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

Generalized averaged Gauss quadrature rules for the approximation of matrix functionals

Lothar Reichel · Miodrag M. Spalević · Tunan Tang

Received: date / Accepted: date

Abstract The need to compute expressions of the form $u^*f(A)v$, where A is a large square matrix, u and v are vectors, and f is a function, arises in many applications, including network analysis, quantum chromodynamics, and the solution of linear discrete ill-posed problems. Commonly used approaches first reduce A to a small matrix by a few steps of the Hermitian or non-Hermitian Lanczos processes and then evaluate the reduced problem. This paper describes a new method to determine error estimates for computed quantities and shows how to achieve higher accuracy than available methods for essentially the same computational effort. Our methods are based on recently proposed generalized averaged Gauss quadrature formulas.

Keywords matrix functional · Gauss quadrature · averaged Gauss rules

Mathematics Subject Classification (2010) 65D15, 65D32, 65F60

1 Introduction

Matrix functionals of the form

$$\mathcal{F}(A) := u^*f(A)v, \tag{1.1}$$

where $A \in \mathbb{C}^{m\times m}$ is a large Hermitian or non-Hermitian matrix, $u, v \in \mathbb{C}^m$ are vectors, and the superscript * denotes transposition and complex conjugation, arise in many applications, such as network analysis [4,5,15,16], quantum chromodynamics and statistics [2], as well as in the solution of linear discrete ill-posed problems by Tikhonov regularization [25]; see also [20] for

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

Miodrag M. Spalević
Department of Mathematics, Faculty of Mechanical Engineering
University of Belgrade, Kraljice Marije 16, 11120 Belgrade 35, Serbia
E-mail: mspalevic@mas.bg.ac.rs

Tunan Tang
Department of Mathematical Sciences
Kent State University, Kent, OH 44242, USA
E-mail: nickdon2006@gmail.com

discussions on applications. We are interested in developing efficient methods for the accurate approximation of this type of functionals.

We assume in this section, unless explicitly stated otherwise, that A is Hermitian, and we let $v = u$. Application of ℓ steps of the Hermitian Lanczos process to A with initial vector u yields the Lanczos decomposition

$$AU_\ell = U_\ell T_\ell + \beta_\ell u_{\ell+1} e_\ell^*, \tag{1.2}$$

where $U_\ell \in \mathbb{C}^{m\times\ell}$ has orthonormal columns with the initial column proportional to the vector u, $T_\ell \in \mathbb{R}^{\ell\times\ell}$ is a symmetric tridiagonal matrix, β_ℓ is a nonnegative scalar, the unit vector $u_{\ell+1} \in \mathbb{C}^m$ is orthogonal to the columns of U_ℓ, and e_ℓ denotes the ℓth column of an identity matrix I of suitable order. The number of Lanczos steps, ℓ, is assumed to be small enough so that the decomposition (1.2) with the stated properties exists. This is the generic situation. When A is large, the dominating computational effort required to construct the decomposition (1.2) is the evaluation of ℓ matrix-vector products with the matrix A; see Section 2 for details. For instance, in several applications to network analysis reported in [15,16], m is $5 \cdot 10^4$ or larger and ℓ is less than 10. Illustrations when $m \gg \ell$ also can be found at the end of Section 4.

The functional (1.1) with $v = u$ is a Stieltjes integral. This can be seen as follows. Consider the spectral factorization

$$A = W\Lambda W^*, \quad \Lambda = \mathrm{diag}[\lambda_1, \ldots, \lambda_m] \in \mathbb{R}^{m\times m}, \quad W \in \mathbb{C}^{m\times m}, \quad W^*W = I.$$

Substituting this factorization into (1.1) yields

$$\mathcal{F}(A) = \sum_{j=1}^{m} f(\lambda_j)\mu_j, \qquad \mu_j := |e_j^* W^* u|^2. \tag{1.3}$$

Let $\mu_{A,u}$ be a nondecreasing piecewise constant function on the real axis with jump μ_j at the eigenvalue λ_j, for $1 \le j \le m$, and let $\mathrm{d}\mu_{A,u}$ denote the associated measure. Then the sum (1.3) can be written as the Stieltjes integral

$$\mathcal{I}f = \int f(x)\mathrm{d}\mu_{A,u}(x). \tag{1.4}$$

Thus, the expression (1.1) with A Hermitian and $v = u$ is equivalent to (1.4). We will discuss the approximation of (1.4) by several quadrature rules, including Gauss quadrature. We say that an ℓ-point quadrature rule $\mathcal{G}_\ell$ is a Gaussian rule for approximating (1.4) if

$$\mathcal{G}_\ell f = \mathcal{I}f \qquad \forall f \in \Pi_{2\ell-1},$$

where $\Pi_{2\ell-1}$ denotes the set of all polynomials of degree at most $2\ell - 1$.

Golub and Meurant show in [19] and [20, Chapter 7] that the ℓ-point Gauss quadrature rule for approximating (1.4) can be written as

$$\mathcal{G}_\ell f = \|u\|^2 e_1^* f(T_\ell) e_1, \tag{1.5}$$

where T_ℓ is the symmetric tridiagonal matrix defined by (1.2). Here and throughout this paper, $\|\cdot\|$ denotes the Euclidean vector norm. The fact that the right-hand side indeed is an ℓ-point quadrature rule can be seen by substituting the spectral factorization of T_ℓ into (1.5). This shows that the eigenvalues of T_ℓ are the nodes and the squared magnitudes of the first components of suitably normalized eigenvectors are the weights of the Gauss rule. Our reason for using Gauss quadrature rules for approximating (1.4) is that these rules can be determined by applying a few steps of the Lanczos process to A with initial vector u without explicit knowledge of the measure $\mathrm{d}\mu_{A,u}$.

When m is large and ℓ is fairly small, the evaluation of (1.5) is much cheaper than the computation of the expression (1.1). It is therefore attractive to evaluate (1.5) instead of (1.1). We are interested in estimating the incurred error,

$$\mathcal{E}_\ell f := \mathcal{I} f - \mathcal{G}_\ell f. \tag{1.6}$$

In the special situation when the matrix A is Hermitian with a known upper bound for its largest eigenvalue, a known lower bound for its smallest eigenvalue, and the derivatives of f of orders 2ℓ and $2\ell+1$ are of constant sign on the convex hull of the spectrum of A, an $(\ell+1)$-point Gauss–Radau quadrature rule $\mathcal{R}_{\ell+1}$ with a suitably allocated fixed node ζ can be constructed, so that the values $\mathcal{G}_\ell f$ and $\mathcal{R}_{\ell+1} f$ bracket $\mathcal{I} f$.[1] Details about this approach to approximate $\mathcal{I} f$ are described by Golub and Meurant; see [19] and [20, Chapter 11]. This way of approximating $\mathcal{I} f$ is very useful when it can be applied. We are interested in situations when the derivatives of f of orders 2ℓ or $2\ell+1$ change sign on the convex hull of the spectrum of A, or when A is non-Hermitian. Then pairs of Gauss and Gauss–Radau rules $\mathcal{G}_\ell f$ and $\mathcal{R}_{\ell+1} f$ are not guaranteed to bracket $\mathcal{I} f$. We propose to use averaged Gauss quadrature rules to estimate the quadrature error in this situation.

The accurate estimation of the error (1.6) for general integrands f and measures $\mathrm{d}\mu_{A,u}$ on the real axis is a difficult problem that has received considerable attention in the literature. For instance, it has lead to the development of Gauss–Kronrod quadrature rules; see [1,8,24,26] and references therein. The difficulty of estimating the error (1.6) stems from the fact that the evaluation of bounds for the quadrature error requires knowledge of quantities that are typically difficult or impossible to determine. We remark that the difference $\mathcal{G}_{\ell+1} f - \mathcal{G}_\ell f$ is known not to provide a reliable estimate of the error (1.6); see [10] for a discussion.

Recently, Spalević [31,32] proposed a new quadrature rule, referred to as a generalized averaged Gauss quadrature rule and denoted by $\widehat{\mathcal{G}}_{2\ell+1}$ below, for estimating the error (1.6). This rule has $2\ell+1$ nodes and the nodes of $\mathcal{G}_\ell$ form a subset. This is similar to the situation for the $(2\ell+1)$-point Gauss–Kronrod rule $\mathcal{H}_{2\ell+1}$ associated with $\mathcal{G}_\ell$. An attractive feature of the generalized averaged Gauss rule $\widehat{\mathcal{G}}_{2\ell+1}$ is that it exists also when the Gauss–Kronrod rule $\mathcal{H}_{2\ell+1}$ does not. The rule $\widehat{\mathcal{G}}_{2\ell+1}$ is the optimal stratified extension of the Gauss rule $\mathcal{G}_\ell$; see [31]. This rule is of particular interest since it covers nested and stratified formulas; see Peherstorfer [28, p. 2245] for a discussion.

Our interest in the rule $\widehat{\mathcal{G}}_{2\ell+1}$ stems from the fact that it easily can be determined for measures $\mathrm{d}\mu_{A,u}$. The execution of $\ell+1$ steps of the Hermitian Lanczos process applied to A with initial vector u suffices. We will show that the rule $\widehat{\mathcal{G}}_{2\ell+1}$ is exact for all polynomials in $\Pi_{2\ell+2}$. We remark that the standard $(\ell+1)$-point Gauss rule, $\mathcal{G}_{\ell+1}$, also can be determined with $\ell+1$ steps of the Hermitian Lanczos process. It is exact for all polynomials in $\Pi_{2\ell+1}$. The rule $\widehat{\mathcal{G}}_{2\ell+1}$ therefore can be expected to give a more accurate approximation of $\mathcal{I} f$ than the Gauss rule $\mathcal{G}_{\ell+1}$ for essentially the same computational effort. The fact that $\widehat{\mathcal{G}}_{2\ell+1}$ may give a smaller quadrature error than $\mathcal{G}_{\ell+1}$ can be seen as follows. Let $p_0, p_1, p_2, \ldots$ be a sequence of orthonormal polynomials with respect to the inner product

$$\langle p, q \rangle = \mathcal{I}(pq),$$

defined for polynomials of not too high degree, and assume that the integrand is smooth enough so that the coefficients γ_j in the expansion

$$f(x) = \gamma_0 p_0(x) + \gamma_1 p_1(x) + \gamma_2 p_2(x) + \ldots$$

[1] The requirements on the bounds of the spectrum of A can be weakened depending on the location of the node ζ.

converge to zero quickly. Then the quadrature error is dominated by the integral of the first term in the above expansion that is not integrated exactly, and the error $|(\mathcal{I}-\widehat{\mathcal{G}}_{2\ell+1})f|$ is smaller than the error $|(\mathcal{I}-\mathcal{G}_{\ell+1})f|$. Similarly, the error $|(\mathcal{I}-\mathcal{G}_{\ell+1})f|$ is smaller than the error $|(\mathcal{I}-\mathcal{G}_{\ell})f|$.

