Electronic Transactions on Numerical Analysis.
Volume 47, pp. 197–205, 2017.

ISSN 1068–9613.

ETNA
Kent State University and
Johann Radon Institute (RICAM)

ENHANCED MATRIX FUNCTION APPROXIMATION*

NASIM ESHGHI†, LOTHAR REICHEL†, AND MIODRAG M. SPALEVIĆ‡

Abstract. Matrix functions of the form $f(A)v$, where A is a large symmetric matrix, f is a function, and $v \neq 0$ is a vector, are commonly approximated by first applying a few, say n, steps of the symmetric Lanczos process to A with the initial vector v in order to determine an orthogonal section of A. The latter is represented by a (small) $n \times n$ tridiagonal matrix to which f is applied. This approach uses the n first Lanczos vectors provided by the Lanczos process. However, n steps of the Lanczos process yield $n+1$ Lanczos vectors. This paper discusses how the $(n+1)$st Lanczos vector can be used to improve the quality of the computed approximation of $f(A)v$. Also the approximation of expressions of the form $v^T f(A)v$ is considered.

Key words. matrix function, symmetric Lanczos process, Gauss quadrature

AMS subject classifications. 65D32, 65F10, 65F60

1. Introduction.

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

$$f(A)v \quad \text{or} \quad v^T f(A)v, \tag{1.1}$$

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. Here, the superscript ^{T} denotes transposition. Applications for the expressions (1.1) include the solution of systems of ordinary differential equations [5, 10], network analysis [2, 12], and the solution of ill-posed problems [3].

Consider the spectral factorization

$$A = U\Lambda U^T, \qquad \Lambda = \operatorname{diag}[\lambda_1, \lambda_2, \ldots, \lambda_N], \tag{1.2}$$

where λ_j are the eigenvalues of A and the matrix $U \in \mathbb{R}^{N\times N}$ is orthogonal. We assume that f is continuous on the convex hull of the spectrum of A. We may define $f(A)$ with the aid of the spectral factorization (1.2), i.e.,

$$f(A) = U f(\Lambda) U^T. \tag{1.3}$$

When A is of small to moderate size, we can easily compute the spectral factorization (1.2) and evaluate $f(A)$ according to (1.3). Knowing $f(A)$, it is straightforward to compute (1.1). However, when the matrix A is very large, the computation of the spectral factorization (1.2) may be too expensive to be practical. Also other techniques that require a factorization of A to compute $f(A)$ typically are too expensive when A is very large.

When A is large, the expressions in (1.1) are commonly approximated by first applying a few, say $n \ll N$, steps of the symmetric Lanczos process to A with the initial vector v to determine the Lanczos decomposition

$$AV_n = V_n T_n + \beta_n v_{n+1} e_n^T, \tag{1.4}$$

*Received November 16, 2017. Accepted November 28, 2017. Published online on December 14, 2017. Recommended by K. Jbilou. The research of M. M. Spalević is supported in part by the Serbian Ministry of Education, Science and Technological Development (Research Project: "Methods of numerical and nonlinear analysis with applications" (# 174002)). The research of L. Reichel is supported in part by NSF grants DMS-1720259 and DMS-1729509.

†Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA (neshghi@kent.edu, reichel@math.kent.edu).

‡Department of Mathematics, Faculty of Mechanical Engineering, University of Belgrade, Kraljice Marije 16, 11120 Belgrade 35, Serbia (mspalevic@mas.bg.ac.rs).

ETNA
Kent State University and
Johann Radon Institute (RICAM)

where $V_n = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^{N\times n}$ has orthonormal columns with the initial column $v_1 = v/\|v\|$, 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_j denotes the jth column of an identity matrix of suitable order, and $\|\cdot\|$ stands for the Euclidean vector norm. The matrix T_n is symmetric and tridiagonal,

$$T_n = \begin{bmatrix} \alpha_0 & \beta_1 & & \mathrm{O} \\ \beta_1 & \alpha_1 & \beta_2 & \\ & \ddots & \ddots & \ddots \\ \mathrm{O} & & \beta_{n-1} & \alpha_{n-1} \end{bmatrix} \in \mathbb{R}^{n\times n}; \tag{1.5}$$

it 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 (1.4) with the stated properties exists, typically $1 \leq n \ll N$; see, e.g., Golub and Meurant [8] or Saad [13] for discussions on the symmetric Lanczos process. Having computed the Lanczos decomposition (1.4), it is used to approximate the expressions (1.1) by

$$V_n f(T_n) e_1 \|v\| \quad \text{or} \quad \|v\|^2 e_1^T f(T_n) e_1, \tag{1.6}$$

respectively; see, e.g., [1, 2, 6, 8, 10]. Hence, the evaluation of $f(A)$ is replaced by the much simpler task of computing $f(T_n)$. Higham [9] discusses and analyzes many numerical methods for the evaluation functions of a small matrix. We remark that the existence of $f(T_n)$ is secured, e.g., when f is continuous on the convex hull of the spectrum of A.

