is given by* (15) *and the matrix* V_{n+q} *is defined in* (11)*. It follows that the error estimate* (16) *satisfies*

$$E_{n,q}(f) = \|f(A)v - V_n f(T_n) e_1 \|v\|\| \qquad \forall f \in \mathbb{P}_n, \tag{19}$$

i.e., the error estimate is exact for $f \in \mathbb{P}_n$*.*

Proof The property (17) is well known. It can easily be shown for increasing powers $f(t) = t^j$, $j = 0, 1, \ldots, n-1$; see, e.g., [11,30]. We turn to (18) and obtain from (10) that

$$AV_{n+q} = V_{n+q}\widetilde{T}_{n+q,q} + V_{n+q}(T_{n+q} - \widetilde{T}_{n+q,q}) + \beta_{n+q} v_{n+q+1} e_{n+q}^T.$$

It follows that

$$Av_1 = V_{n+q}\widetilde{T}_{n+q,q} e_1 + V_{n+q}(T_{n+q} - \widetilde{T}_{n+q,q}) e_1 + \beta_{n+q} v_{n+q+1} e_{n+q}^T e_1.$$

If $n \geq 1$ and $n+q > 1$, then this expression simplifies to

$$Av_1 = V_{n+q}\widetilde{T}_{n+q,q} e_1, \tag{20}$$

i.e., property (18) holds for $f \in \mathbb{P}_1$.

Multiplying (20) from the left by A and substituting (10) gives

$$\begin{aligned} A^2 v_1 &= AV_{n+q}\widetilde{T}_{n+q,q} e_1 \\ &= (V_{n+q}T_{n+q} + \beta_{n+q} v_{n+q+1} e_{n+q}^T)\widetilde{T}_{n+q,q} e_1 \\ &= V_{n+q}T_{n+q}\widetilde{T}_{n+q,q} e_1 \\ &= V_{n+q}\widetilde{T}_{n+q,q}^2 e_1, \end{aligned}$$

where the last two equalities hold if $n+q>2$ and $n\geq 2$. Thus, under these conditions equation (18) is valid for $f\in\mathbb{P}_2$. Similarly,

$$\begin{aligned}A^3v_1 &= AV_{n+q}\widetilde{T}^2_{n+q,q}e_1\\ &= (V_{n+q}T_{n+q}+\beta_{n+q}v_{n+q+1}e^T_{n+q})\widetilde{T}^2_{n+q,q}e_1\\ &= V_{n+q}T_{n+q}\widetilde{T}^2_{n+q,q}e_1\\ &= V_{n+q}\widetilde{T}^3_{n+q,q}e_1,\end{aligned}$$

where the last two equalities hold if $n+q>3$ and $n\geq 3$. We can similarly show that

$$A^kv_1 = V_{n+q}\widetilde{T}^k_{n+q,q}e_1,$$

provided that $n+q>k$ and $n\geq k$. This shows (18).

The property (19) can be shown by expressing (16) in the form

$$E_{n,q}(f)=\left\|V_{n+q}f(\widetilde{T}_{n+q,q})e_1\|v\|-V_nf(T_n)e_1\|v\|\right\|, \tag{21}$$

which is obtained by replacing the matrix T_{n+q} in (12) by $\widetilde{T}_{n+q,q}$. Equation (19) now follows from (18).

We remark that a special case of this result is stated in [16, Corollary 2.3]. Moreover, the expression (18) may hold for a larger set of polynomials when the matrices T_n, $n=1,2,\dots$, in (4) have a special structure. For instance, if T_{n+q} is a tridiagonal Toeplitz matrix, then $\widetilde{T}_{n+q,q}=T_{n+q}$, and it follows that

$$V_{n+q}f(\widetilde{T}_{n+q,q})e_1\|v\|=V_{n+q}f(T_{n+q})e_1\|v\|=f(A)v \qquad \forall f\in\mathbb{P}_{n+q-1},$$

where the last equality follows from (17). It follows that higher accuracy than (18) also can be achieved if only a trailing principal submatrix of the matrix (5) is Toeplitz. Measures $d\mu$ that give rise to orthogonal polynomials with recursion coefficients that determine a symmetric tridiagonal matrix (5) of this kind are said to belong to the class $\mathcal{M}_\ell^{(\alpha,\beta)}[a,b]$, which was introduced by Gautschi and Notaris [19]. Subsequently, this kind of measures were used by Spalević [34] and more recently by Djukić et al. [10] in their investigations of Spalević quadrature rules.