Since the quadrature rule $\mathcal{G}_{\ell+1}f$ typically gives a more accurate approximation of $\mathcal{I}f$ than $\mathcal{G}_{\ell}f$, and the evaluation of the rule $\widehat{\mathcal{G}}_{2\ell+1}f$ requires the execution of $\ell+1$ steps of the Lanczos process, similarly as the computation of $\mathcal{G}_{\ell+1}f$, we will use the value $\widehat{\mathcal{G}}_{2\ell+1}f$ to estimate the error in $\mathcal{G}_{\ell+1}f$. We remark that when the matrix A is large and ℓ is small, the additional computational work required to evaluate $\widehat{\mathcal{G}}_{2\ell+1}f$ when $\mathcal{G}_{\ell+1}f$ already has been computed is relatively small.

We also consider "truncations" of the generalized averaged Gauss quadrature rule $\widehat{\mathcal{G}}_{2\ell+1}$. They also are exact for all polynomials in $\Pi_{2\ell+2}$. Our interest in these truncations stems from the fact that they may have all nodes in the convex hull of the support of the measure $\mathrm{d}\mu_{A,u}$ when $\widehat{\mathcal{G}}_{2\ell+1}$ does not. This property is important when the integrand f only is defined on the convex hull of the spectrum of A. Example 4.1 in Section 4 provides an illustration.

It is the purpose of the present paper to discuss the application of generalized averaged Gaussian quadrature rules to the approximation of matrix functionals (1.1). Section 2 reviews results by Spalević [31,32] and introduces truncated generalized averaged Gauss rules. This section is concerned with the situation when A in (1.1) is Hermitian. Section 3 discusses quadrature rules that can be applied when A is non-Hermitian. Then A can be reduced to a small non-Hermitian tridiagonal matrix by the application of a few steps of the non-Hermitian Lanczos process. The non-Hermitian tridiagonal matrix so obtained defines a Gauss-type quadrature rule, and we introduce associated generalized averaged Gauss rules. These rules are new. Computed examples are presented in Section 4, and concluding remarks can be found in Section 5.

2 Quadrature rules for Hermitian matrices

We first review results shown in [31,32]. Let $\mathrm{d}\mu$ be a nonnegative measure defined on the real axis such that all moments

$$m_j := \int x^j \mathrm{d}\mu(x), \qquad j = 0, 1, 2, \dots \,, \tag{2.1}$$

exist and are bounded. For notational simplicity, we require the measure to have infinitely many points of support and to be of total mass one. Application to the approximation of matrix functionals (1.1) is discussed at the end of the section.

Introduce the inner product

$$\langle f, g\rangle := \int f(x)g(x)\mathrm{d}\mu(x)$$

and let $\{p_j\}_{j=0}^{\infty}$ denote a family of orthonormal polynomials associated with this inner product. Thus, p_j is of degree j with positive leading coefficient and such that

$$\langle p_j, p_k\rangle = \begin{cases} 1 & j = k, \\ 0 & j \neq k. \end{cases}$$

The p_j satisfy a recursion relation of the form

$$\beta_{j+1}p_{j+1}(x) = (x-\alpha_j)p_j(x) - \beta_j p_{j-1}(x), \qquad j = 0, 1, 2, \dots \,, \tag{2.2}$$

where $p_{-1}(x) := 0$, $p_1(x) := 1$, and $\beta_0 := 0$. The recursion coefficients α_j are real and the coefficients β_j, $j \geq 1$, can be chosen real and nonnegative.

Define the symmetric tridiagonal matrix

$$T_\ell = \begin{bmatrix} \alpha_0 & \beta_1 & & & 0 \\ \beta_1 & \alpha_1 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{\ell-2} & \alpha_{\ell-2} & \beta_{\ell-1} \\ 0 & & & \beta_{\ell-1} & \alpha_{\ell-1} \end{bmatrix} \in \mathbb{R}^{\ell\times\ell} \tag{2.3}$$

determined by the first $2\ell-1$ nontrivial recursion coefficients. It is well known that the eigenvalues and the squares of the moduli of the first components of normalized eigenvectors give the ℓ-point Gauss quadrature rule for the approximation of

$$\mathcal{I}f := \int f(x)\mathrm{d}\mu(x);$$

see, e.g., [21]. As mentioned in Section 1, this rule can be expressed as

$$\mathcal{G}_\ell f = e_1^* f(T_\ell) e_1; \tag{2.4}$$

see Gautschi [18] and Golub and Meurant [20] for detailed discussions on orthogonal polynomials and Gauss quadrature.

Introduce for $0 \le r < \ell$ the reverse matrices

$$\widetilde{T}_{\ell-r,r} = \begin{bmatrix} \alpha_{\ell-1} & \beta_{\ell-1} & & & 0 \\ \beta_{\ell-1} & \alpha_{\ell-2} & \beta_{\ell-2} & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{r+2} & \alpha_{r+1} & \beta_{r+1} \\ 0 & & & \beta_{r+1} & \alpha_r \end{bmatrix} \in \mathbb{R}^{(\ell-r)\times(\ell-r)} \tag{2.5}$$

as well as the concatenated symmetric tridiagonal matrices

$$\widehat{T}_{2\ell+1-r,r} = \begin{bmatrix} T_\ell & \beta_\ell e_\ell & 0 \\ \beta_\ell e_\ell^* & \alpha_\ell & \beta_{\ell+1} e_1^* \\ 0 & \beta_{\ell+1} e_1 & \widetilde{T}_{\ell-r,r} \end{bmatrix} \in \mathbb{R}^{(2\ell+1-r)\times(2\ell+1-r)}. \tag{2.6}$$

Spalević [31,32] considered the latter matrix for $r = 0$ and referred to the associated quadrature rule

$$\widehat{\mathcal{G}}_{2\ell+1} f = e_1^* f(\widehat{T}_{2\ell+1,0}) e_1 \tag{2.7}$$

as a generalized averaged Gauss formula. Substituting the spectral factorization of $\widehat{T}_{2\ell+1,0}$ into (2.7) shows that this quadrature rule has $2\ell+1$ nodes. Special cases of this quadrature rule for Laguerre and Hermite measures have previously been considered by Ehrich [11], but without defining the associated matrix (2.6) for $r = 0$. We provide some properties of the matrix $\widehat{T}_{2\ell+1,0}$ and the corresponding quadrature rule.

Proposition 2.1 *The quadrature rule (2.7) is exact for all $f \in \Pi_{2\ell+2}$. If the measure $\mathrm{d}\mu$ is symmetric with respect to the origin, then the quadrature rule (2.7) is exact for all $f \in \Pi_{2\ell+3}$. The nodes of the Gauss rule (2.4) form a subset of the nodes of the rule (2.7).*

Proof First assume that $\mathrm{d}\mu$ is a general, not necessarily symmetric, measure. Consider the Chebyshev algorithm described in, e.g., [18, Section 2.1.7]. It determines recursion coefficients α_j and β_j of orthogonal polynomials from the moments (2.1). More precisely, the entries $\alpha_0, \alpha_1, \ldots, \alpha_\ell$ and $\beta_1, \beta_2, \ldots, \beta_{\ell+1}$ of (2.6) are determined by the moments $m_0, m_1, \ldots, m_{2\ell+2}$ in the order $\alpha_0, \beta_1, \alpha_1, \beta_2, \ldots, \beta_\ell, \alpha_\ell, \beta_{\ell+1}$; see, e.g., Peherstorfer [28, Theorem 2.1] for details. We conclude that the quadrature rule (2.7) is exact for (at least) all polynomials in $\Pi_{2\ell+2}$. When the measure $\mathrm{d}\mu$ is symmetric with respect to the origin, all diagonal entries of the matrix (2.6) vanish. Therefore, the $(\ell+2)$nd diagonal entry of (2.6) can be thought of having been determined by the moments $m_0, m_1, \ldots, m_{2\ell+3}$. It follows that the quadrature rule (2.7) is exact for all $f \in \Pi_{2\ell+3}$.

The leading and trailing principal $\ell \times \ell$ submatrices of $\widehat{T}_{2\ell+1,0}$ have the same eigenvalues. By considering the characteristic polynomial for $\widehat{T}_{2\ell+1,0}$ and expanding the determinant along row $\ell+1$, one can show that the spectrum of the leading principal submatrix T_ℓ of $\widehat{T}_{2\ell+1,0}$ is a subset of the spectrum of the latter matrix; see [24, Lemma 1] for details. □

The property that the nodes of the Gauss rule (2.4) form a subset of the nodes of the rule (2.7) can be utilized in computations. Calvetti et al. [8] show how a divide-and-conquer method described in [6] can be used efficiently in the context of the computation of the nodes and weights of Gauss–Kronrod quadrature rules. The same approach can be applied to evaluate the nodes and weights of the rule (2.7). We note, however, that (2.7) can be evaluated for many functions f without computing the spectral factorization of $\widehat{T}_{2\ell+1,0}$; see Higham [23] for algorithms and analyses of many methods for evaluating functions of matrices of small to moderate size. We remark that in typical applications the matrix A is large, while the matrices $T_{\ell+1}$ and $\widehat{T}_{2\ell+1,0}$ are small. The additional computational effort required for evaluating $f(\widehat{T}_{2\ell+1,0})$ instead of $f(T_{\ell+1})$ then is negligible in comparison with the effort required to evaluate $\ell+1$ matrix-vector products with the matrix A.