Error bounds and error estimates for the left-hand side of (1.6) can be found in [1, 6]. The expression on the right-hand side of (1.6) can be interpreted as a Gauss quadrature rule. Our discussion follows Golub and Meurant [8]. Let $[\omega_1, \omega_2, \ldots, \omega_N] = v^T U$. Substituting the spectral factorization (1.2) into the right-hand side expression of (1.1) yields

$$v^T f(A) v = \sum_{j=1}^{N} f(\lambda_j)\omega_j^2 = \int f(t)\, d\omega(t) =: \mathcal{I}(f), \tag{1.7}$$

where $\omega(t)$ is a nondecreasing piecewise constant distribution function with jumps at the eigenvalues λ_j of A and $d\omega(t)$ is the associated measure. Thus, the left-hand side of (1.7) may be considered a Stieltjes integral determined by the nonnegative measure $d\omega$ with support in the convex hull of the spectrum of A. We define an inner product associated with this measure for polynomials of sufficiently low degree,

$$(f, g) := (f(A)v)^T g(A) v = \sum_{j=1}^{N} f(\lambda_j) g(\lambda_j) \omega_j^2 = \int f(t) g(t)\, d\omega(t) = \mathcal{I}(fg).$$

Substituting the spectral factorization of T_n into the right-hand side expression of (1.6), one can see that it is an n-point quadrature rule

$$\mathcal{G}_n(f) = \|v\|^2 e_1^T f(T_n) e_1 \tag{1.8}$$

for approximating the integral (1.7). Golub and Meurant [8] show that this quadrature rule is a Gauss rule, i.e.,

$$\mathcal{I}(f) = \mathcal{G}_n(f), \qquad \forall f \in \mathbb{P}_{2n-1}, \tag{1.9}$$

where $\mathbb{P}_{2n-1}$ denotes the set of all polynomials of degree at most $2n-1$. This observation can be used to determine bounds or estimates for the quadrature error $\mathcal{G}_n(f) - \mathcal{I}(f)$ when $f \notin \mathbb{P}_{2n-1}$; see [4, 8, 11, 12, 14].

ETNA
Kent State University and
Johann Radon Institute (RICAM)

In the situation when the matrix A is large, the computational effort required for the evaluation of the Lanczos decomposition (1.4) is dominated by n matrix-vector product evaluations with A (see, e.g., [8, 13] for details on the symmetric Lanczos process), and each matrix-vector product evaluation is expensive. We therefore would like to compute accurate approximations of the expressions (1.1) by carrying out as few steps n of the Lanczos process as possible.

Neither the constant β_n nor the vector v_{n+1} in (1.4) are used in the expressions (1.6). It is the purpose of the present paper to explore how these quantities can be applied to obtain more accurate approximations of the expressions (1.1) than (1.6). This is described in Section 2. A few computed examples that illustrate the performance of the proposed schemes are presented in Section 3, and Section 4 contains concluding remarks and discusses extensions.

While the focus of this paper is the evaluation of expressions of the form (1.1), the technique discussed also may be of interest for computing quadrature rules for approximating integrals

$$\mathcal{I}(f) = \int f(t)d\omega(t)$$

that are defined by a nonnegative measure with support on the real axis for which the recursion coefficients of the associated orthogonal polynomials are not explicitly known. Gautschi [7] discusses the computation of recursion coefficients and Gauss quadrature rules in this situation. Our approach for determining quadrature rules may be attractive when it is expensive to compute the recursion coefficients.