Our error estimation technique is based on replacing the matrix T_{n+q} in the Lanczos decomposition (10) by the matrix $\widetilde{T}_{n+q,q}$. It is natural to ask whether there is a matrix $\widetilde{A}\in\mathbb{R}^{N\times N}$ such that

$$\widetilde{A}V_{n+q}=V_{n+q}\widetilde{T}_{n+q,q}+\beta_{n+q}v_{n+q+1}e^T_{n+q}$$

is a Lanczos decomposition. Here the matrix V_{n+q}, vector v_{n+q+1}, and scalar β_{n+q} are the same as in (10). The following result shows how a matrix $\widetilde{A}$ can be determined from A if the matrix $\widehat{T}_{n+q}$ differs from T_{n+q} in one entry only. Such rank-one modifications can be applied repeatedly to determine the matrix (15) from T_{n+q}.

Theorem 2 *Assume that the Lanczos decomposition* (10) *exists and let* $\widehat{T}_{n+q} = T_{n+q} + \alpha e_i e_j^T$ *for some* $\alpha \in \mathbb{R}$ *and* $1 \leq i, j \leq n+q$. *Then*

$$\widetilde{A} V_{n+q} = V_{n+q} \widehat{T}_{n+q} + \beta_{n+q} v_{n+q+1} e_{n+q}^T, \tag{22}$$

with the matrix $V_{n+q} = [v_1, v_2, \ldots, v_{n+q}]$, *vector* v_{n+q+1}, *and scalar* β_{n+q} *the same as in* (10), *is a Lanczos decomposition of the matrix*

$$\widetilde{A} = A + \alpha v_i v_j^T. \tag{23}$$

Proof We obtain from (10) that

$$A V_{n+q} = V_{n+q} \widehat{T}_{n+q} - \alpha v_i e_j^T + \beta_{n+q} v_{n+q+1} e_{n+q}^T,$$

which we express as

$$(A + \alpha v_i v_j^T) V_{n+q} = V_{n+q} \widehat{T}_{n+q} + \beta_{n+q} v_{n+q+1} e_{n+q}^T.$$

Thus, the decomposition (22) holds for $\widetilde{A}$ given by (23).

The theorem shows that the matrix $\widetilde{T}_{n+q,q}$ defined by (15) can be determined by applying $n+q$ steps of the Lanczos process to some matrix. However, we do not use this fact in the computations.

3 Numerical examples

We present a few computed examples that illustrate the accuracy of the proposed error estimates. All computations were carried out using MATLAB R2016b on a 64-bit Lenovo personal computer with approximately 15 significant decimal digits.

Table 1 Example 3.1: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N \times N}$ a symmetric positive definite Toeplitz matrix, $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$, and $f(t) = \exp(t)$ when $n = 5$ and $q \in \{1, 2, 3\}$.

N	$\|f(A)v\|$		$q = 1$	$q = 2$	$q = 3$
200	$2.74836 \cdot 10^{1}$	$\frac{E_n(f)}{\|f(A)\|}$	$2.90175 \cdot 10^{-8}$	$2.90175 \cdot 10^{-8}$	$2.90175 \cdot 10^{-8}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$2.90006 \cdot 10^{-8}$	$2.90175 \cdot 10^{-8}$	$2.90175 \cdot 10^{-8}$
2000	$8.70859 \cdot 10^{1}$	$\frac{E_n(f)}{\|f(A)\|}$	$9.25347 \cdot 10^{-9}$	$9.25347 \cdot 10^{-9}$	$9.25347 \cdot 10^{-9}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$9.24808 \cdot 10^{-9}$	$9.25347 \cdot 10^{-9}$	$9.25347 \cdot 10^{-9}$
5000	$1.37713 \cdot 10^{2}$	$\frac{E_n(f)}{\|f(A)\|}$	$5.85533 \cdot 10^{-9}$	$5.85533 \cdot 10^{-9}$	$5.85533 \cdot 10^{-9}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$5.85192 \cdot 10^{-9}$	$5.85533 \cdot 10^{-9}$	$5.85533 \cdot 10^{-9}$

Example 3.1. Let $A \in \mathbb{R}^{N \times N}$ be the symmetric positive definite Toeplitz matrix with first row $[1/3, \ldots, 1/3^N]$. The initial vector is chosen to be $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$ unless explicitly stated otherwise. We will let $N \in \{200, 2000, 5000\}$. The matrices $A \in \mathbb{R}^{N \times N}$ so defined have the smallest eigenvalue about 1/6.

We first apply $n = 5$ steps of the symmetric Lanczos process to A. This yields the Lanczos decomposition (4). Even though A is Toeplitz, the symmetric tridiagonal matrix T_n in the Lanczos decomposition is not. Define the matrix (15) for

Table 2 Example 3.1: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive definite Toeplitz matrix, $v \in \mathbb{R}^N$ is a random vector, and $f(t) = \exp(t)$ when $n = 5$ and $q \in \{1, 2, 3\}$.