The quadrature rule $\widehat{\mathcal{G}}_{2\ell+1}f$ has at most two nodes outside the convex hull of the support of the measure $\mathrm{d}\mu_{A,u}$, see [31] for a discussion, where also a detailed analysis of the location of the extreme nodes as a function of $\alpha, \beta > 0$ is provided for the Jacobi measure $\mathrm{d}\mu^{(\alpha,\beta)}(x) = (1-x)^\alpha(1+x)^\beta \mathrm{d}x$, $-1 < x < 1$. Conditions for when all quadrature nodes are in the interval $[-1, 1]$ are given. The location of extreme nodes for the Laguerre and Hermite weight functions has been investigated by Ehrich [11].

We are also interested in the quadrature rules

$$\widehat{\mathcal{G}}_{2\ell+1-r,r}f = e_1^* f(\widehat{T}_{2\ell+1-r,r})e_1 \tag{2.8}$$

for $1 \le r < \ell$. We note that the nodes of the Gauss rule $\mathcal{G}_\ell f$ do not form a subset of the nodes of the rules $\widehat{\mathcal{G}}_{2\ell+1-r,r}f$ for $1 \le r < \ell$. Numerical experiments show that the rule $\widehat{\mathcal{G}}_{2\ell+1-r,r}f$ for some $1 \le r < \ell$ may have all nodes in the convex hull of the support of the measure $\mathrm{d}\mu_{A,u}$ when the rule (2.7) does not; see Example 4.1 below. Moreover, the rules (2.8) integrate the same class of functions exactly as the rule (2.7). The latter property can be shown in the same manner as Proposition 2.1. We therefore omit the proof.

Proposition 2.2 *The quadrature rules (2.8), for $1 \le r < \ell$, are exact for all $f \in \Pi_{2\ell+2}$. If the measure $\mathrm{d}\mu$ is symmetric with respect to the origin, then they are exact for all $f \in \Pi_{2\ell+3}$.*

The actual accuracy achieved with the quadrature rules (2.7) and (2.8) may be higher than suggested by Propositions 2.1 and 2.2. Let the support of measure $\mathrm{d}\mu$ be in a bounded real interval. Then the recursion coefficients for orthonormal polynomials determined by this measure have limits, i.e., $\alpha_j \to \alpha$ and $\beta_j \to \beta$ for some constants α and β as j increases. This means that if ℓ is sufficiently large, then for a given constant $\varepsilon > 0$, there is a constant $k < \ell$ such

that $|\alpha_j - \alpha| \leq \varepsilon$ and $|\beta_j - \beta| \leq \varepsilon$ for all $j \geq k$. We then can conclude similarly as in [32] that the quadrature rule $\widehat{\mathcal{G}}_{2\ell+1-k,k}f$ is nearly exact for all $f \in \Pi_{4\ell-2k+1}$. Let $k < r < \ell$. Then the quadrature rule $\widehat{\mathcal{G}}_{2\ell+1-r,r}f$ integrates all $f \in \Pi_{4\ell-2r+1}$ nearly exactly. Since $4\ell - 2r + 1 < 4\ell - 2k + 1$, we expect the rule $\widehat{\mathcal{G}}_{2\ell+1-r,r}f$ to yield a larger quadrature error than $\widehat{\mathcal{G}}_{2\ell+1-k,k}f$. This is illustrated in Section 4. We conclude that the main advantage of the rules (2.8), when compared to the rule (2.7), is that the former rule may have all nodes in the convex hull of the support of the measure when the latter does not.This property follows from the fact that the eigenvalues of the matrix $\widehat{T}_{2\ell-r,r+1}$ interlace the eigenvalues of $\widehat{T}_{2\ell+1-r,r}$ for $0 \leq r < \ell$.

We remark that one could use the Gauss rules $\mathcal{G}_{2\ell+1-r}f = e_1^* f(T_{2\ell+1-r})e_1$ instead of the rules (2.8) to estimate the error in $\mathcal{G}_{\ell+1}f$. Our interest in using the rules (2.8) stems from the fact that their computation only requires the evaluation of $\ell + 1$ steps with the Lanczos algorithm, while computing $\mathcal{G}_{2\ell+1-r}f$ would demand $2\ell + 1 - r$ steps and, therefore, be more expensive.

We turn to the problem of computing approximations of matrix functionals (1.1) with a large Hermitian matrix $A \in \mathbb{C}^{m\times m}$. Assume that $v = u$. Then the decomposition (1.2) can be computed by ℓ steps of the Hermitian Lanczos algorithm with initial vector u. The computation of the entries of the matrix (2.6) requires $\ell + 1$ steps. We therefore describe the algorithm for this number of steps.

Algorithm 2.1: The Hermitian Lanczos algorithm
Initialization: $u_1 := u/\|u\|$; $u_0 := 0$; $\beta_0 := 0$;

for $j = 1, 2, \ldots, \ell + 1$ do

$w := Au_j - \beta_{j-1}u_{j-1}$;
$\alpha_{j-1} := w^* u_j$;
$w := w - \alpha_{j-1}u_j$;
$\beta_j := \|w\|$;
$u_{j+1} := w/\beta_j$;

end j

Introduce the matrix $U_{\ell+1} = [u_1, \ldots, u_{\ell+1}] \in \mathbb{R}^{m\times(\ell+1)}$ with orthonormal columns determined by Algorithm 2.1, and let the scalars $\alpha_0, \alpha_1, \ldots, \alpha_\ell$ and $\beta_1, \beta_2, \ldots, \beta_\ell$, also computed by the algorithm, define the symmetric tridiagonal matrix

$$T_{\ell+1} = \begin{bmatrix} \alpha_0 & \beta_1 & & & 0 \\ \beta_1 & \alpha_1 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{\ell-1} & \alpha_{\ell-1} & \beta_\ell \\ 0 & & & \beta_\ell & \alpha_\ell \end{bmatrix} \in \mathbb{R}^{(\ell+1)\times(\ell+1)}. \tag{2.9}$$

The matrix (2.3) is the leading $\ell \times \ell$ principal submatrix of $T_{\ell+1}$. Analogously to (1.2), we have

$$AU_{\ell+1} = U_{\ell+1}T_{\ell+1} + \beta_{\ell+1}u_{\ell+2}e_{\ell+1}^*. \tag{2.10}$$

The entries of the matrices (2.6) and (2.9) can be determined from (2.10). We recall that the latter matrix determines an $(\ell + 1)$-point Gauss quadrature rule that is exact for all $f \in \Pi_{2\ell+1}$, while the quadrature rules associated with (2.6) for $0 \leq r < \ell$ are exact for at least all $f \in \Pi_{2\ell+2}$.

In each step of Algorithm 2.1, one matrix-vector product with the matrix A is evaluated. When A is large, this is the dominant computational work required by the algorithm. Therefore the determination of the rules (2.8) and of the Gauss rule $\mathcal{G}_{\ell+1}$ requires essentially the same computational work. Our interest in the former rules stems from the fact that they are exact for a larger class of polynomials than the latter. Moreover, they typically yield accurate estimates of the quadrature error (1.6) with ℓ replaced by $\ell+1$.

We remark that Saad [29] proposed an interesting approach, referred to as a "corrected scheme", to improve the accuracy when approximating the matrix exponential $\exp(A)u$ by a Krylov subspace method. The corrected scheme also can be applied to improve Krylov subspace approximations of expressions of the form $u^*\exp(A)u$. Similarly to our method, the corrected scheme uses the vector $u_{\ell+1}$ and coefficient β_ℓ in the Lanczos decomposition (1.2). One row and one column are appended to the symmetric tridiagonal matrix T_ℓ in (1.2) to obtain a non-Hermitian matrix. Hence, Saad's corrected scheme and our approach are different. Moreover, the corrected scheme is only derived for the exponential function and uses properties of this function.

In many applications, the number of steps, $\ell+1$, can be chosen fairly small. Then loss of orthogonality of the vectors $u_1, u_2, \ldots, u_{\ell+2}$ caused by propagated round-off errors introduced during the execution of Algorithm 2.1 generally is not a problem. When ℓ is not small or for matrices A with certain eigenvalue distributions, the vectors u_j may have to be reorthogonalized to secure accurate evaluation of the computed quadrature rule.

In the Lanczos decomposition (2.10), the remainder $AU_{\ell+1} - U_{\ell+1}T_{\ell+1}$ is of rank one. The following result describes a decomposition involving the tridiagonal matrix (2.6) in which the remainder is of rank less than or equal to two. The decomposition uses the permutation matrix

$$P_\ell = [e_\ell, e_{\ell-1}, \ldots, e_1] \in \mathbb{R}^{\ell\times\ell}. \tag{2.11}$$

Theorem 2.1 *Let $\widehat{U}_{2\ell+1} = [U_{\ell+1}, U_\ell P_\ell] \in \mathbb{C}^{m\times(2\ell+1)}$, where P_ℓ is given by (2.11). Then the expression*

$$A\widehat{U}_{2\ell+1} - \widehat{U}_{2\ell+1}\widehat{T}_{2\ell+1,0} \tag{2.12}$$

is of rank at most two.

