AN ITERATIVE METHOD WITH ERROR ESTIMATORS

D. CALVETTI *, S. MORIGI †, L. REICHEL ‡, AND F. SGALLARI §

Abstract. Iterative methods for the solution of linear systems of equations produce a sequence of approximate solutions. In many applications it is desirable to be able to compute estimates of the norm of the error in the approximate solutions generated and terminate the iterations when the estimates are sufficiently small. This paper presents a new iterative method based on the Lanczos process for the solution of linear systems of equations with a symmetric matrix. The method is designed to allow the computation of estimates of the Euclidean norm of the error in the computed approximate solutions. These estimates are determined by evaluating certain Gauss, anti-Gauss, or Gauss-Radau quadrature rules.

Key words. Lanczos process, conjugate gradient method, symmetric linear system, Gauss quadrature.

1. Introduction.

Large linear systems of equations

$$Ax = b, \qquad A \in \mathbb{R}^{n\times n}, \qquad x \in \mathbb{R}^n, \qquad b \in \mathbb{R}^n, \tag{1}$$

with a nonsingular symmetric matrix are frequently solved by iterative methods, such as the conjugate gradient method and variations thereof; see, e.g., Golub and Van Loan [12, Chapter 10] or Saad [17, Chapter 6]. It is the purpose of the present paper to describe a modification of the conjugate gradient method that allows the computation of bounds or estimates of the norm of the error in the computed approximate solutions.

Assume for notational simplicity that the initial approximate solution of (1) is given by $\boldsymbol{x}_0 = \boldsymbol{0}$, and let Π_{k-1} denote the set of polynomials of degree at most $k-1$. The iterative method of this paper yields approximate solutions of (1) of the form

$$\boldsymbol{x}_k = q_{k-1}(A)\boldsymbol{b}, \qquad k = 1, 2, \dots , \tag{2}$$

where the iteration polynomials $q_{k-1} \in \Pi_{k-1}$ are determined by the method.

The residual error associated with $\boldsymbol{x}_k$ is defined by

$$\boldsymbol{r}_k := \boldsymbol{b} - A\boldsymbol{x}_k, \tag{3}$$

and the error in $\boldsymbol{x}_k$ is given by

$$\boldsymbol{e}_k := A^{-1}\boldsymbol{r}_k. \tag{4}$$

Using (3) and (4), we obtain

$$\boldsymbol{e}_k^T\boldsymbol{e}_k = \boldsymbol{r}_k^T A^{-2}\boldsymbol{r}_k = \boldsymbol{b}^T A^{-2}\boldsymbol{b} - 2\boldsymbol{b}^T A^{-1}\boldsymbol{x}_k + \boldsymbol{x}_k^T\boldsymbol{x}_k. \tag{5}$$

Thus, the Euclidean norm of $\boldsymbol{e}_k$ can be evaluated by computing the terms in the right-hand side of (5). The evaluation of the term $\boldsymbol{x}_k^T\boldsymbol{x}_k$ is straightforward. This paper

*Department of Mathematics, Case Western Reserve University, Cleveland, OH 44106. E-mail: dxc57@po.cwru.edu. Research supported in part by NSF grant DMS-9806702.

†Dipartimento di Matematica, Universitá di Bologna, Bologna, Italy. E-mail: morigi@dm.unibo.it.

‡Department of Mathematics and Computer Science, Kent State University, Kent, OH 44242. E-mail: reichel@mcs.kent.edu. Research supported in part by NSF grant DMS-9806413.

§Dipartimento di Matematica, Universitá di Bologna, Bologna, Italy. E-mail: sgallari@dm.unibo.it.

discusses how to evaluate bounds or estimates of the other terms in the right-hand side of (5). The evaluation is made possible by requiring that the iteration polynomials satisfy

$$q_{k-1}(0) = 0, \qquad k = 1, 2, \dots . \tag{6}$$

Then $\boldsymbol{b}^T A^{-1}\boldsymbol{x}_k = \boldsymbol{b}^T A^{-1} q_{k-1}(A)\boldsymbol{b}$ can be computed for every k without using A^{-1}, and this makes easy evaluation of the middle term in the right-hand side of (5) possible. The iterative method obtained is closely related to the SYMMLQ method, see, e.g., Paige and Saunders [16] or Fischer [8, Section 6.5], and can be applied to solve linear systems of equations (1) with a positive definite or indefinite symmetric matrix. Details of the method are presented in Section 2.