N	$\|f(A)v\|$		$q = 1$	$q = 2$	$q = 3$
200	$2.00852 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$3.08261 \cdot 10^{-7}$	$3.08261 \cdot 10^{-7}$	$3.08261 \cdot 10^{-7}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$3.08010 \cdot 10^{-7}$	$3.08261 \cdot 10^{-7}$	$3.08261 \cdot 10^{-7}$
2000	$6.63092 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$3.56874 \cdot 10^{-7}$	$3.56874 \cdot 10^{-7}$	$3.56874 \cdot 10^{-7}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$3.56669 \cdot 10^{-7}$	$3.56874 \cdot 10^{-7}$	$3.56874 \cdot 10^{-7}$
5000	$1.02805 \cdot 10^2$	$\frac{E_n(f)}{\|f(A)\|}$	$3.61104 \cdot 10^{-7}$	$3.61104 \cdot 10^{-7}$	$3.61104 \cdot 10^{-7}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$3.60899 \cdot 10^{-7}$	$3.61104 \cdot 10^{-7}$	$3.61104 \cdot 10^{-7}$

Table 3 Example 3.1: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive definite Toeplitz matrix, $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$, and $f(t) = 1/t$ when $n = 5$ and $q \in \{1, 2, 3\}$.

N	$\|f(A)v\|$		$q = 1$	$q = 2$	$q = 3$
200	$2.13454 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$6.43076 \cdot 10^{-4}$	$6.43076 \cdot 10^{-4}$	$6.43076 \cdot 10^{-4}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$5.42303 \cdot 10^{-4}$	$6.26182 \cdot 10^{-4}$	$6.40509 \cdot 10^{-4}$
2000	$6.71240 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$2.05517 \cdot 10^{-4}$	$2.05517 \cdot 10^{-4}$	$2.05517 \cdot 10^{-4}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$1.73306 \cdot 10^{-4}$	$2.00118 \cdot 10^{-4}$	$2.04698 \cdot 10^{-4}$
5000	$1.06092 \cdot 10^2$	$\frac{E_n(f)}{\|f(A)\|}$	$1.30069 \cdot 10^{-4}$	$1.30069 \cdot 10^{-4}$	$1.30069 \cdot 10^{-4}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$1.09683 \cdot 10^{-4}$	$1.26651 \cdot 10^{-4}$	$1.29550 \cdot 10^{-4}$

Table 4 Example 3.1: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive definite Toeplitz matrix, $v \in \mathbb{R}^N$ a random vector, and $f(t) = 1/t$ when $n = 5$ and $q \in \{1, 2, 3\}$.

N	$\|f(A)v\|$		$q = 1$	$q = 2$	$q = 3$
200	$5.59057 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$4.62715 \cdot 10^{-3}$	$4.62715 \cdot 10^{-3}$	$4.62715 \cdot 10^{-4}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$3.67313 \cdot 10^{-3}$	$4.46681 \cdot 10^{-3}$	$4.60622 \cdot 10^{-3}$
2000	$1.87581 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$5.09331 \cdot 10^{-3}$	$5.09331 \cdot 10^{-3}$	$5.09331 \cdot 10^{-3}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$4.31285 \cdot 10^{-3}$	$4.96157 \cdot 10^{-3}$	$5.07309 \cdot 10^{-3}$
5000	$2.89799 \cdot 10^2$	$\frac{E_n(f)}{\|f(A)\|}$	$5.13185 \cdot 10^{-3}$	$5.13185 \cdot 10^{-3}$	$5.13185 \cdot 10^{-3}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$4.34610 \cdot 10^{-3}$	$4.99922 \cdot 10^{-3}$	$5.11160 \cdot 10^{-3}$

Table 5 Example 3.1: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive definite Toeplitz matrix, $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$, and $f(t) = \sqrt{t - 0.1}$ when $n = 5$ and $q \in \{1, 2, 3\}$.

N	$\|f(A)v\|$		$q = 1$	$q = 2$	$q = 3$
2000	$3.36576 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$1.11024 \cdot 10^{-5}$	$1.11024 \cdot 10^{-5}$	$1.11024 \cdot 10^{-5}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$9.09738 \cdot 10^{-6}$	$1.06697 \cdot 10^{-5}$	$1.10102 \cdot 10^{-5}$

$q \in \{1, 2, 3\}$. Table 1 shows the relative approximation errors $E_n(f)/\|f(A)\|$ defined

Table 6 Example 3.1: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive definite Toeplitz matrix, $v \in \mathbb{R}^N$ random vector, and $f(t) = \sqrt{t-0.1}$ when $n = 5$ and $q \in \{1, 2, 3\}$.