Proof Introduce the matrix $\widetilde{U}_\ell = U_\ell P_\ell$ and note that $\widetilde{T}_{\ell,0} = P_\ell T_\ell P_\ell$. We obtain from (1.2) that

$$\begin{aligned} A\widetilde{U}_\ell &= AU_\ell P_\ell = (U_\ell T_\ell + \beta_\ell u_{\ell+1}e_\ell^*)P_\ell \\ &= U_\ell P_\ell P_\ell T_\ell P_\ell + \beta_\ell u_{\ell+1}e_\ell^* P_\ell = \widetilde{U}_\ell\widetilde{T}_{\ell,0} + \beta_\ell u_{\ell+1}e_1^*. \end{aligned} \tag{2.13}$$

It follows that

$$A[U_{\ell+1}, \widetilde{U}_\ell] = [U_{\ell+1}T_{\ell+1}, \widetilde{U}_\ell\widetilde{T}_{\ell,0}] + [\beta_{\ell+1}u_{\ell+2}e_{\ell+1}^*, \beta_\ell u_{\ell+1}e_1^*].$$

Moreover,

$$[U_{\ell+1}, \widetilde{U}_\ell]\widehat{T}_{2\ell+1,0} = [U_{\ell+1}T_{\ell+1} + \beta_{\ell+1}u_\ell e_{\ell+1}^*, \widetilde{U}_\ell\widetilde{T}_{\ell,0} + \beta_{\ell+1}u_{\ell+1}e_1^*].$$

Therefore,

$$A[U_{\ell+1}, \widetilde{U}_\ell] - [\beta_{\ell+1}u_{\ell+2}e_{\ell+1}^*, \beta_\ell u_{\ell+1}e_1^*] = [U_{\ell+1}, \widetilde{U}_\ell]\widehat{T}_{2\ell+1,0} - [\beta_{\ell+1}u_\ell e_{\ell+1}^*, \beta_{\ell+1}u_{\ell+1}e_1^*].$$

Rearranging the terms, we obtain

$$A\widehat{U}_{2\ell+1} = \widehat{U}_{2\ell+1}\widehat{T}_{2\ell+1,0} + [\beta_{\ell+1}(u_{\ell+2} - u_\ell)e_{\ell+1}^*, (\beta_\ell - \beta_{\ell+1})u_{\ell+1}e_1^*],$$

which shows the theorem. If $\beta_\ell = \beta_{\ell+1}$, then the expression (2.12) is of rank one. □

The theorem illustrates both the similarity and difference of the expression (2.12) and the expression $AU_{2\ell+1} - U_{2\ell+1}T_{2\ell+1}$ that is obtained when carrying out $2\ell+1$ steps of the Lanczos process. Both expressions are of low rank, but the matrix $\widehat{U}_{2\ell+1}$ is of rank $\ell+1$ only.

We assumed above that $v = u$ in (1.1). This restriction can be removed by writing (1.1) in the form

$$u^*f(A)v = \frac{1}{4}(u+v)^*f(A)(u+v) - \frac{1}{4}(u-v)^*f(A)(u-v). \tag{2.14}$$

Each term in the right-hand side can be approximated by quadrature rules in the manner described above. Since there are two terms, the work is doubled compared with the situation when $u = v$. The following section discusses another approach to approximate the functional in the left-hand side of (2.14). This approach can also be applied when the matrix A is non-Hermitian.

3 Quadrature rules for non-Hermitian matrices

This section discusses quadrature rules that can be applied when $A \in \mathbb{C}^{m\times m}$ is a non-Hermitian diagonalizable matrix with spectral factorization $A = S\Lambda S^{-1}$, where $S \in \mathbb{C}^{m\times m}$ is nonsingular and $\Lambda = \mathrm{diag}[\lambda_1, \ldots, \lambda_m] \in \mathbb{C}^{m\times m}$. The vectors $u, v \in \mathbb{C}^m$ in (1.1) are assumed to be scaled so that $u^*v = 1$.

Substituting the spectral factorization into (1.1) yields

$$\mathcal{F}(A) = u^*f(A)v = u^*Sf(\Lambda)S^{-1}v = \sum_{j=1}^{m} f(\lambda_i)\mu_j\mu'_j, \tag{3.1}$$

where $[\mu_1, \ldots, \mu_m] := u^*S$ and $[\mu'_1, \ldots, \mu'_m]^T := S^{-1}v$. The right-hand side of (3.1) can be written as an integral

$$\mathcal{I}f = \int f(x)\mathrm{d}\mu_{A,u,v}(x), \tag{3.2}$$

with the measure

$$\mathrm{d}\mu_{A,u,v}(x) := \sum_{j=1}^{m} \delta(x-\lambda_j)\mu_j\mu'_j, \tag{3.3}$$

where $\delta(\cdot)$ denotes the Dirac δ-function.

Introduce the bilinear form

$$\langle f, g\rangle := \int f(x)g(x)\mathrm{d}\mu_{A,u,v}(x). \tag{3.4}$$

There are two sequences of polynomials $p_0, p_1, p_2, \ldots$ and $q_0, q_1, q_2, \ldots$ that are biorthonormal with respect to the bilinear form (3.4). These polynomials can be determined by applying the non-Hermitian Lanczos process to A with initial vectors u, v such that $u^*v = 1$. Properties and implementation details of the non-Hermitian Lanczos process are discussed, e.g., by Saad [30] and Ye [34]. The following implementation ignores the possibility of breakdown.

Algorithm 3.1: The non-Hermitian Lanczos algorithm
Initialization: $u_1 := u$; $v_1 := v$; $u_0 := 0$; $v_0 := 0$; $\beta_0 := 0$; $\gamma_0 := 0$;

for $j = 1, 2, \ldots, \ell + 1$ do

$$\begin{aligned}
&\alpha_{j-1} := u_j^*(Av_j - \gamma_{j-1}v_{j-1});\\
&r := Av_j - \alpha_{j-1}v_j - \gamma_{j-1}v_{j-1};\\
&s := A^*u_j - \alpha_{j-1}u_j - \beta_{j-1}u_{j-1};\\
&\beta_j := |r^*s|^{1/2}; \gamma_j := r^*s/\beta_j;\\
&v_{j+1} := r/\beta_j;\ u_{j+1} := s/\gamma_j;
\end{aligned}$$

end j

The algorithm is said to break down if some β_j vanishes. Breakdown is likely to take place when the matrix A is sparse and the initial vectors u and v only contain few nonvanishing entries. We will not dwell on this situation. A nice way to handle breakdown is described by Ye [34].

Introduce the matrices

$$U_{\ell+1} = [u_1, \ldots, u_{\ell+1}] \in \mathbb{C}^{m\times(\ell+1)}, \qquad V_{\ell+1} = [v_1, \ldots, v_{\ell+1}] \in \mathbb{C}^{m\times(\ell+1)}.$$

The columns of these matrices are biorthonormal, i.e., $U_{\ell+1}^* V_{\ell+1} = I$. Define the tridiagonal matrix

$$T_{\ell+1} = \begin{bmatrix}
\alpha_0 & \gamma_1 & & & 0\\
\beta_1 & \alpha_1 & \gamma_2 & & \\
& \ddots & \ddots & \ddots & \\
& & \beta_{\ell-1} & \alpha_{\ell-1} & \gamma_\ell\\
0 & & & \beta_\ell & \alpha_\ell
\end{bmatrix} \in \mathbb{C}^{(\ell+1)\times(\ell+1)}. \tag{3.5}$$

We recall that a tridiagonal matrix is said to be unreduced if all its subdiagonal entries are nonvanishing. By the assumption of non-breakdown, all sub- and super-diagonal entries of $T_{\ell+1}$ are nonvanishing. The recursions of Algorithm 3.1 can be written as

$$\begin{aligned}
AV_{\ell+1} &= V_{\ell+1}T_{\ell+1} + \beta_{\ell+1}v_{\ell+2}e_{\ell+1}^*,\\
A^*U_{\ell+1} &= U_{\ell+1}T_{\ell+1}^* + \gamma_{\ell+1}u_{\ell+2}e_{\ell+1}^*.
\end{aligned} \tag{3.6}$$

Moreover, the recursion formulas of Algorithm 3.1 show that

$$u_{j+1} = q_j(A^*)u_1, \qquad v_{j+1} = p_j(A)v_1, \qquad 0 \le j \le \ell + 1, \tag{3.7}$$

for some polynomials p_j and q_j of degree j. It follows from the definition of the measure in (3.4) that

$$\langle q_j, p_k\rangle = u^*q_j(A)p_k(A)v = u_{j+1}^*v_{k+1} = \begin{cases} 1 & j = k,\\ 0 & j \ne k.\end{cases}$$

This shows that the Lanczos polynomials (3.7), which are implicitly determined by Algorithm 3.1, are biorthonormal with respect to the bilinear form (3.4). In particular, the recursion coefficients α_j, β_j, and γ_j computed by the algorithm are recursion coefficients for the biorthonormal polynomials. It follows from (3.7) that the p_j and q_j satisfy the same recursion relations as the vectors u_j and v_j, i.e.,

$$\begin{aligned}
\beta_j p_j(x) &= (x - \alpha_{j-1})p_{j-1}(x) - \gamma_{j-1}p_{j-2}(x),\\
\gamma_j q_j(x) &= (x - \alpha_{j-1})q_{j-1}(x) - \beta_{j-1}q_{j-2}(x),
\end{aligned} \qquad j = 1, 2, \ldots, \ell + 1, \tag{3.8}$$

with $p_{-1}(x) := 0$, $q_{-1}(x) := 0$, $\gamma_0 := 0$, and $\beta_0 := 0$.

Define the row vectors

$$\breve{p}_\ell(x) = [p_0(x), \ldots, p_{\ell-1}(x)], \qquad \breve{q}_\ell(x) = [q_0(x), \ldots, q_{\ell-1}(x)],$$

and let T_ℓ be the leading $\ell \times \ell$ principal submatrix of (3.5), i.e.,

$$T_\ell = \begin{bmatrix} \alpha_0 & \gamma_1 & & & 0 \\ \beta_1 & \alpha_1 & \gamma_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{\ell-2} & \alpha_{\ell-2} & \gamma_{\ell-1} \\ 0 & & & \beta_{\ell-1} & \alpha_{\ell-1} \end{bmatrix}. \tag{3.9}$$

The recurrence relations (3.8) for $0 \leq j < \ell$ can be expressed as

$$\begin{aligned} x\breve{p}_\ell(x) &= \breve{p}_\ell(x)T_\ell + \beta_\ell p_\ell(x)e_\ell^*, \\ x\breve{q}_\ell(x) &= \breve{q}_\ell(x)T_\ell^* + \gamma_\ell q_\ell(x)e_\ell^*. \end{aligned}$$

This shows that the zeros of p_ℓ are eigenvalues of T_ℓ. Conversely, if x is an eigenvalue of T_ℓ, then $p_\ell(x) = 0$. We will assume that T_ℓ has ℓ distinct eigenvalues. Ammar et al. [1, Proposition 2.2] show that this is the case if and only if T_ℓ is diagonalizable (since the matrix is unreduced). We say that a quadrature rule with ℓ nodes (possibly in the complex plane) is a Gauss quadrature rule with respect to the measure $d\mu_{A,u,v}$ if it is exact for all polynomials of degree $2\ell - 1$.