2. New methods for approximating matrix functions. If we would apply $n+1$ steps of the symmetric Lanczos process to A with the initial vector v, then we would obtain the decomposition

$$AV_{n+1} = V_{n+1}T_{n+1} + \beta_{n+1}v_{n+2}e_{n+1}^T, \tag{2.1}$$

which is analogous to (1.4). As usual, we assume that breakdown does not occur. In particular, $V_{n+1} = [v_1, v_2, \ldots, v_n, v_{n+1}] \in \mathbb{R}^{N\times(n+1)}$ has orthonormal columns with $v_1 = v/\|v\|$, and the matrix T_{n+1} is symmetric and tridiagonal,

$$T_{n+1} = \begin{bmatrix} \alpha_0 & \beta_1 & & & \mathrm{O} \\ \beta_1 & \alpha_1 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{n-1} & \alpha_{n-1} & \beta_n \\ \mathrm{O} & & & \beta_n & \alpha_n \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}. \tag{2.2}$$

Thus, the leading $N \times n$ submatrix of V_{n+1} is the matrix V_n in (1.4), and the leading $n \times n$ principal submatrix of (2.2) is the matrix (1.5). The decomposition (2.1) can be used to evaluate

$$V_{n+1}f(T_{n+1})e_1\|v\| \quad \text{and} \quad \|v\|^2 e_1^T f(T_{n+1})e_1, \tag{2.3}$$

which are analogues of (1.6). Typically, the expressions (2.3) are more accurate approximations of the matrix functions (1.1) than (1.6), but the computation of (2.3) requires the evaluation of one more matrix-vector product with A than the calculation of (1.6).

The decomposition (1.4) determines the matrix V_{n+1} in (2.3) and all entries of T_{n+1} except for the last diagonal entry α_n. This suggests that we estimate α_n by a scalar $\widehat{\alpha}_n$, define

ETNA
Kent State University and
Johann Radon Institute (RICAM)

the symmetric tridiagonal matrix

$$\widehat{T}_{n+1} = \begin{bmatrix} \alpha_0 & \beta_1 & & & \mathrm{O} \\ \beta_1 & \alpha_1 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{n-1} & \alpha_{n-1} & \beta_n \\ \mathrm{O} & & & \beta_n & \widehat{\alpha}_n \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}, \tag{2.4}$$

and replace T_{n+1} by $\widehat{T}_{n+1}$ in (2.3). This yields the approximations

$$V_{n+1} f(\widehat{T}_{n+1}) e_1 \|v\| \quad \text{and} \quad \|v\|^2 e_1^T f(\widehat{T}_{n+1}) e_1 \tag{2.5}$$

of the expressions (1.1).

We first discuss the error in the expression in the right-hand side of (2.5). Equation (2.1) yields

$$AV_{n+1} = V_{n+1}\widehat{T}_{n+1} + (\alpha_n - \widehat{\alpha}_n) v_{n+1} e_{n+1}^T + \beta_{n+1} v_{n+2} e_{n+1}^T. \tag{2.6}$$

This expression is used in the proof of the following results.

THEOREM 2.1. *Define the reduced inner products*

$$(f,g)_n = \|v\|^2 e_1^T (f(T_n))^T g(T_n) e_1$$

and

$$\langle f,g\rangle_{n+1} = \|v\|^2 e_1^T (f(\widehat{T}_{n+1}))^T g(\widehat{T}_{n+1}) e_1. \tag{2.7}$$

Then

$$(f,g)_n = (f,g), \qquad \forall fg \in \mathbb{P}_{2n-1}, \tag{2.8}$$

and

$$\langle f,g\rangle_{n+1} = (f,g), \qquad \forall fg \in \mathbb{P}_{2n}, \tag{2.9}$$

for any entry $\widehat{\alpha}_n$ of $\widehat{T}_{n+1}$.

Proof. Let $h = fg \in \mathbb{P}_{2n-1}$. Then equation (2.8) can be expressed as

$$\|v\|^2 e_1^T h(T_n) e_1 = \mathcal{I}(h).$$

This equality holds due to (1.8) and (1.9).

We turn to (2.9). It suffices to show that

$$(A^k v_1)^T (A^j v_1) = (\widehat{T}_{n+1}^k e_1)^T \widehat{T}_{n+1}^j e_1, \qquad 0 \le j,k \le n. \tag{2.10}$$

It follows from (2.6) that $Av_1 = V_{n+1}\widehat{T}_{n+1} e_1$ for $n \ge 1$. Further, the decomposition (2.6) yields

$$\begin{aligned} A^2 v_1 &= A(AV_{n+1}e_1) = A(V_{n+1}\widehat{T}_{n+1} + (\alpha_n - \widehat{\alpha}_n) v_{n+1} e_{n+1}^T + \beta_{n+1} v_{n+2} e_{n+1}^T) e_1 \\ &= AV_{n+1}\widehat{T}_{n+1} e_1 = (V_{n+1}\widehat{T}_{n+1} + (\alpha_n - \widehat{\alpha}_n) v_{n+1} e_{n+1}^T + \beta_{n+1} v_{n+2} e_{n+1}^T)\widehat{T}_{n+1} e_1 \\ &= V_{n+1}\widehat{T}_{n+1}^2 e_1, \end{aligned}$$