N	$\|f(A)v\|$		$q = 1$	$q = 2$	$q = 3$
2000	$2.23382 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$1.26399 \cdot 10^{-3}$	$1.26399 \cdot 10^{-3}$	$1.26399 \cdot 10^{-3}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$1.02055 \cdot 10^{-3}$	$1.20801 \cdot 10^{-3}$	$1.25135 \cdot 10^{-3}$

by (7) and relative error estimates $E_{n,q}(f)/\|f(A)\|$ given by (16) for $f(t) = \exp(t)$. The errors estimates are seen to be quite accurate for all values of N and q.

Table 2 differs from Table 1 only in that the initial vector $v \in \mathbb{R}^N$ has normally distributed random entries with mean zero and variance one. The value of $f(A)v$ differs for this table from that for Table 1, but the computed error estimates $E_{n,q}(f)/\|f(A)\|$ for all values of N and q can be seen to be quite accurate.

Tables 3 and 5 are analogous to Table 1 and shows error estimates $E_{n,q}(f)/\|f(A)\|$ for $f(t) = 1/t$ and $f(t) = \sqrt{t-0.1}$, respectively; in the interest of brevity the latter table only displays results for $N = 2000$. Tables 4 and 6 are analogous to Tables 3 and 5; they differ in the initial vector, which is of the same kind as in Table 2.

The error in the computed approximation of $f(A)v$ can be expected to converge to zero slower as n increases, the closer there is an eigenvalue of A to a singularity of f. Slow convergence of the error to zero also may result in less accurate error estimates than when convergence is fast. Since $f(t) = \exp(t)$ has no singularity in the finite complex plane, we expect faster convergence of the error to zero with increasing n for this function than for the other functions of this example. The distance between the singularity of f and the closest eigenvalue of A is about 1/6 for $f(t) = 1/t$, and about 1/15 for $f(t) = \sqrt{t-0.1}$ for all values of N in this example. Nevertheless, Tables 1-6 show the error estimates to be quite accurate for all functions and for both kinds of initial vectors v used. The accuracy of the error estimates is seen to increase with q.

We conclude this example with a comparison of the accuracy of the error estimates $\widetilde{E}_{n+q,q}(f)$ and $E_{n,q}(f)$ defined by (13) and (16), respectively. Note that the computation of $\widetilde{E}_{n+q,q}(f)$ demands that $n+q$ steps of the Lanczos process be carried out, while the calculation of $E_{n,q}(f)$ only requires n steps of the Lanczos process. Table 7 displays the error estimates $\widetilde{E}_{n+q,q}(f)$ for $n = 5$ and $q \in \{1, 3\}$ for three functions f and initial vector $v = [1, 1, \ldots, 1]^T$. Table 8 differs from Table 7 only in the choice of initial vector v for the Lanczos process; it is a random vector with normally distributed entries with mean zero and variance one in the latter table.

Table 7 Example 3.1: Relative error estimates $\widetilde{E}_{n+q,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive definite Toeplitz matrix, $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$, and several functions f for $N = 2000$, $n = 5$, and $q \in \{1, 3\}$.

$f(t)$	$\frac{\widetilde{E}_{6,1}(f)}{\|f(A)\|}$	$\frac{\widetilde{E}_{8,3}(f)}{\|f(A)\|}$
$\exp(t)$	$9.24808 \cdot 10^{-9}$	$9.25347 \cdot 10^{-9}$
$1/t$	$1.73306 \cdot 10^{-4}$	$2.04698 \cdot 10^{-4}$
$\sqrt{t-0.1}$	$9.09738 \cdot 10^{-6}$	$1.10102 \cdot 10^{-5}$

Table 8 Example 3.1: Relative error estimates $\widetilde{E}_{n+q,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive definite Toeplitz matrix, $v \in \mathbb{R}^N$ a random vector, and several functions f for $N = 2000$, $n = 5$, and $q \in \{1, 3\}$.