Proposition 3.1 *Let the matrix (3.9) be diagonalizable. Then*

$$\mathcal{G}_\ell f = e_1^* f(T_\ell) e_1 \tag{3.10}$$

is an ℓ-point Gauss quadrature rule associated with the measure $\mathrm{d}\mu_{A,u,v}(x)$ in (3.4), i.e., $\mathcal{G}_\ell$ satisfies

$$\mathcal{G}_\ell f = \mathcal{I} f \qquad \forall f \in \Pi_{2\ell-1}$$

for the integral operator (3.2).

Proof We provide a simple proof for completeness; it is a simplification of a proof presented in [16] for quadrature rules associated with the non-Hermitian block Lanczos process.

The assumption that Algorithm 3.1 does not break down secures that the matrix T_ℓ exists, and the assumption that T_ℓ is diagonalizable guarantees that all eigenvalues are distinct; see [1, Proposition 2.2]. Substituting the spectral factorization of T_ℓ into (3.10) shows that $\mathcal{G}_\ell$ is an ℓ-point quadrature rule with real or complex conjugate nodes.

It follows by induction over the degree of the polynomials p and q that for fixed ℓ,

$$\begin{aligned} p(A)v_1 &= V_\ell p(T_\ell)e_1, & p &\in \Pi_{\ell-1}, \\ V_\ell^* q(A^*)u_1 &= q(T_\ell^*)e_1, & q &\in \Pi_\ell. \end{aligned}$$

Let $f \in \Pi_{2\ell-1}$. Factoring $f = qp$, where $p \in \Pi_{\ell-1}$ and $q \in \Pi_\ell$, yields

$$\mathcal{I} f = u^* q(A)p(A)v = u^* q(A)V_\ell p(T_\ell)e_1 = e_1^* q(T_\ell)p(T_\ell)e_1 = e_1^* f(T_\ell)e_1 = \mathcal{G}_\ell f. \quad \square$$

Introduce for $0 \leq r < \ell$ the reverse matrices

$$\widetilde{T}_{\ell-r,r} = \begin{bmatrix} \alpha_{\ell-1} & \gamma_{\ell-1} & & & 0 \\ \beta_{\ell-1} & \alpha_{\ell-2} & \gamma_{\ell-2} & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{r+2} & \alpha_{r+1} & \gamma_{r+1} \\ 0 & & & \beta_{r+1} & \alpha_r \end{bmatrix} \in \mathbb{C}^{(\ell-r)\times(\ell-r)} \tag{3.11}$$

as well as the concatenated tridiagonal matrices

$$\widehat{T}_{2\ell+1-r,r} = \begin{bmatrix} T_\ell & \gamma_\ell e_\ell & 0 \\ \beta_\ell e_\ell^* & \alpha_\ell & \gamma_{\ell+1} e_1^* \\ 0 & \beta_{\ell+1} e_1 & \widetilde{T}_{\ell-r,r} \end{bmatrix} \in \mathbb{C}^{(2\ell+1-r)\times(2\ell+1-r)}. \tag{3.12}$$

The matrices (3.11) and (3.12) are analogues of the matrices (2.5) and (2.6) of Section 2. We can use the matrix (3.12) to define the quadrature rule

$$\widehat{\mathcal{G}}_{2\ell+1} f = e_1^* f(\widehat{T}_{2\ell+1,0}) e_1, \tag{3.13}$$

which is a generalization of the rule (2.7) to measures $\mathrm{d}\mu_{A,u,v}$ in (3.2). Another generalization is defined at the end of this section.

Proposition 3.2 *Let all leading principal submatrices of $\widehat{T}_{2\ell+1,0}$ be diagonalizable. Then (3.13) is a $(2\ell+1)$-point quadrature rule that is exact for all $f \in \Pi_{2\ell+2}$. If the measure is symmetric with respect to the imaginary axis, then the quadrature rule (3.13) is exact for all $f \in \Pi_{2\ell+3}$. The nodes of the Gauss quadrature rule (3.10) form a subset of the nodes of the rule (3.13).*

Proof Let the measure $\mathrm{d}\mu_{A,u,v}$ in (3.4) be defined by (3.3) and assume for now that this measure does not possess any additional symmetry properties. Consider the matrix $T_{\ell+2}$, which is analogous to the matrix (3.9). We will assume this matrix to be diagonalizable. Then it follows from Proposition 3.1 that

$$\mathcal{G}_{\ell+2} f = e_1^* f(T_{\ell+2}) e_1$$

is an $(\ell+2)$-point Gauss quadrature rule. This rule is exact for all polynomials in $\Pi_{2\ell+3}$. An arbitrary modification of the last diagonal entry of $T_{\ell+2}$ results in a quadrature rule that is exact for all polynomials in $\Pi_{2\ell+2}$. This is a consequence of the recursions of the Chebyshev algorithm, which from the moments $m_j = \int x^j \mathrm{d}\mu_{A,u,v}(x)$, $j = 0, 1, \ldots, 2\ell+2$, computes all entries except the last diagonal entry of $T_{\ell+2}$; the last diagonal entry of $T_{\ell+2}$ is determined by $m_{2\ell+3}$. A modified Chebyshev algorithm is described in, e.g., [7]; the (standard) Chebyshev algorithm is a special case.

We remark that the moments determine the products $\beta_j \gamma_j$ of the off-diagonal entries. The scaling of the β_j is arbitrary.

When the measure $\mathrm{d}\mu_{A,u,v}$ is symmetric about the imaginary axis in the complex plane, all recursion coefficients α_j vanish. The leading $(\ell+2)\times(\ell+2)$ principal submatrix of (3.12) then is $T_{\ell+2}$. This means that the quadrature rule (3.13) is exact for all $f \in \Pi_{2\ell+3}$.

The fact that the nodes of the quadrature rule (3.10) form a subset of the nodes of rule (3.13) can be shown similarly as outlined in the proof of Proposition 2.1. □

Ammar et al. [1] discuss the computation of nodes and weights of Gauss–Kronrod rules associated with a real non-Hermitian tridiagonal matrix. Some computational aspects carry over to the determination of the nodes and weights of the quadrature rule (3.13). However, as already

mentioned in Section 2, the evaluation of (3.13) can be done efficiently for many functions of interest without using the spectral factorization of $\widehat{T}_{2\ell+1}$.

Similarly as in Section 2, we define the quadrature rules

$$\widehat{\mathcal{G}}_{2\ell+1-r,r} f = e_1^* f(\widehat{T}_{2\ell+1-r,r}) e_1, \qquad 1 \leq r < \ell. \tag{3.14}$$

Proposition 3.3 *Under the conditions of Proposition 3.2, the quadrature rules (3.13) and (3.14) are exact for the same class of functions.*

Proof The leading $(\ell+2)\times(\ell+2)$ principal submatrix of the matrix $\widehat{T}_{2\ell+1-r,r}$ in (3.14) is the same as the corresponding leading principal submatrix of the rule (3.13). Therefore, the proof of Proposition 3.2 carries over to the quadrature rules (3.14). □

The following result yields some properties of decompositions that involve the matrix (3.12).

Theorem 3.1 *The matrices T_ℓ and $\widetilde{T}_{\ell,0}$ have the same eigenvalues. Let their spectral factorizations be given by $T_\ell = S_\ell \Lambda_\ell S_\ell^{-1}$ and $\widetilde{T}_{\ell,0} = \widetilde{S}_\ell \Lambda_\ell \widetilde{S}_\ell^{-1}$ with $\Lambda_\ell \in \mathbb{C}^{\ell\times\ell}$ diagonal. Let $V_{\ell+1}$ and $U_{\ell+1}$ be determined by the decompositions (3.6), and let V_ℓ and U_ℓ consist of the first ℓ columns of $V_{\ell+1}$ and $U_{\ell+1}$, respectively. Define the matrices $\widetilde{V}_\ell = V_\ell S_\ell \widetilde{S}_\ell^{-1}$ and $\widetilde{U}_\ell = U_\ell S_\ell \widetilde{S}_\ell^{-1}$, and let $\widehat{V}_{2\ell+1} = [V_{\ell+1}, \widetilde{V}_\ell]$ and $\widehat{U}_{2\ell+1} = [U_{\ell+1}, \widetilde{U}_\ell]$. Then the matrices $A\widehat{V}_{2\ell+1} - \widehat{V}_{2\ell+1}\widehat{T}_{2\ell+1,0}$ and $A\widehat{U}_{2\ell+1} - \widehat{U}_{2\ell+1}\widehat{T}_{2\ell+1,0}$ have rank at most two.*

Proof The matrix (3.11) with $r = 0$ satisfies $T_\ell = S_\ell \widetilde{S}_\ell^{-1} \widetilde{T}_{\ell,0} \widetilde{S}_\ell S_\ell^{-1}$. We obtain, with $h_j = \beta_j v_{j+1}$ for $j \in \{\ell, \ell+1\}$, that

$$\begin{aligned} A\widetilde{V}_\ell &= AV_\ell S_\ell \widetilde{S}_\ell^{-1} = (V_\ell T_\ell + h_\ell e_\ell^*) S_\ell \widetilde{S}_\ell^{-1} \\ &= (V_\ell S_\ell \widetilde{S}_\ell^{-1} \widetilde{T}_{\ell,0} \widetilde{S}_\ell S_\ell^{-1} + h_\ell e_\ell^*) S_\ell \widetilde{S}_\ell^{-1} \\ &= V_\ell S_\ell \widetilde{S}_\ell^{-1} \widetilde{T}_{\ell,0} + h_\ell e_\ell^* S_\ell \widetilde{S}_\ell^{-1} \\ &= \widetilde{V}_\ell \widetilde{T}_{\ell,0} + h_\ell e_\ell^* S_\ell \widetilde{S}_\ell^{-1}. \end{aligned}$$