ETNA
Kent State University and
Johann Radon Institute (RICAM)

provided that $n \geq 2$. We obtain similarly that

$$A^k v_1 = V_{n+1} \widehat{T}_{n+1}^k e_1, \qquad k = 3, 4, \ldots, n, \tag{2.11}$$

and (2.10) follows. □

Introduce the quadrature rule

$$\widehat{\mathcal{G}}_{n+1}(f) = \|v\|^2 e_1^T f(\widehat{T}_{n+1}) e_1. \tag{2.12}$$

COROLLARY 2.2. *The quadrature rule* (2.12) *satisfies*

$$\widehat{\mathcal{G}}_{n+1}(f) = \mathcal{I}(f), \qquad \forall f \in \mathbb{P}_{2n}, \tag{2.13}$$

where $\mathcal{I}(f)$ *is defined by* (1.7)*. Moreover, using* (2.6)*, we have*

$$v_1^T A^{2n+1} v_1 = e_1^T \widehat{T}_{n+1}^{2n+1} e_1 + (\alpha_n - \widehat{\alpha}_n)\, e_1^T \widehat{T}_{n+1} e_n\, e_1^T \widehat{T}_{n+1} e_{n+1}, \tag{2.14}$$

which shows that the quadrature rule (2.13) *is exact for* $f \in \mathbb{P}_{2n+1}$ *when* $\widehat{\alpha}_n = \alpha_n$*. Further, the quadrature error* $(\mathcal{I} - \widehat{\mathcal{G}}_{n+1})(f)$ *for* $f \in \mathbb{P}_{2n+1}$ *is small in a relative sense when* $\widehat{\alpha}_n$ *is close to* α_n *or when* $e_1^T \widehat{T}_{n+1} e_n\, e_1^T \widehat{T}_{n+1} e_{n+1}$ *is small.*

Proof. Let $g, h \in \mathbb{P}_n$ and $f = gh$. Then, by (2.7) and (2.9),

$$\widehat{\mathcal{G}}_{n+1}(f) = \langle g, h \rangle_{n+1} = (g, h) = \mathcal{I}(f).$$

This shows (2.13). Equation (2.14) follows from (2.6). □

Corollary 2.2 suggests that the quadrature rule (2.12) may yield more accurate approximations of (1.7) than (1.8) for many integrands. Extensive numerical experience, some of which is reported in Section 3, indicate that this is indeed the case. We recall that the computation of the rule (2.12) can be determined by carrying out n steps of the symmetric Lanczos process, similarly as the Gauss rule (1.8). The computational effort to evaluate the quadrature rules (1.8) and (2.12) therefore is essentially the same when the matrix A is so large that the matrix-vector product evaluations required by the Lanczos process dominate the computational work.

We turn to the error in the expression on the left-hand side of (2.5). It is well-known that

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

see, e.g., [1]. The approximation on the left-hand side of (2.5) is exact for a larger class of polynomials.

COROLLARY 2.3. *The expression on the left-hand side of* (2.5) *satisfies*

$$f(A)v = V_{n+1} f(\widehat{T}_{n+1}) e_1 \|v\| \qquad \forall f \in \mathbb{P}_n.$$

Proof. The result follows from the proof of Theorem 2.1, specifically from (2.11). □

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

EXAMPLE 3.1. Let $A \in \mathbb{R}^{N \times N}$ with $N \in \{200, 2000, 5000, 10000\}$ be a symmetric Toeplitz matrix with first row $[1, 1/2, \ldots, 1/2^{(N-1)}]$, and let $v = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$. We apply n steps of the symmetric Lanczos process to A with the initial vector v. This yields the Lanczos decomposition (1.4). We choose the last diagonal entry in the matrix (2.4) to be $\widehat{\alpha}_n = \alpha_{n-1}$, where α_{n-1} is the last diagonal entry of the matrix (1.5). Table 3.1 displays

ETNA
Kent State University and
Johann Radon Institute (RICAM)

TABLE 3.1
Example 3.1: Relative error of computed approximations of $v^T f(A)v$ for $A \in \mathbb{R}^{N\times N}$ a Toeplitz matrix, $f(t) = 1/t$, and $v = [1, 1, \ldots, 1]^T$.