$f(t)$	$\frac{\widetilde{E}_{6,1}(f)}{\|f(A)\|}$	$\frac{\widetilde{E}_{8,3}(f)}{\|f(A)\|}$
$\exp(t)$	$3.56669 \cdot 10^{-7}$	$3.56874 \cdot 10^{-7}$
$1/t$	$4.13285 \cdot 10^{-3}$	$5.07309 \cdot 10^{-3}$
$\sqrt{t-0.1}$	$1.02055 \cdot 10^{-3}$	$1.25135 \cdot 10^{-3}$

Comparing results for $f(t) = \exp(t)$ in Tables 1 and 7 shows the error estimate $E_{5,1}(f)$ to be as accurate as $\widetilde{E}_{6,1}(f)$, and the estimate $E_{5,3}(f)$ to be as accurate as $\widetilde{E}_{8,3}(f)$, but the estimates $E_{5,1}(f)$ and $E_{5,3}(f)$ only require the evaluation of 5 steps of the Lanczos process, while the estimates $\widetilde{E}_{6,1}(f)$ and $\widetilde{E}_{8,3}(f)$ demand the execution of 6 and 8 Lanczos steps, respectively. Analogous conclusions can be drawn for the other functions in Table 7. Tables 2 and 8 shows similar results. In summary, the error estimates (16) are as accurate as the error estimates (13), but require less computational effort to evaluate. □

It is clear that executing more steps of the Lanczos process demands more computing time. For instance, for the matrices in Table 7 the execution time increases by 24% when increasing the number of steps of the Lanczos process from 5 to 10, but the total CPU time is small, less than 0.1 second for both 5 or 10 steps. However, the CPU time grows quickly with the matrix size. An example that employs hierarchical compression of $\mathcal{H}^2$-matrices is described in [15, Example 1.1]. The evaluation of one matrix-vector product with an $\mathcal{H}^2$-matrix of order $N = 262146$ required 26.6 minutes of CPU time on a laptop computer; see [15] for details. Here it suffices to point out that there are problems for which it is important to keep the number of matrix-vector product evaluations as small as possible. For those problems, the evaluation of the estimates (13) is more attractive than the evaluation of the estimates (16). Unfortunately, we cannot evaluate the exact error (7) for problems of very large size and, thus, cannot assess the accuracy of the estimates (21). We therefore in the following restrict ourselves to problems that are small enough to allow the evaluation of (7) in moderate time.

Table 9 Example 3.2: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)/\|f(A)\|$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive semidefinite matrix, $v = [1, 1, \ldots, 1]^T$, and $f(t) = \exp(t)$, when $n = 5$ and $q = n - 1$.

N	$\|f(A)v\|$		
200	$1.812147 \cdot 10^{1}$	$\frac{E_n(f)}{\|f(A)\|}$	$1.66291 \cdot 10^{-9}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$1.66265 \cdot 10^{-9}$
2000	$5.74098 \cdot 10^{1}$	$\frac{E_n(f)}{\|f(A)\|}$	$5.32845 \cdot 10^{-10}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$5.32759 \cdot 10^{-10}$
5000	$9.07839 \cdot 10^{1}$	$\frac{E_n(f)}{\|f(A)\|}$	$3.37257 \cdot 10^{-10}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$3.37202 \cdot 10^{-10}$

Table 10 Example 3.2: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)$ for $A \in \mathbb{R}^{N\times N}$ a symmetric positive semidefinite matrix, $v = [1, 1, \ldots, 1]^T$, and $f(t) = 1/(t+0.1)$, when $n = 5$ and $q = n - 1$.

N	$\|f(A)v\|$		
200	$4.09089 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$7.93734 \cdot 10^{-4}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$8.07596 \cdot 10^{-4}$
2000	$1.27946 \cdot 10^2$	$\frac{E_n(f)}{\|f(A)\|}$	$2.55772 \cdot 10^{-4}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$2.60275 \cdot 10^{-4}$
5000	$2.02150 \cdot 10^2$	$\frac{E_n(f)}{\|f(A)\|}$	$1.61959 \cdot 10^{-4}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$1.64812 \cdot 10^{-4}$

Example 3.2. Let $A \in \mathbb{R}^{N\times N}$ be a symmetric positive semidefinite matrix. It is defined by $A = (B - \lambda_1 I)^2$, where $B \in \mathbb{R}^{N\times N}$ is a symmetric Toeplitz matrix with first row $[1/3, \ldots, 1/3^N]$ and largest eigenvalue λ_1. We apply $n = 5$ steps of the symmetric Lanczos process to A with initial vector $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$ and compute the relative errors $E_5(f)/\|f(A)\|$ and relative error estimates $E_{5,4}(f)/\|f(A)\|$ for the functions $f(t) = \exp(t)$ and $f(t) = 1/(t + 0.1)$. Thus, we choose q to be as large as possible, i.e., $n - 1$.