The above relation combined with (3.6) gives

$$A[V_{\ell+1}, \widetilde{V}_\ell] = [V_{\ell+1} T_{\ell+1}, \widetilde{V}_\ell \widetilde{T}_{\ell,0}] + [h_{\ell+1} e_{\ell+1}^*, h_\ell e_\ell^* S_\ell \widetilde{S}_\ell^{-1}].$$

Moreover,

$$[V_{\ell+1}, \widetilde{V}_\ell]\widehat{T}_{2\ell+1,0} = [V_{\ell+1} T_{\ell+1} + \beta_{\ell+1} v_\ell S_\ell \widetilde{S}_\ell^{-1} e_{\ell+1}^*, \gamma_{\ell+1} v_{\ell+1} e_1^* + \widetilde{V}_\ell \widetilde{T}_{\ell,0}].$$

Therefore,

$$A[V_{\ell+1}, \widetilde{V}_\ell] - [h_{\ell+1} e_{\ell+1}^*, h_\ell e_\ell^* S_\ell \widetilde{S}_\ell^{-1}] = [V_{\ell+1}, \widetilde{V}_\ell]\widehat{T}_{2\ell+1,0} - [\beta_{\ell+1} v_\ell S_\ell \widetilde{S}_\ell^{-1} e_{\ell+1}^*, \gamma_{\ell+1} v_{\ell+1} e_1^*],$$

and rearranging terms yields

$$A\widehat{V}_{2\ell+1} = \widehat{V}_{2\ell+1}\widehat{T}_{2\ell+1,0} + [(h_{\ell+1} - \beta_{\ell+1} v_\ell S_\ell \widetilde{S}_\ell^{-1}) e_{\ell+1}^*, h_\ell e_\ell^* S_\ell \widetilde{S}_\ell^{-1} - \gamma_{\ell+1} v_{\ell+1} e_1^*]. \tag{3.15}$$

We can express $\widehat{U}_{2\ell+1} A^*$ in a similar fashion. This shows the theorem. □

A different extension of the Gauss rule (3.10) is obtained by replacing the matrix (3.11) in (3.12) by its conjugate transpose. This gives the matrix

$$\widehat{T}'_{2\ell+1-r,r} = \begin{bmatrix} T_\ell & \gamma_\ell e_\ell & 0 \\ \beta_\ell e_\ell^* & \alpha_\ell & \gamma_{\ell+1} e_1^* \\ 0 & \beta_{\ell+1} e_1 & \widetilde{T}_{\ell-r,r}^* \end{bmatrix} \tag{3.16}$$

and the associated quadrature rule

$$\widehat{\mathcal{G}}'_{2\ell+1-r,r} f = e_1^* f(\widehat{T}'_{2\ell+1-r,r}) e_1.$$

When $r = 0$, we define

$$\widehat{\mathcal{G}}'_{2\ell+1} f = e_1^* f(\widehat{T}'_{2\ell+1,0}) e_1. \tag{3.17}$$

Analogues of Proposition 3.2, Corollary 3.3, and Theorem 3.1 can be shown. In our experience, the quadrature rules (3.14) and (3.17) yield about the same accuracy. We therefore illustrate the performance of the former rules only in the following section.

4 Numerical examples

We exhibit the accuracy of several of the quadrature rules discussed. All computations were carried out using MATLAB R2010b on a 64-bit DELL personal computer. The first four examples illustrate the performance of the quadrature rules when applied to several integrands f and matrices A; the last three examples are concerned with network analysis.

Table 4.1 Magnitude of relative errors in computed approximations of $F(A) = u^*(I+A^2)^{-1}u$ with A symmetric indefinite.

$m = 100$	$\ell = 6$	$\ell = 10$	$\ell = 12$	$\ell = 14$
$\mathcal{G}_{\ell+1} f$	8.4003×10^{-2}	1.4300×10^{-2}	6.9629×10^{-3}	1.2098×10^{-3}
$\widehat{\mathcal{G}}_{2\ell+1} f$	5.5165×10^{-3}	8.7300×10^{-4}	2.1057×10^{-3}	4.0794×10^{-4}
$\widehat{\mathcal{G}}_{\frac{3}{2}\ell+1,\frac{\ell}{2}} f$	4.2586×10^{-2}	8.8065×10^{-4}	1.6913×10^{-3}	5.5927×10^{-4}
$\widehat{\mathcal{G}}_{\ell+2,\ell-1} f$	6.6907×10^{-2}	4.5364×10^{-3}	1.9222×10^{-3}	1.9274×10^{-3}

Table 4.2 Magnitude of relative differences of computed approximations of $F(A) = u^*(I + A^2)^{-1}u$ with A symmetric indefinite.

$m = 100$	$\ell = 6$	$\ell = 10$	$\ell = 12$	$\ell = 14$
$\lvert(\mathcal{G}_{\ell+1} - \widehat{\mathcal{G}}_{2\ell+1})f\rvert/F(A)$	8.9519×10^{-2}	1.3427×10^{-2}	4.8572×10^{-3}	1.6178×10^{-3}
$\lvert(\widehat{\mathcal{G}}_{\frac{3}{2}\ell+1,\frac{\ell}{2}} - \widehat{\mathcal{G}}_{2\ell+1})f\rvert/F(A)$	3.7069×10^{-2}	1.7537×10^{-3}	4.1449×10^{-4}	1.5132×10^{-4}

Example 4.1 We would like to determine an approximation of the functional $F(A) := u^*(I + A^2)^{-1}u$, where $A \in \mathbb{R}^{100\times100}$ has randomly generated uniformly distributed real eigenvalues in the interval $[-5, 5]$ and a random orthogonal eigenvector matrix, and u is a random vector of unit norm. The exact value of $F(A)$ is 0.2942. Table 4.1 displays the magnitude of the relative errors achieved with quadrature rule $\mathcal{G}_{\ell+1} f$ and with the rules $\widehat{\mathcal{G}}_{2\ell+1-r,r} f$ for $r \in \{0, \ell/2, \ell-1\}$ and several values of ℓ. We note that $r = 0$ and $r = \ell-1$ are the smallest and largest possible r-values.

For all ℓ-values, except for $\ell = 12$, the rules $\widehat{\mathcal{G}}_{2\ell+1-r,r}f$ give the most accurate approximations for $r = 0$. The relative error in $\widehat{\mathcal{G}}_{2\ell+1}f$ can be seen to be smaller than in $\mathcal{G}_{\ell+1}f$.

Table 4.2 shows the relative differences $|(\mathcal{G}_{\ell+1}-\widehat{\mathcal{G}}_{2\ell+1})f|/F(A)$ and $|(\widehat{\mathcal{G}}_{\frac{3}{2}\ell+1,\frac{\ell}{2}}-\widehat{\mathcal{G}}_{2\ell+1})f|/F(A)$. The former differences provide fairly accurate estimates of the magnitude of the relative error in $\mathcal{G}_{\ell+1}f$, but overestimate the magnitude of the relative error in $\widehat{\mathcal{G}}_{2\ell+1}f$. Table 4.2 suggests that $\widehat{\mathcal{G}}_{2\ell+1}f$ be used as an approximation of $\mathcal{I}f$.

The relative differences $|(\mathcal{G}_{\ell+1} - \widehat{\mathcal{G}}_{2\ell+1})f|/F(A)$ are larger than $|(\widehat{\mathcal{G}}_{\frac{3}{2}\ell+1,\frac{\ell}{2}} - \widehat{\mathcal{G}}_{2\ell+1})f|/F(A)$ for all ℓ-values. This suggests that one also may use the latter as an estimate of the magnitude of the relative error in $\widehat{\mathcal{G}}_{2\ell+1}f$, but this may give an underestimate of the actual error.

We note that the derivatives of $f(x) = (1 + x^2)^{-1}$ change sign on $[-5, 5]$. Therefore the technique described in [19,20] for bounding $F(A)$ based on evaluating pairs of Gauss and Gauss–Radau quadrature rules is not guaranteed to yield lower and upper bounds.

This example illustrates the situation when the nodes of generalized averaged Gauss quadrature rules are not contained in the convex hull of the support of the measure, which is the interval $[\lambda_{\min}, \lambda_{\max}]$, where $\lambda_{\min}$ and $\lambda_{\max}$ denote the smallest and largest eigenvalues of the matrix A, respectively. For $\ell = 12$ and $\ell = 14$, the smallest node of $\widehat{\mathcal{G}}_{2\ell+1}$ is smaller than $\lambda_{\min}$. This would make it difficult to approximate the expression $u^*(A - \lambda_{\min}I)^{1/2}u$ by 25- and 29-node generalized averaged Gauss quadrature rules. However, the smallest nodes of the truncated generalized averaged Gauss quadrature rules $\widehat{\mathcal{G}}_{\frac{3}{2}\ell+1,\frac{\ell}{2}}$ for $\ell = 12$ and $\ell = 14$ are strictly larger than $\lambda_{\min}$ and, therefore, could be applied. □

Table 4.3 Magnitude of relative errors of $F(A) = u^* \exp(A)u$ with A a banded nonsymmetric Toeplitz matrix.

$m = 1000$	$\ell = 2$	$\ell = 3$	$\ell = 6$
$\mathcal{G}_{\ell+1}f$	1.9025×10^{-4}	2.1698×10^{-5}	5.7383×10^{-9}
$\widehat{\mathcal{G}}_{2\ell+1}f$	2.3086×10^{-5}	2.2268×10^{-7}	6.7648×10^{-12}
$\widehat{\mathcal{G}}_{\ell+2,\ell-1}f$	2.3251×10^{-5}	1.8832×10^{-6}	2.3465×10^{-10}
$\widehat{\mathcal{G}}_{2\ell+1-r,r}f$	2.9705×10^{-4}	1.9623×10^{-6}	3.8471×10^{-11}

Table 4.4 Magnitude of relative differences of computed approximations of $F(A) = u^* exp(A)u$ with A a banded nonsymmetric Toeplitz matrix.