Section 3 discusses how bounds or estimates of the first term in the right-hand side of (5) can be computed by evaluating certain quadrature rules of Gauss-type. Specifically, when the matrix A is positive definite and we have evaluated $\boldsymbol{x}_k$, a lower bound of $\boldsymbol{b}^T A^{-2}\boldsymbol{b}$ can be computed inexpensively by evaluating a k-point Gauss quadrature rule. An estimate of an upper bound is obtained by evaluating an associated k-point anti-Gauss rule. When A is indefinite, an estimate of the Euclidean norm of the error $\boldsymbol{e}_k$ is obtained by evaluating a $(k+1)$-point Gauss-Radau quadrature rule with a fixed node at the origin. We also describe how the quadrature rules can be updated inexpensively when k is increased. Section 4 presents a few computed examples, and Section 5 contains concluding remarks.

The application of quadrature rules of Gauss-type to the computation of error bounds for approximate solutions generated by an iterative method was first described by Dahlquist et al. [6], who discussed the Jacobi iteration method. When the matrix A is symmetric and positive definite, the linear system (1) can conveniently be solved by the conjugate gradient method. Dahlquist et al. [7], and subsequently Golub and Meurant [10, 14], describe methods for computing bounds in the A-norm of approximate solutions determined by the conjugate gradient method. A new approach, based on extrapolation, for computing estimates of the norm of the error in approximate solutions determined by iterative methods has recently been proposed by Brezinski [1].

Assume for the moment that the matrix A in (1) is symmetric and positive definite, and approximate solutions $\boldsymbol{x}_k$ of the linear system (1) are computed by the conjugate gradient method. The method of Golub and Meurant [10] for computing upper bounds for the A-norm of the error in the approximate solutions requires that a lower positive bound for the smallest eigenvalue of the matrix A is available, and so does the scheme by Meurant [14], based on two-point Gauss quadrature rules, for computing upper bounds of the Euclidean norm of the error in the iterates. Estimates of the smallest eigenvalue can be computed by using the connection between the conjugate gradient method and the Lanczos method, see, e.g., Golub and Van Loan [12, Chapter 10]; however, it is generally difficult to determine positive lower bounds. The methods of the present paper for computing error estimates do not require knowledge of any of the eigenvalues of the matrix A.

The performance of iterative methods is often enhanced by the use of preconditioners; see, e.g., Golub and Van Loan [12, Chapter 10] and Saad [17, Chapters 9-10]. In the present paper, we assume that the linear system of equations (1) represents the preconditioned system. Alternatively, one can let (1) represent the unpreconditioned linear system and modify the iterative method to incorporate the preconditioner. Meurant [15] shows how the computation of upper and lower bounds of the A-norm of the error in approximate solutions determined by the conjugate gradient method can be carried out when this approach is used. Analogous formulas can be derived for the iterative method of the present paper.

2. The iterative method. This section presents an iterative method for the solution of linear systems of equations (1) with a nonsingular symmetric matrix A. The description is divided into two subsections, the first of which discusses basic properties of the method. The second subsection derives updating formulas for the approximate solutions $\boldsymbol{x}_k$ computed. The method may be considered a modification of the conjugate gradient method or of the SYMMLQ method, described, e.g., in [8, 16].