N		$n = 5$	$n = 10$	$n = 15$
200	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$9.57 \cdot 10^{-6}$	$9.31 \cdot 10^{-9}$	$9.06 \cdot 10^{-12}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.36 \cdot 10^{-6}$	$1.33 \cdot 10^{-9}$	$1.29 \cdot 10^{-12}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$2.39 \cdot 10^{-6}$	$2.33 \cdot 10^{-9}$	$2.26 \cdot 10^{-12}$
2000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$9.76 \cdot 10^{-7}$	$9.52 \cdot 10^{-10}$	$9.31 \cdot 10^{-13}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.39 \cdot 10^{-7}$	$1.36 \cdot 10^{-10}$	$1.34 \cdot 10^{-13}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$2.44 \cdot 10^{-7}$	$2.38 \cdot 10^{-10}$	$2.34 \cdot 10^{-13}$
5000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$3.91 \cdot 10^{-7}$	$3.81 \cdot 10^{-10}$	$3.73 \cdot 10^{-13}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$5.58 \cdot 10^{-8}$	$5.45 \cdot 10^{-11}$	$5.36 \cdot 10^{-14}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$9.76 \cdot 10^{-8}$	$9.53 \cdot 10^{-11}$	$9.34 \cdot 10^{-14}$
10000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.95 \cdot 10^{-7}$	$1.91 \cdot 10^{-10}$	$1.86 \cdot 10^{-13}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$2.79 \cdot 10^{-8}$	$2.72 \cdot 10^{-11}$	$2.66 \cdot 10^{-14}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert//\lvert\mathcal{I}(f)\rvert$	$4.88 \cdot 10^{-8}$	$4.77 \cdot 10^{-11}$	$4.66 \cdot 10^{-14}$

TABLE 3.2
Example 3.1: Relative error of computed approximations of $v^T f(A)v$ for $A \in \mathbb{R}^{N\times N}$ a Toeplitz matrix, $f(t) = \exp(t)$, and $v = [1, 1, \ldots, 1]^T$.

N		$n = 5$	$n = 10$	$n = 15$
200	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$4.88 \cdot 10^{-11}$	$3.23 \cdot 10^{-14}$	$2.52 \cdot 10^{-14}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$8.70 \cdot 10^{-13}$	$1.98 \cdot 10^{-14}$	$1.80 \cdot 10^{-14}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.60 \cdot 10^{-13}$	$2.16 \cdot 10^{-14}$	$1.44 \cdot 10^{-14}$
2000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$4.99 \cdot 10^{-12}$	$7.09 \cdot 10^{-14}$	$2.66 \cdot 10^{-14}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$8.66 \cdot 10^{-13}$	$3.37 \cdot 10^{-14}$	$2.30 \cdot 10^{-14}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.40 \cdot 10^{-13}$	$1.95 \cdot 10^{-14}$	$3.37 \cdot 10^{-14}$
5000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.99 \cdot 10^{-12}$	$5.49 \cdot 10^{-13}$	$6.02 \cdot 10^{-14}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$3.04 \cdot 10^{-13}$	$5.13 \cdot 10^{-13}$	$6.19 \cdot 10^{-13}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.24 \cdot 10^{-13}$	$5.31 \cdot 10^{-13}$	$6.73 \cdot 10^{-13}$
10000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$9.91 \cdot 10^{-13}$	$7.61 \cdot 10^{-14}$	$8.85 \cdot 10^{-14}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.06 \cdot 10^{-13}$	$8.49 \cdot 10^{-14}$	$6.37 \cdot 10^{-14}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$5.66 \cdot 10^{-14}$	$9.02 \cdot 10^{-14}$	$9.55 \cdot 10^{-14}$

the relative errors for the computed approximations (1.8) and (2.12) of (1.7) for $f(t) = 1/t$ and several values of N. For comparison, the table also displays the relative error in the approximations $\mathcal{G}_{n+1}(f)$, which are obtained by replacing the index n in (1.8) by $n+1$. While the evaluation of (1.8) and (2.12) requires the computation of n steps of the Lanczos process, the determination of $\mathcal{G}_{n+1}(f)$ needs $n + 1$ Lanczos steps be carried out. Table 3.1 shows the quadrature rules (2.12) to achieve higher accuracy than the n-point Gauss rules (1.8) for all N-values. Indeed, the errors obtained with the rules (2.12) are seen also to be smaller than the errors in the $(n + 1)$-point Gauss rules $\mathcal{G}_{n+1}(f)$. This situation takes place for some matrices A and functions f, though we would expect the errors achieved with the rules (2.12) to be smaller than the errors in the n-point Gauss rules (1.8) and larger than the errors in the $(n + 1)$-point Gauss rules $\mathcal{G}_{n+1}(f)$. This situation is illustrated in Table 3.2, which differs from Table 3.1 only in that the function is $f(t) = \exp(t)$. For this function the approximations (2.12) of (1.7) are more accurate than those determined with the n-point Gauss rules (1.8)

ETNA
Kent State University and
Johann Radon Institute (RICAM)

TABLE 3.3
Example 3.1: Relative error of computed approximations of $v^T f(A)v$ for $A \in \mathbb{R}^{N\times N}$ a Toeplitz matrix, $f(t) = \ln(t)$, and $v = [1, 1, \ldots, 1]^T$.