$m = 1000$	$\ell = 2$	$\ell = 3$	$\ell = 6$
$\lvert(\mathcal{G}_{\ell+1} - \widehat{\mathcal{G}}_{2\ell+1})f\rvert/F(A)$	1.6716×10^{-4}	2.1475×10^{-5}	5.7315×10^{-9}
$\lvert(\widehat{\mathcal{G}}_{2\ell+1-r,r} - \widehat{\mathcal{G}}_{2\ell+1})f\rvert/F(A)$	3.2014×10^{-4}	1.7396×10^{-6}	3.1706×10^{-11}

Example 4.2 This example determines approximations of $F(A) := u^* \exp(A)u$, where $A = [a_{i-j}] \in \mathbb{R}^{1000\times1000}$ is a pentadiagonal nonsymmetric Toeplitz matrix with $a_0 = 1$, $a_1 = 3/2$, $a_2 = 2$, $a_{-1} = 2$, and $a_{-2} = 3$. We let $u = [1/\sqrt{1000}, \ldots, 1/\sqrt{1000}]^* \in \mathbb{R}^{1000}$. The exact value of $F(A)$ is 1.3273×10^4.

Table 4.3 displays the magnitude of the relative errors achieved with the different quadrature rules. We choose $r = \ell/2$ when ℓ is even, otherwise $r = (\ell+1)/2$. The quadrature error for $\widehat{\mathcal{G}}_{2\ell+1}f$ is the smallest for all values of ℓ in this example.

The magnitudes of relative differences between quadrature errors are reported in Table 4.4. Similarly as in Table 4.2, the relative differences $|(\mathcal{G}_{\ell+1} - \widehat{\mathcal{G}}_{2\ell+1})f|/F(A)$ are fairly accurate estimates of the relative errors in $\mathcal{G}_{\ell+1}f$. The relative differences $|(\widehat{\mathcal{G}}_{2\ell+1-r,r} - \widehat{\mathcal{G}}_{2\ell+1})f|/F(A)$

are smaller for $\ell \geq 3$. Since A is nonsymmetric pairs of Gauss and Gauss–Radau rules are not guaranteed to provide lower and upper bounds for $F(A)$. □

Table 4.5 Magnitude of relative errors in computed approximations of $F(A) = u^* \log(A+cI)u$ with A a banded nonsymmetric Toeplitz matrix.

$m = 100$	$\ell = 4$	$\ell = 14$	$\ell = 19$
$\mathcal{G}_{\ell+1}f$	4.5839×10^{-6}	1.7569×10^{-7}	5.1568×10^{-9}
$\widehat{\mathcal{G}}_{2\ell+1}f$	1.4315×10^{-7}	1.5645×10^{-7}	3.2633×10^{-9}
$\widehat{\mathcal{G}}_{\ell+2,\ell-1}f$	1.9361×10^{-6}	1.6364×10^{-7}	4.2304×10^{-9}
$\widehat{\mathcal{G}}_{2\ell+1-r,r}f$	1.8950×10^{-6}	1.1401×10^{-7}	3.6063×10^{-9}

Table 4.6 Magnitude of relative differences of computed approximations of $F(A) = u^* \log(A + cI)v$ with A a banded nonsymmetric Toeplitz matrix.

$m = 100$	$\ell = 4$	$\ell = 14$	$\ell = 19$
$\lvert(\mathcal{G}_{\ell+1} - \widehat{\mathcal{G}}_{2\ell+1})f\rvert/F(A)$	4.4408×10^{-6}	1.9519×10^{-8}	8.4201×10^{-9}
$\lvert(\widehat{\mathcal{G}}_{2\ell+1-r,r} - \widehat{\mathcal{G}}_{2\ell+1})f\rvert/F(A)$	1.7518×10^{-6}	4.2441×10^{-8}	6.8696×10^{-9}

Example 4.3 In this example, we approximate the value of $F(A) := u^* \log(A + cI)u$, where $A = [a_{i-j}] \in \mathbb{R}^{100\times100}$ is a pentadiagonal nonsymmetric Toeplitz matrix with $a_0 = 1$, $a_1 = 3/2$, $a_2 = 2$, $a_{-1} = 2$, and $a_{-2} = 3$. The constant c is set to 4.3. Then all eigenvalues of $A + cI$ have positive real part, some of which are small. We let $u = [1/10, \ldots, 1/10]^* \in \mathbb{R}^{100}$. The exact value of $F(A)$ is 2.6127.

Table 4.5 displays the magnitude of the relative errors achieved with the different quadrature rules. We choose $r = \ell/2$ when ℓ is even, otherwise $r = (\ell+1)/2$. The magnitude of relative differences between quadrature rules is reported in Table 4.6. We observe that the relative differences $|(\mathcal{G}_{\ell+1} - \widehat{\mathcal{G}}_{2\ell+1})f|/F(A)$ are fairly accurate estimates of the relative errors in $\mathcal{G}_{\ell+1}f$. □

The number of steps of the symmetric or nonsymmetric Lanczos process required to determine an approximation of matrix functionals (1.1) depends on the function f and the location of the eigenvalues of the matrix A. For instance, the following example requires more steps than the example above to yield comparable accuracy.

Table 4.7 Magnitude of relative errors in computed approximations of $F(A) = u^* \log(A+cI)v$ with A nonsymmetric.

$m = 200$	$\ell = 9$	$\ell = 19$	$\ell = 24$
$\mathcal{G}_{\ell+1}f$	3.1529×10^{-5}	1.7917×10^{-8}	8.7185×10^{-9}
$\widehat{\mathcal{G}}_{2\ell+1}f$	3.1256×10^{-5}	4.3359×10^{-9}	8.5918×10^{-9}
$\widehat{\mathcal{G}}_{\ell+2,\ell-1}f$	3.1322×10^{-5}	1.2428×10^{-8}	8.5948×10^{-9}
$\widehat{\mathcal{G}}_{2\ell+1-r,r}f$	3.1276×10^{-5}	2.2239×10^{-8}	8.5687×10^{-9}

Example 4.4 We approximate the value $F(A) := u^* \log(A + cI)v$, where $u = e_2, v = e_2 + e_3$. The result is the sum of the matrix entries $[\log(A + cI)]_{2,2}$ and $[\log(A + cI)]_{2,3}$. The matrix $A \in \mathbb{R}^{200\times200}$ has random entries and the constant c is chosen so that all eigenvalues of $A + cI$

Table 4.8 Magnitude of relative differences of computed approximations of $F(A) = u^* \log(A + cI)v$ with A nonsymmetric.

$m = 200$	$\ell = 9$	$\ell = 19$	$\ell = 24$
$\lvert(\mathcal{G}_{\ell+1} - \widehat{\mathcal{G}}_{2\ell+1})f\rvert/F(A)$	2.7347×10^{-7}	2.2253×10^{-8}	1.2668×10^{-10}
$\lvert(\widehat{\mathcal{G}}_{2\ell+1-r,r} - \widehat{\mathcal{G}}_{2\ell+1})f\rvert/F(A)$	2.0600×10^{-8}	2.6575×10^{-8}	2.3078×10^{-11}

have positive real part and some are close to the origin. In our experiment $c = 13.5$. The exact value of $F(A)$ is 0.229.

Table 4.7 displays the magnitude of relative errors achieved with the different quadrature rules. We choose $r = \ell/2$ when ℓ is even, otherwise $r = (\ell+1)/2$. Table 4.8 reports the magnitudes of relative differences between quadrature rules. □

The following three examples illustrate the application of the quadrature rules of this paper to quantities of interest in network analysis. A network is defined by a graph G, which is made up of a set of m vertices $\mathcal{V}$ and a set of edges $\mathcal{E}$. We assume that G is an unweighted connected graph with simple edges, and with no loop of length one. This kind of graphs arise in many applications, including genetics, epidemiology, and telecommunication; see, e.g., Estrada and Higham [12,13]. The adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{m\times m}$ associated with G has the entry $a_{ij} = 1$ if there is an edge between node i and node j; otherwise $a_{ij} = 0$. If the graph G is undirected, then A is symmetric. The size of the entries of $\exp(A)$ is commonly used to measure properties of vertices of a network. For instance, when the diagonal entry $[\exp(A)]_{ii}$ is large in comparison with other diagonal entries, the vertex i is well connected and, therefore, important. A large off-diagonal entry $[\exp(A)]_{ij}$, $i \neq j$, signals that communication between nodes i and j is easy; see [4,5,12, 13,15,16] for discussions.

The *communicability betweenness* of vertex p is defined by

$$\frac{1}{(m-1)(m-2)} \sum_{i\neq p} \sum_{\substack{j\neq p \\ j\neq i}} \frac{[\exp(A)]_{ij} - [\exp(A_p)]_{ij}}{[\exp(A)]_{ij}}, \tag{4.1}$$

where A_p is the adjacency matrix of the graph obtained by removing from G all edges involving node p. This is a measure of the amount of communication passing through node p; see Estrada et al. [14].

The quantity (4.1) is cumbersome to compute when A is large. Therefore, Fenu et al. [16] introduced the *alternative communicability betweenness* of node p, given by

$$\frac{c_p^* \exp(A) c_p - d_p^* \exp(A_p) d_p}{c_p^* \exp(A) c_p}, \tag{4.2}$$

where $c_p \in \mathbb{R}^m$ and $d_p \in \mathbb{R}^{m-1}$ are vectors with all entries one, except for the pth entry, which vanishes. The quotient (4.2) is related to (4.1), but differs from the latter in that it takes into consideration the effect of removing node p on the diagonal elements of $\exp(A)$.

Table 4.9 Magnitude of relative errors in computed approximations of (4.2) for $p = 2$.

	$\ell = 2$	$\ell = 3$	$\ell = 6$
$\mathcal{G}_{\ell+1}$	2.0738×10^{-1}	1.0574×10^{-2}	3.6370×10^{-5}
$\widehat{\mathcal{G}}_{2\ell+1}$	6.4565×10^{-2}	4.0547×10^{-2}	1.1034×10^{-5}
$\widehat{\mathcal{G}}_{\ell+2,\ell-1}$	5.3863×10^{-2}	2.6791×10^{-2}	8.5372×10^{-6}

Example 4.5 We consider a network that describes the protein interaction in yeast. Each edge represents the interaction of two proteins; see Sun et al. [33]. The data set is available at [3]. The graph has 2114 vertices, 4480 edges, and is undirected.