Tables 9 and 10 display the relative errors and relative error estimates for $N \in \{200, 2000, 5000\}$. The tables show the relative errors in the approximation (6) to be larger for $f(t) = 1/(t + 0.1)$ than for the exponential function, but the relative error estimates are seen to be quite accurate for both functions. □

Table 11 Example 3.3: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,q}(f)/\|f(A)\|$ for symmetric positive definite Toeplitz matrices $A \in \mathbb{R}^{N\times N}$, $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$, for $n = \{5, 10\}$ and $f(t) = \exp(t)$.

N	$\|f(A)v\|$		$n = 5$	$n = 10$
200	$2.80339 \cdot 10^2$	$\frac{E_n(f)}{\|f(A)\|}$	$6.72185 \cdot 10^{-5}$	$2.54432 \cdot 10^{-10}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$6.72198 \cdot 10^{-5}$	$2.54436 \cdot 10^{-10}$
2000	$8.97085 \cdot 10^2$	$\frac{E_n(f)}{\|f(A)\|}$	$2.14220 \cdot 10^{-5}$	$8.12756 \cdot 10^{-11}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$2.14220 \cdot 10^{-5}$	$8.12755 \cdot 10^{-11}$
5000	$1.41952 \cdot 10^3$	$\frac{E_n(f)}{\|f(A)\|}$	$1.35532 \cdot 10^{-5}$	$5.14173 \cdot 10^{-11}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$1.35532 \cdot 10^{-5}$	$5.14173 \cdot 10^{-11}$

Example 3.3. We let $A \in \mathbb{R}^{N\times N}$ with $N \in \{200, 2000, 5000\}$ be symmetric positive definite Toeplitz matrices with first row $[1, 1/2, \ldots, 1/2^{(N-1)}]$. The aim of this example is to illustrate how the error in the function approximation (6) decreases when increasing the number of Lanczos steps n. Note that the symmetric tridiagonal matrices T_n determined by the Lanczos process are not Toeplitz.

Table 11 shows results for the exponential function $f(t) = \exp(t)$ and $n \in \{5, 10\}$. The parameter q is chosen as large as possible, i.e., $q = n - 1$. The error in the function approximation (6) is seen to be much smaller for $n = 10$ than for $n = 5$; the error estimates (16) are accurate for both values of n. Table 12 is

Table 12 Example 3.3: Relative error of errors $E_n(f)$ and error estimates $E_{n,q}(f)$ for symmetric positive definite Toeplitz matrices $A \in \mathbb{R}^{N\times N}$, $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$, for $n = \{5, 10\}$ and $f(t) = \sqrt{t}$.

N	$\|f(A)v\|$		$n = 5$	$n = 10$
200	$2.44131 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$4.65712 \cdot 10^{-5}$	$4.42032 \cdot 10^{-7}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$4.61394 \cdot 10^{-5}$	$4.41994 \cdot 10^{-7}$
2000	$7.74338 \cdot 10^1$	$\frac{E_n(f)}{\|f(A)\|}$	$1.47928 \cdot 10^{-5}$	$1.40452 \cdot 10^{-7}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$1.46549 \cdot 10^{-5}$	$1.40450 \cdot 10^{-7}$
5000	$1.22458 \cdot 10^2$	$\frac{E_n(f)}{\|f(A)\|}$	$9.35800 \cdot 10^{-6}$	$8.88454 \cdot 10^{-8}$
		$\frac{E_{n,q}(f)}{\|f(A)\|}$	$9.27069 \cdot 10^{-6}$	$8.88441 \cdot 10^{-8}$

analogous to Table 11 and shows results for $f(t) = \sqrt{t}$. The errors are larger in this table than in Table 11, but the error estimates are quite accurate. □

Our last example is concerned with the analysis of large networks. A network is represented by a graph $G = \{\mathcal{V}, \mathcal{E}\}$ that is defined by a set of vertices $\mathcal{V}$ and a set of edges $\mathcal{E}$. We assume G to be a simple connected graph with N nodes, i.e., G is undirected, unweighted, and without self-loops and multiple edges. The adjacency matrix $A = [a_{ij}]_{i,j=1}^N$ associated with G has the entry $a_{ij} = 1$ if there is an edge between node i and node j, and $a_{ij} = 0$ otherwise. The adjacency matrix is symmetric. Typically, the number of edges in a graph is much smaller than N^2. Therefore, adjacency matrices generally are sparse; see, e.g., [13, 27] for more details.

A walk of length k in an undirected graph is a sequence of $k + 1$ vertices $v_1, v_2, \ldots, v_{k+1}$ such that there is an edge between vertex v_i and vertex v_{i+1} for $i = 1, 2, \ldots, k$. Vertices and edges in a walk may be repeated. The entry $a_{ij}^{(\ell)}$ of the matrix $A^\ell = [a_{ij}^{(\ell)}]_{i,j=1}^N$ is equal to the number of walks of length ℓ between node i and node j.