N		$n = 5$	$n = 10$	$n = 15$
200	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$3.80 \cdot 10^{-7}$	$1.63 \cdot 10^{-10}$	$1.00 \cdot 10^{-12}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$3.81 \cdot 10^{-8}$	$1.97 \cdot 10^{-11}$	$1.22 \cdot 10^{-13}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$7.59 \cdot 10^{-8}$	$3.67 \cdot 10^{-11}$	$2.28 \cdot 10^{-13}$
2000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$3.82 \cdot 10^{-8}$	$1.65 \cdot 10^{-11}$	$1.13 \cdot 10^{-13}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$3.84 \cdot 10^{-9}$	$1.99 \cdot 10^{-12}$	$2.02 \cdot 10^{-14}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$7.64 \cdot 10^{-9}$	$3.70 \cdot 10^{-12}$	$4.45 \cdot 10^{-14}$
5000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.53 \cdot 10^{-8}$	$6.59 \cdot 10^{-12}$	$6.47 \cdot 10^{-14}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$1.53 \cdot 10^{-9}$	$7.98 \cdot 10^{-13}$	$3.64 \cdot 10^{-14}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$3.06 \cdot 10^{-9}$	$1.48 \cdot 10^{-12}$	$4.24 \cdot 10^{-14}$
10000	$\lvert\mathcal{G}_n(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$7.64 \cdot 10^{-9}$	$3.30 \cdot 10^{-12}$	$5.05 \cdot 10^{-14}$
	$\lvert\widehat{\mathcal{G}}_{n+1}(f) - \mathcal{I}(f)\rvert/\lvert\mathcal{I}(f)\rvert$	$7.68 \cdot 10^{-10}$	$4.01 \cdot 10^{-13}$	$2.43 \cdot 10^{-14}$
	$\lvert\mathcal{G}_{n+1}(f) - \mathcal{I}(f)\rvert//\lvert\mathcal{I}(f)\rvert$	$1.53 \cdot 10^{-9}$	$7.44 \cdot 10^{-13}$	$4.45 \cdot 10^{-14}$

TABLE 3.4
Example 3.2: Relative error of computed approximations of $f(A)v$ for $A \in \mathbb{R}^{N\times N}$ a Toeplitz matrix, $f(t) = 1/t$, and $v = [1, 1, \ldots, 1]^T$.

N		$n = 5$	$n = 10$
200	$\lVert\mathcal{P}_n(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$6.80 \cdot 10^{-3}$	$2.14 \cdot 10^{-4}$
	$\lVert\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$3.20 \cdot 10^{-3}$	$9.93 \cdot 10^{-4}$
	$\lVert\mathcal{P}_{n+1}(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$3.40 \cdot 10^{-3}$	$1.07 \cdot 10^{-4}$
2000	$\lVert\mathcal{P}_n(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$2.20 \cdot 10^{-3}$	$6.89 \cdot 10^{-5}$
	$\lVert\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$1.00 \cdot 10^{-3}$	$3.20 \cdot 10^{-5}$
	$\lVert\mathcal{P}_{n+1}(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$1.10 \cdot 10^{-3}$	$3.40 \cdot 10^{-5}$
5000	$\lVert\mathcal{P}_n(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$1.40 \cdot 10^{-3}$	$4.36 \cdot 10^{-5}$
	$\lVert\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$6.40 \cdot 10^{-4}$	$2.02 \cdot 10^{-5}$
	$\lVert\mathcal{P}_{n+1}(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$6.98 \cdot 10^{-4}$	$2.10 \cdot 10^{-5}$
10000	$\lVert\mathcal{P}_n(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$9.85 \cdot 10^{-4}$	$3.09 \cdot 10^{-5}$
	$\lVert\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$4.59 \cdot 10^{-4}$	$1.44 \cdot 10^{-5}$
	$\lVert\mathcal{P}_{n+1}(f)v - f(A)v\rVert/\lVert f(A)v\rVert$	$4.93 \cdot 10^{-4}$	$1.54 \cdot 10^{-5}$

and less accurate than approximations achieved with the $(n+1)$-point Gauss rule $\mathcal{G}_{n+1}(f)$. Table 3.3 displays results for the function $f(t) = \ln(t)$. The matrix A, the vector v, and the orders N are the same as for Table 3.2.