Our description uses the spectral factorization

$$\begin{array}{rcll} A &=& U_n \Lambda_n U_n^T, \qquad U_n \in \mathbb{R}^{n\times n}, & U_n^T U_n = I_n, \\ \Lambda_n &=& \mathrm{diag}[\lambda_1, \lambda_2, \dots, \lambda_n], & \lambda_1 \le \lambda_2 \le \cdots \le \lambda_n. \end{array} \tag{7}$$

Here and throughout this paper, I_j denotes the identity matrix of order j. Let $\hat{\boldsymbol{b}} = [\hat{b}_1, \hat{b}_2, \dots, \hat{b}_n]^T := U_n^T \boldsymbol{b}$ and express the matrix functional

$$F(A) := \boldsymbol{b}^T f(A) \boldsymbol{b}, \qquad f(t) := 1/t^2, \tag{8}$$

as a Stieltjes integral

$$F(A) = \hat{\boldsymbol{b}}^T f(\Lambda_n) \hat{\boldsymbol{b}} = \sum_{k=1}^{n} f(\lambda_k) \hat{b}_k^2 = \int_{-\infty}^{\infty} f(t) d\omega(t). \tag{9}$$

The measure ω is a nondecreasing step function with jump discontinuities at the eigenvalues λ_k of A. We will use the notation

$$\mathcal{I}(f) := \int_{-\infty}^{\infty} f(t) d\omega(t). \tag{10}$$

2.1. Basic properties. Our method is based on the Lanczos process. Given the right-hand side vector $\boldsymbol{b}$, k steps of the Lanczos process yield the Lanczos decomposition

$$A V_k = V_k T_k + \boldsymbol{f}_k \tilde{\boldsymbol{e}}_k^T, \tag{11}$$

where $V_k = [\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_k] \in \mathbb{R}^{n\times k}$ and $\boldsymbol{f}_k \in \mathbb{R}^n$ satisfy $V_k^T V_k = I_k$, $V_k^T \boldsymbol{f}_k = \boldsymbol{0}$ and

$$\boldsymbol{v}_1 = \boldsymbol{b}/\|\boldsymbol{b}\|. \tag{12}$$

Moreover, $T_k \in \mathbb{R}^{k\times k}$ is symmetric and tridiagonal. Throughout this paper $\tilde{\boldsymbol{e}}_j$ denotes the jth axis vector and $\|\cdot\|$ the Euclidean vector norm. We may assume that T_k has nonvanishing subdiagonal entries; otherwise the Lanczos process breaks down and the solution of (1) can be computed as a linear combination of the columns $\boldsymbol{v}_j$ generated before break down.

Equation (11) defines a recursion relation for the columns of V_k. This relation, combined with (12), shows that

$$\boldsymbol{v}_j = s_{j-1}(A) \boldsymbol{b}, \qquad 1 \le j \le k, \tag{13}$$

for certain polynomials s_{j-1} of degree $j-1$. These polynomials are orthogonal with respect to the following inner product induced by (10) for functions g and h defined on the spectrum of A,

$$(g,h) := \mathcal{I}(gh). \tag{14}$$

We have

$$\begin{aligned}(s_{j-1}, s_{\ell-1}) &= \int_{-\infty}^{\infty} s_{j-1}(t)s_{\ell-1}(t)d\omega(t) = \boldsymbol{b}^T U_n s_{j-1}(\Lambda_n)s_{\ell-1}(\Lambda_n)U_n^T\boldsymbol{b} \\ &= \boldsymbol{b}^T s_{j-1}(A)s_{\ell-1}(A)\boldsymbol{b} \\ &= \boldsymbol{v}_j^T\boldsymbol{v}_\ell = \begin{cases} 0, & j\neq \ell, \\ 1, & j=\ell, \end{cases}\end{aligned} \tag{15}$$

where we have applied manipulations analogous to those used in equation (9). The last equality of (15) follows from the orthogonality of the columns $\boldsymbol{v}_j$ of V_k. Since the polynomial s_ℓ is of degree ℓ, the columns of V_k span the Krylov subspace

$$\mathbb{K}_k(A,\boldsymbol{b}) := \text{span}\{\boldsymbol{b}, A\boldsymbol{b}, \ldots, A^{k-1}\boldsymbol{b}\},$$

i.e.,

$$\text{range}(V_k) = \mathbb{K}_k(A,\boldsymbol{b}). \tag{16}$$

We also will use the following form of the Lanczos decomposition

$$AV_{k-1} = V_k T_{k,k-1}, \tag{17}$$

where $T_{k,k-1}$ is the leading principal $k\times(k-1)$ submatrix of T_k.