Consider a function

$$f(A) = \sum_{\ell=0}^{\infty} c_\ell A^\ell \tag{24}$$

with positive coefficients c_ℓ chosen to guarantee convergence. The entry $[f(A)]_{ij}$ of $f(A)$ can be interpreted as a measure of the ease of traveling between nodes i and j. The term $c_0 I$ has no specific meaning and is introduced for convenience. The coefficients c_ℓ are generally chosen to be strictly decreasing functions of ℓ, since this models that short walks are more important than long walks. A popular choice is $c_\ell = 1/\ell!$ for all $\ell \geq 0$, which yields $f(A) = \exp(A)$; see, e.g., Estrada and Higham [14] for a nice introduction to the application of matrix functions in network analysis. The expression $f(A)v$ with $v = [1, 1, \ldots, 1]^T$ can be used to measure the relative importance of nodes in a network. If the entry $[f(A)v]_i$ is relatively large, when compared to the other entries of the vector $f(A)v$, then this indicates that node i is important in the network; see, e.g., Benzi and Klymko [4]. A nice recent discussion on the use of matrix functions in network analysis is provided by Benzi and Boito [3]. The following example determines error estimates for approximations of $f(A)v$.

Table 13 Example 3.4: Relative errors $E_n(f)/\|f(A)\|$ and relative error estimates $E_{n,n-1}(f)/\|f(A)\|$ for an adjacency matrix $A \in \mathbb{R}^{N\times N}$ defined by a network and $v = [1,1,\dots,1]^T \in \mathbb{R}^N$, for $n = \{5,10,15\}$ and $f(t) = \exp(t)$.

N	$\|f(A)v\|$		$n=5$	$n=10$	$n=15$
2114	$2.53067 \cdot 10^4$	$\frac{E_n(f)}{\|f(A)\|}$	$1.33460 \cdot 10^{-1}$	$1.06073 \cdot 10^{-3}$	$9.30448 \cdot 10^{-7}$
		$\frac{E_{n,n-1}(f)}{\|f(A)\|}$	$1.39036 \cdot 10^{-1}$	$1.10350 \cdot 10^{-3}$	$9.43031 \cdot 10^{-7}$

Example 3.4. Consider the network yeast, which is represented by an undirected graph with 2114 vertices and 4480 edges. It describes the protein-protein interaction of yeast. Each node represents a protein and each edge represents an interaction between two proteins. The adjacency matrix $A \in \mathbb{R}^{2114\times 2114}$ for this graph is symmetric and is available at [35].

Table 13 shows the relative error of the function approximations (6) and relative error estimates obtained from (16) for the exponential function $f(t) = \exp(t)$ when $q = n-1$ and $n \in \{5,10,15\}$. The relative error in the function approximations (6) can be seen to decrease rapidly as n increases. The relative error estimates (16) are quite accurate for all values of n. □

4 Conclusion

A new method for estimating the error in approximations of functions of symmetric matrices is presented. Tables 1-13 show the proposed error estimates to be quite accurate. For large matrices A, the dominating computational effort for computing the matrix function approximation (6) is the evaluation of n steps of the symmetric Lanczos process. An attraction of the error estimates (16) is that their computation does not require that additional Lanczos steps be carried out.

Acknowledgment

The authors would like to thank the referees for comments that lead to improvements of the presentation. Research by L.R. was supported in part by NSF grant DMS-1720259.

References

1. G. S. Ammar, D. Calvetti, and L. Reichel, *Computation of Gauss–Kronrod quadrature rules with non-positive weights*, Electron. Trans. Numer. Anal., 9 (1999), pp. 26–38.
2. B. Beckermann and L. Reichel, *Error estimation and evaluation of matrix functions via the Faber transform*, SIAM J. Numer. Anal., 47 (2009), pp. 3849–3883.
3. M. Benzi and P. Boito, *Matrix functions in network analysis*, GAMM Mitteilungen, 43 (2020), Art. e202000012.
4. M. Benzi and C. Klymko, *Total communicability as a centrality measure*, J. Complex Networks, 1 (2013), pp. 1–26.
5. D. Calvetti, G. H. Golub, W. B. Gragg, and L. Reichel, *Computation of Gauss–Kronrod quadrature rules*, Math. Comp., 69 (2000), pp. 1035–1052.
6. D. Calvetti and L. Reichel, *Lanczos-based exponential filtering for discrete ill-posed problems*, Numer. Algorithms, 29 (2002), pp. 45–65.