EXAMPLE 3.2. This example illustrates the accuracy of the expressions

$$\mathcal{P}_n(f)v := V_n f(T_n)e_1\|v\|, \tag{3.1}$$

$$\widehat{\mathcal{P}}_{n+1}(f)v := V_{n+1} f(\widehat{T}_{n+1})e_1\|v\|, \tag{3.2}$$

when applied to approximate $f(A)v$. The matrices $A \in \mathbb{R}^{N\times N}$ and vectors v used in the present example are the same as in Example 3.1. Table 3.4 displays the relative errors in the quantities (3.1) and (3.2) for $f(t) = 1/t$. In addition, the table presents the relative errors in $\mathcal{P}_{n+1}(f)v$. While the computation of (3.1) and (3.2) requires that n steps of the Lanczos

ETNA
Kent State University and
Johann Radon Institute (RICAM)

TABLE 3.5
Example 3.2: Relative error of computed approximations of $f(A)v$ for $A \in \mathbb{R}^{N\times N}$ a Toeplitz matrix, $f(t) = \exp(t)$, and $v = [1, 1, \ldots, 1]^T$.

N		$n = 5$	$n = 10$
200	$\|\mathcal{P}_n(f)v - f(A)v\|/\|f(A)v\|$	$6.72 \cdot 10^{-5}$	$2.54 \cdot 10^{-10}$
	$\|\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$7.51 \cdot 10^{-6}$	$1.58 \cdot 10^{-11}$
	$\|\mathcal{P}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$7.15 \cdot 10^{-6}$	$1.52 \cdot 10^{-11}$
2000	$\|\mathcal{P}_n(f)v - f(A)v\|/\|f(A)v\|$	$2.14 \cdot 10^{-5}$	$8.13 \cdot 10^{-11}$
	$\|\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$2.39 \cdot 10^{-6}$	$5.07 \cdot 10^{-12}$
	$\|\mathcal{P}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$2.28 \cdot 10^{-6}$	$4.86 \cdot 10^{-12}$
5000	$\|\mathcal{P}_n(f)v - f(A)v\|/\|f(A)v\|$	$1.36 \cdot 10^{-5}$	$5.14 \cdot 10^{-11}$
	$\|\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$1.51 \cdot 10^{-6}$	$3.20 \cdot 10^{-12}$
	$\|\mathcal{P}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$1.44 \cdot 10^{-6}$	$3.07 \cdot 10^{-12}$
10000	$\|\mathcal{P}_n(f)v - f(A)v\|/\|f(A)v\|$	$9.58 \cdot 10^{-6}$	$3.64 \cdot 10^{-11}$
	$\|\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$1.07 \cdot 10^{-6}$	$2.27 \cdot 10^{-12}$
	$\|\mathcal{P}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$1.02 \cdot 10^{-6}$	$2.17 \cdot 10^{-12}$

TABLE 3.6
Example 3.2: Relative error of computed approximations of $f(A)v$ for $A \in \mathbb{R}^{N\times N}$ a Toeplitz matrix, $f(t) = \ln(t)$, and $v = [1, 1, \ldots, 1]^T$.

N		$n = 5$	$n = 10$
200	$\|\mathcal{P}_n(f)v - f(A)v\|/\|f(A)v\|$	$4.83 \cdot 10^{-4}$	$7.10 \cdot 10^{-6}$
	$\|\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$1.85 \cdot 10^{-4}$	$3.00 \cdot 10^{-6}$
	$\|\mathcal{P}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$1.97 \cdot 10^{-4}$	$3.21 \cdot 10^{-6}$
2000	$\|\mathcal{P}_n(f)v - f(A)v\|/\|f(A)v\|$	$1.53 \cdot 10^{-4}$	$2.25 \cdot 10^{-6}$
	$\|\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$5.87 \cdot 10^{-5}$	$9.50 \cdot 10^{-7}$
	$\|\mathcal{P}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$6.25 \cdot 10^{-5}$	$1.02 \cdot 10^{-6}$
5000	$\|\mathcal{P}_n(f)v - f(A)v\|/\|f(A)v\|$	$9.67 \cdot 10^{-5}$	$1.42 \cdot 10^{-6}$
	$\|\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$3.71 \cdot 10^{-5}$	$6.01 \cdot 10^{-7}$
	$\|\mathcal{P}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$3.95 \cdot 10^{-5}$	$6.43 \cdot 10^{-7}$
10000	$\|\mathcal{P}_n(f)v - f(A)v\|/\|f(A)v\|$	$6.84 \cdot 10^{-5}$	$1.01 \cdot 10^{-6}$
	$\|\widehat{\mathcal{P}}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$2.63 \cdot 10^{-5}$	$4.25 \cdot 10^{-7}$
	$\|\mathcal{P}_{n+1}(f)v - f(A)v\|/\|f(A)v\|$	$2.80 \cdot 10^{-5}$	$4.55 \cdot 10^{-7}$