Introduce the QR-factorization of T_k, i.e., let

$$T_k = Q_kR_k, \qquad Q_k, R_k \in \mathbb{R}^{k\times k}, \qquad Q_k^TQ_k = I_k, \tag{18}$$

where $R_k = [r_{j\ell}^{(k)}]_{j,\ell=1}^k$ is upper triangular. Also define

$$T_{k,k-1} = Q_k \begin{bmatrix} \bar{R}_{k-1} \\ 0 \end{bmatrix} = Q_{k,k-1}\bar{R}_{k-1}, \tag{19}$$

where $\bar{R}_{k-1}$ is the leading principal submatrix of order $k-1$ of R_k, and $Q_{k,k-1} \in \mathbb{R}^{k\times(k-1)}$ consists of the first $k-1$ columns of Q_k. For definiteness, we assume that the diagonal entries in the triangular factors in all QR-factorizations of this paper are nonnegative.

The following manipulations of the Lanczos decomposition (11) give an iterative method, whose associated iteration polynomials satisfy (6). The manipulations are closely related to those required in the derivation of the implicitly restarted Lanczos method; see, e.g., [5]. Substituting the QR-factorization (18) into the Lanczos decomposition (11) yields

$$AV_k = V_kQ_kR_k + \boldsymbol{f}_k\tilde{\boldsymbol{e}}_k^T, \tag{20}$$

which after multiplication by Q_k from the right gives

$$A\tilde{V}_k = \tilde{V}_k\tilde{T}_k + \boldsymbol{f}_k\tilde{\boldsymbol{e}}_k^TQ_k, \qquad \tilde{V}_k := V_kQ_k, \qquad \tilde{T}_k := R_kQ_k. \tag{21}$$

The matrix $\tilde{V}_k = [\tilde{\boldsymbol{v}}_1^{(k)}, \tilde{\boldsymbol{v}}_2^{(k)}, \dots, \tilde{\boldsymbol{v}}_k^{(k)}]$ has orthonormal columns and $\tilde{T}_k$ is the symmetric tridiagonal matrix obtained from T_k by applying one step of the QR-algorithm with shift zero.

A relation between the first columns $\boldsymbol{v}_1$ and $\tilde{\boldsymbol{v}}_1^{(k)}$ of V_k and $\tilde{V}_k$, respectively, is easily shown. Assume that $k > 1$ and multiply (20) by $\tilde{\boldsymbol{e}}_1$ from the right. We obtain

$$AV_k\tilde{\boldsymbol{e}}_1 = \tilde{V}_k R_k \tilde{\boldsymbol{e}}_1 + \boldsymbol{f}_k \tilde{\boldsymbol{e}}_k^T \tilde{\boldsymbol{e}}_1,$$

which simplifies to

$$A\boldsymbol{v}_1 = r_{11}^{(k)} \tilde{\boldsymbol{v}}_1^{(k)},$$

where we have used that $R_k\tilde{\boldsymbol{e}}_1 = r_{11}^{(k)}\tilde{\boldsymbol{e}}_1$. Thus,

$$\tilde{\boldsymbol{v}}_1^{(k)} = A\boldsymbol{b}/\|A\boldsymbol{b}\|.$$

Since T_k is tridiagonal, the orthogonal matrix Q_k in the QR-factorization (18) is of upper Hessenberg form. It follows that all but the last two components of the vector $\tilde{\boldsymbol{e}}_k^T Q_k$ are guaranteed to vanish. Therefore, the decomposition (21) differs from a Lanczos decomposition in that the last two columns of the matrix $\boldsymbol{f}_k \tilde{\boldsymbol{e}}_k^T Q_k$ may be nonvanishing.