Table 4.9 shows the magnitude of relative errors in computed approximations of alternative communicability betweenness (4.2) of node $p = 2$ determined by quadrature rules discussed in this paper. Only few quadrature nodes are required to yield approximations with higher accuracy than what typically is required in applications. □

Table 4.10 Magnitude of relative errors in computed approximations of $c^* \exp(A)c$, $c = [1, 1, \ldots, 1]^*$.

	$\ell = 2$	$\ell = 3$	$\ell = 6$
$\mathcal{G}_{\ell+1}$	1.7370×10^{-1}	2.8278×10^{-2}	1.8624×10^{-5}
$\widehat{\mathcal{G}}_{2\ell+1}$	3.0256×10^{-2}	6.5685×10^{-4}	1.1743×10^{-6}
$\widehat{\mathcal{G}}_{\ell+2,\ell-1}$	2.0679×10^{-3}	4.7213×10^{-3}	2.0722×10^{-6}

Example 4.6 This example displays estimates of the total communicability determined by the quadrature rules considered in this paper for the yeast network; see the previous example for a description. Table 4.10 shows the averaged Gaussian rule $\widehat{\mathcal{G}}_{2\ell-1}$ to give approximations with the smallest relative errors for $\ell = 3$ and $\ell = 6$ Gaussian nodes. Quite high accuracy is achieved with very few nodes. □

Table 4.11 Magnitude of relative errors in computed approximations of $F(A) = e_2^* \exp(A)e_2$.

	$\ell = 4$	$\ell = 7$	$\ell = 9$
$\mathcal{G}_{\ell+1}f$	1.6516×10^{-3}	7.8789×10^{-9}	1.8381×10^{-12}
$\widehat{\mathcal{G}}_{2\ell+1}f$	6.4931×10^{-4}	5.7566×10^{-10}	2.0508×10^{-14}
$\widehat{\mathcal{G}}_{\ell+2,\ell-1}f$	5.4074×10^{-5}	7.1766×10^{-10}	1.2048×10^{-14}

Example 4.7 Our last example is concerned with a network for air traffic. The adjacency matrix $A \in \mathbb{R}^{2649\times2649}$ is determined by the network defined by flight connections between airports in the United States during May of 2005. There are 2649 airports (nodes) that are connected by 13106 links (edges). The network is directed, i.e., the adjacency matrix is nonsymmetric. The network is available at [27]. Assume that we are interested in the importance of the second airport in the list. We therefore compute approximations of $e_2^* f(A)e_2$, where $f(t) = \exp(t)$ and $e_2 = [0, 1, 0, \ldots, 0]^*$ is the second axis vector.

Table 4.11 shows the magnitude of the relative errors in the computed approximations. Both the quadrature rules $\widehat{\mathcal{G}}_{2\ell+1}$ and $\widehat{\mathcal{G}}_{\ell+2,\ell-1}$ determine more accurate approximations than the Gauss rule $\mathcal{G}_{\ell+1}$. Since A is nonsymmetric, pairs of Gauss and Gauss–Radau quadrature rules are not guaranteed to determine lower and upper bounds for $e_2^* f(A)e_2$. □

5 Conclusion

Matrix functionals (1.1) with a Hermitian matrix A and $v = u$ are commonly approximated by the use of Gauss quadrature rules. We propose that the generalized averaged Gauss quadrature rules (2.7) or (2.8) be used instead of the associated $(\ell + 1)$-point Gauss rule, because the latter

typically yield higher accuracy and require essentially the same computational effort when the matrix A is so large that the dominating computational work in Algorithm 2.1 is the evaluation of matrix-vector products. The relative difference $|(\mathcal{G}_{\ell+1} - \widehat{\mathcal{G}}_{2\ell+1})f|/F(A)$ is found to often provide an upper bound for the magnitude of the relative errors in $\mathcal{G}_{\ell+1}f$ and $\widehat{\mathcal{G}}_{2\ell+1}f$.

New quadrature rules (3.13) and (3.14) are developed for the approximation of matrix functionals (3.1) with a non-Hermitian matrix. The advantages of the rules (3.13) and (3.14) over the corresponding $(\ell+1)$-point Gauss-type rule are the same as of the rules (2.7) or (2.8) for a Hermitian matrix A.

Acknowledgments

LR would like to thank Miodrag Spalević for an enjoyable visit to Belgrade, during which work on this paper was begun. MS was supported in part by the Serbian Ministry of Education, Science and Technological Development (Research Project: "Methods of numerical and nonlinear analysis with applications" (No. #174002)". LR was supported in part by NSF grant DMS-1115385. The authors would like to thank a referee for carefully reading the paper and for suggestions that clarified the presentation.

References

1. G. S. Ammar, D. Calvetti, and L. Reichel, *Computation of Gauss–Kronrod quadrature rules with nonpositive weights*, Electron. Trans. Numer. Anal., 9 (1999), pp. 29–38.
2. Z. Bai, M. Fahey, and G. Golub, *Some large-scale matrix computation problems*, J. Comput. Appl. Math., 74 (1996), pp. 71–89.
3. V. Batagelj and A. Mrvar, *Pajek data sets*, (2006). Available at `http://vlado.fmf.uni-lj.si/pub/networks/data/`.
4. M. Benzi and P. Boito, *Quadrature rule-based bounds for functions of adjacency matrices*, Linear Algebra Appl., 433 (2010), pp. 637–652.
5. M. Benzi and C. Klymko, *Total communicability as a centrality measure*, J. Complex Networks, 1 (2014), pp. 124–149.
6. C. F. Borges and W. B. Gragg, *A parallel divide and conquer algorithm for the generalized symmetric definite tridiagonal eigenvalue problem*, in Numerical Linear Algebra, eds. L. Reichel, A. Ruttan, and R. S. Varga, de Gruyter, Berlin, 1993, pp. 11–29.
7. D. Calvetti, G. H. Golub, and L. Reichel, *An adaptive Chebyshev iterative method for nonsymmetric linear systems of equations based on modified moments*, Numer. Math., 67 (1997), pp. 21–40.
8. D. Calvetti, G. H. Golub, W. B. Gragg, and L. Reichel, *Computation of Gauss–Kronrod quadrature rules*, Math. Comp., 69 (2000), pp. 1035–1052.
9. D. Calvetti, L. Reichel, and F. Sgallari, *Applications of anti-Gauss quadrature rules in linear algebra*, in Applications and Computation of Orthogonal Polynomials, eds. W. Gautschi, G. H. Golub, and G. Opfer, Birkhäuser, Basel, 1999, pp. 41–56.
10. C. W. Clenshaw and A. R. Curtis, *A method for numerical integration on an automatic computer*, Numer. Math., 2 (1960), pp. 197–205.
11. S. Ehrich, *On stratified extensions of Gauss–Laguerre and Gauss–Hermite quadrature formulas*, J. Comput. Appl. Math., 140 (2002), pp. 291–299.
12. E. Estrada, *The Structure of Complex Networks*, Oxford University Press, Oxford, 2012.
13. E. Estrada and D. J. Higham, *Network properties revealed through matrix functions*, SIAM Rev., 52 (2010), pp. 696–714.
14. E. Estrada, D. J. Higham, and N. Hatano, *Communicability betweenness in complex networks*, Physica A, 388 (2009), pp. 764–774.
15. C. Fenu, D. Martin, L. Reichel, and G. Rodriguez, *Network analysis via partial spectral factorization and Gauss quadrature*, SIAM J. Sci. Comput., 35 (2013), pp. A2046–A2068.
16. C. Fenu, D. Martin, L. Reichel, and G. Rodriguez, *Block Gauss and anti-Gauss quadrature with application to networks*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 1655–1684.
17. W. Gautschi, *On generating orthogonal polynomials*, SIAM J. Sci. Stat. Comput., 3 (1982), pp. 289–317.

18. W. Gautschi, *Orthogonal Polynomials: Approximation and Computation*, Oxford University Press, Oxford, 2004.
19. G. H. Golub and G. Meurant, *Matrices, moments and quadrature*, in Numerical Analysis 1993, eds. D. F. Griffiths and G. A. Watson, Longman, Essex, England, 1994, pp. 105–156.
20. G. H. Golub and G. Meurant, *Matrices, Moments and Quadrature with Applications*, Princeton University Press, Princeton, 2010.
21. G. H. Golub and J. H. Welsch, *Calculation of Gauss quadrature rules*, Math. Comp., 23 (1969), pp 221–230.
22. W. B. Gragg, *Matrix interpretation and application of the continued fraction algorithm*, Rocky Mountain J. Math., 4 (1974), pp. 213–225.
23. N. J. Higham, *Functions of Matrices: Theory and Computation*, SIAM, Philadelphia, 2008.
24. D. P. Laurie, *Calculation of Gauss–Kronrod quadrature rules*, Math. Comp., 66 (1997), pp. 1133–1145.
25. D. R. Martin and L. Reichel, *Minimization of functionals on the solution of a large-scale discrete ill-posed problem*, BIT, 53 (2013), pp. 153–173.
26. G. Monegato, *An overview of the computational aspects of Kronrod quadrature rules*, Numer. Algorithms, 26 (2001), pp. 173–196.
27. Lev Muchnik's Data Sets web page, `http://www.levmuchnik.net/Content/Networks/NetworkData.html#USAirTrafficData`
28. F. Peherstorfer, *Positive quadrature formulas III: Asymptotics of weights*, Math. Comp., 77 (2008), pp. 2241–2259.
29. Y. Saad, *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 29 (1992), pp. 209–228.
30. Y. Saad, *Iterative methods for sparse linear systems*, SIAM, Philadelphia, 2003.
31. M. M. Spalević, *On generalized averaged Gaussian formulas*, Math. Comp., 76 (2007), pp. 1483–1492.
32. M. M. Spalević, *A note on generalized averaged Gaussian formulas*, Numer. Algorithms, 46 (2007), pp. 253–264.
33. 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.
34. Q. Ye, *A breakdown-free variation of the nonsymmetric Lanczos algorithm*, Math. Comp., 62 (1994), pp. 179–207.