algorithm be carried out, the evaluation of $\mathcal{P}_{n+1}(f)v$ demands $n + 1$ steps. The relative error in $\widehat{\mathcal{P}}_{n+1}(f)v$ is seen to be smaller than the relative error in $\mathcal{P}_n(f)v$ for all values of n and N.

Table 3.5 differs from Table 3.4 only in that the function is $f(t) = \exp(t)$. The relative performance of the approximants (3.1) and (3.2) is similar to that of Table 3.4. Finally, Table 3.6 differs from Table 3.5 only in that the function is $f(t) = \ln(t)$; the matrices A, the vector v, and the sizes N, are the same in all tables.

The performance of the approximants (3.1) and (3.2) for other matrices A, vectors v, functions f, and number of Lanczos steps n is similar to that of the Tables 3.1–3.6. We therefore do not show these results.

4. Conclusion and extension. Many methods for the approximation of functions of a large symmetric matrix are based on the Lanczos process. The application of n steps of the Lanczos process to A with the initial vector v yields the decomposition (1.4), which is

ETNA
Kent State University and
Johann Radon Institute (RICAM)

commonly used to compute approximations (3.1) and (1.8) of the expressions (1.1). This paper shows that the expressions (3.2) and (2.12), which can be computed with essentially the same amount of arithmetic work, may yield more accurate approximations of (1.1) than (3.1) and (1.8).

The technique of this paper also can be applied when v in (1.1) is a "block vector," i.e., a matrix with a few columns. The expressions (1.7) then are matrix-valued. In particular, the measure $d\omega(t)$ is matrix-valued, the quadrature rule (1.8) becomes a block Gauss rule, and the Lanczos process is replaced by a block Lanczos process; see, e.g., Golub and Meurant [8] for details on block Gauss rules.

REFERENCES

[1] 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.
[2] M. BENZI AND P. BOITO, *Quadrature rule-based bounds for functions of adjacency matrices*, Linear Algebra Appl., 433 (2010), pp. 637–652.
[3] D. CALVETTI AND L. REICHEL, *Lanczos-based exponential filtering for discrete ill-posed problems*, Numer. Algorithms, 29 (2002), pp. 45–65.
[4] D. Lj. Djukić, L. Reichel, and M. M. Spalević, *Truncated generalized averaged Gauss quadrature rules*, J. Comput. Appl. Math., 308 (2016), pp. 408–418.
[5] 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.
[6] A. FROMMER AND M. SCHWEITZER, *Error bounds and estimates for Krylov subspace approximations of Stieltjes matrix functions*, BIT, 56 (2016), pp. 865–892.
[7] W. GAUTSCHI, *On generating orthogonal polynomials*, SIAM J. Sci. Statist. Comput., 3 (1982), pp. 289–317.
[8] G. H. GOLUB AND G. MEURANT, *Matrices, Moments and Quadrature with Applications*, Princeton University Press, Princeton, 2010.
[9] N. J. HIGHAM, *Functions of Matrices*, SIAM, Philadelphia, 2008.
[10] M. HOCHBRUCK AND C. LUBICH, *On Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 34 (1997), pp. 1911–1925.
[11] S. E. NOTARIS, *Gauss–Kronrod quadrature formulae – a survey of fifty years of research*, Electron. Trans. Numer. Anal., 45 (2016), pp. 371–404.
http://etna.ricam.oeaw.ac.at/vol.45.2016/pp371-404.dir/pp371-404.pdf
[12] L. REICHEL, M. M. SPALEVIĆ, AND T. TANG, *Generalized averaged Gauss quadrature rules for the approximation of matrix functionals*, BIT, 56 (2016), pp. 1045–1067.
[13] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.
[14] M. M. SPALEVIĆ, *A note on generalized averaged Gaussian formulas*, Numer. Algorithms, 46 (2007), pp. 253–264.