7. C. W. Clenshaw and A. R. Curtis, *A method for numerical integration on an automatic computer*, Numer. Math., 2 (1960), pp. 197–205.
8. D. Lj. Djukić, L. Reichel, and M. M. Spalević, *Truncated generalized averaged Gauss quadrature rules*, J. Comput. Appl. Math., 308 (2016), pp. 408–418.
9. D. Lj. Djukić, L. Reichel, M. M. Spalević, and J. D. Tomanović, *Internality of generalized averaged Gaussian quadratures and their truncated variants with Bernstein–Szegő weights*, Electron. Trans. Numer. Anal., 45 (2016), pp. 405–419.
10. D. Lj. Djukić, L. Reichel, M. M. Spalević, and J. D. Tomanović, *Internality of generalized averaged Gaussian quadrature rules and truncated variants for modified Chebyshev measures of the second kind*, J. Comput. Appl. Math., 345 (2019), pp. 70–85.
11. V. L. Druskin and L. A. Knizhnerman, *Two polynomial methods for the computation of functions of symmetric matrices*, USSR Comput. Math. Math. Phys., 29 (1989), pp. 112–121.
12. V. Druskin, L. Knizhnerman, and M. Zaslavsky, *Solution of large scale evolutionary problems using rational Krylov subspaces with optimized shifts*, SIAM J. Sci. Comput., 31 (2009), pp. 3760–3780.
13. E. Estrada, *The Structure of Complex Networks: Theory and Applications*, Oxford University Press, Oxford, 2012.
14. E. Estrada and D. J. Higham, *Network properties revealed through matrix functions*, SIAM Rev., 52 (2010), pp. 696–714.
15. N. Eshghi, T. Mach, and L. Reichel, *New matrix function approximations and quadrature rules based on the Arnoldi process*, J. Comput. Appl. Math., 391 (2021), Art. 113442.
16. N. Eshghi, L. Reichel, and M. M. Spalević, *Enhanced matrix function approximation*, Electron. Trans. Numer. Anal., 47 (2017), pp. 197–205.
17. A. Frommer, *Monotone convergence of the Lanczos approximations to matrix functions of Hermitian matrices*, Electron. Trans. Numer. Anal., 35 (2009), pp. 118–128.
18. A. Frommer and M. Schweitzer, *Error bounds and estimates for Krylov subspace approximations of Stieltjes matrix functions*, BIT Numer. Math., 56 (2016), pp. 865–892.
19. W. Gautschi and S. E. Notaris, *Stieltjes polynomials and related quadrature formulae for a class of weight functions*, Math. Comp., 65 (1996), pp. 1257–1268.
20. G. H. Golub and G. Meurant, *Matrices, moments and quadrature II: How to compute the norm of the error in iterative methods*, BIT Numer. Math., 37 (1997), pp. 687–705.
21. G. H. Golub and G. Meurant, *Matrices, Moments and Quadrature with Applications*, Princeton University Press, Princeton, 2010.
22. G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., Johns Hopkins University Press, Baltimore, 2013.
23. N. J. Higham, *Functions of Matrices*, SIAM, Philadelphia, 2008.
24. M. Hochbruck and C. Lubich, *On Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 34 (1997), pp. 1911–1925.
25. D. P. Laurie, *Calculation of Gauss–Kronrod quadrature rules*, Math. Comp., 66 (1997), pp. 1133–1145.
26. G. Meurant, *Numerical experiments in computing bounds for the norm of the error in the preconditioned conjugate gradient algorithm*, Numer. Algorithms, 22 (1999), pp. 353–365.
27. M. E. J. Newman, *Networks: An Introduction*, Oxford University Press, Oxford, 2010.
28. S. E. Notaris, *Gauss–Kronrod quadrature formulae – a survey of fifty years of research*, Electron. Trans. Numer. Anal., 45 (2016), pp. 371–404.
29. L. Reichel, M. M. Spalević, and T. Tang, *Generalized averaged Gauss quadrature rules for the approximation of matrix functionals*, BIT Numer. Math., 56 (2016), pp. 1045–1067.
30. Y. Saad, *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 29 (1992), pp. 209–228.
31. Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.
32. M. M. Spalević, *On generalized averaged Gaussian formulas*, Math. Comp., 76 (2007), pp. 1483–1492.
33. M. M. Spalević, *A note on generalized averaged Gaussian formulas*, Numer. Algorithms, 46 (2007), pp. 253–264.
34. M. M. Spalević, *On generalized averaged Gaussian formulas, II*, Math. Comp., 86 (2017), pp. 1877–1885.
35. S. Sun, L. Ling, N. Zhang, G. Li, and R. Chen, *Topological structure analysis of the protein-protein interaction network in budding yeast*, Nucleic Acids Research, 31 (2003), pp. 2443–2450.