Let $\bar{V}_{k-1}$ be the matrix made up by the first $k-1$ columns of $\tilde{V}_k$. Note that

$$\bar{V}_{k-1} = V_k Q_{k,k-1}, \tag{22}$$

where $Q_{k,k-1}$ is defined by (19). Generally, $\bar{V}_{k-1} \neq \tilde{V}_{k-1}$; see Subsection 2.2 for details. Removing the last column from each term in equation (21) yields the decomposition

$$A\bar{V}_{k-1} = \bar{V}_{k-1}\tilde{T}_{k-1} + \bar{\boldsymbol{f}}_{k-1}\tilde{\boldsymbol{e}}_{k-1}^T, \tag{23}$$

where $\bar{V}_{k-1}^T \bar{\boldsymbol{f}}_{k-1} = \boldsymbol{0}$, $\bar{V}_{k-1}^T \bar{V}_{k-1} = I_{k-1}$ and $\tilde{T}_{k-1}$ is the leading principal submatrix of order $k-1$ of the matrix $\tilde{T}_k$. Thus, the decomposition (23) is a Lanczos decomposition with initial vector $\tilde{\boldsymbol{v}}_1^{(k)}$ of $\bar{V}_{k-1}$ proportional to $A\boldsymbol{b}$. Analogously to (16), we have

$$\text{range}(\bar{V}_{k-1}) = \mathbb{K}_{k-1}(A, A\boldsymbol{b}). \tag{24}$$

We determine the iteration polynomials (2), and thereby the approximate solutions $\boldsymbol{x}_k$ of (1), by requiring that

$$\boldsymbol{x}_k = q_{k-1}(A)\boldsymbol{b} = \bar{V}_{k-1}\boldsymbol{z}_{k-1} \tag{25}$$

for some vector $\boldsymbol{z}_{k-1} \in \mathbb{R}^{k-1}$. It follows from (24) that any polynomial q_{k-1} determined by (25) satisfies (6). We choose $\boldsymbol{z}_{k-1}$, and thereby $q_{k-1} \in \Pi_{k-1}$, so that the residual error (3) associated with the approximate solution $\boldsymbol{x}_k$ of (1) satisfies the Petrov-Galerkin equation

$$\boldsymbol{0} = V_{k-1}^T \boldsymbol{r}_k = V_{k-1}^T \boldsymbol{b} - V_{k-1}^T A \bar{V}_{k-1} \boldsymbol{z}_{k-1}, \tag{26}$$

which, by using (12) and the factorization (22), simplifies to

$$\|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1 = (AV_{k-1})^T V_k Q_{k,k-1} \boldsymbol{z}_{k-1}. \tag{27}$$

We remark that if the matrix $\bar{V}_{k-1}$ in (26) is replaced by V_{k-1}, then the standard SYMMLQ method [16] is obtained. The iteration polynomial q_{k-1} associated with the standard SYMMLQ method, in general, does not satisfy the condition (6). The implementation of our method uses the QR-factorization of the matrix T_k, similarly as the implementation of the SYMMLQ method described by Fischer [8, Section 6.5]. In contrast, the implementation of the SYMMLQ method presented by Paige and Saunders [16] is based on the LQ-factorization of T_k.

It follows from (17) and (19) that

$$(AV_{k-1})^T V_k Q_{k,k-1} = T_{k,k-1}^T Q_{k,k-1} = \bar{R}_{k-1}^T. \tag{28}$$

Substituting (28) into (27) yields

$$\bar{R}_{k-1}^T \boldsymbol{z}_{k-1} = \|\boldsymbol{b}\| \tilde{\boldsymbol{e}}_1. \tag{29}$$

This defines the iterative method.

Recursion formulas for updating the approximate solutions $\boldsymbol{x}_k$ inexpensively are derived in Subsection 2.2. In the remainder of this subsection, we discuss how to evaluate the right-hand side of (5). Equations (24) and (25) show that $\boldsymbol{x}_k \in \mathbb{K}_{k-1}(A, A\boldsymbol{b})$, and therefore there is a vector $\boldsymbol{y}_{k-1} \in \mathbb{R}^{k-1}$, such that

$$A^{-1}\boldsymbol{x}_k = V_{k-1}\boldsymbol{y}_{k-1}. \tag{30}$$

Thus, by (17),

$$\boldsymbol{x}_k = AV_{k-1}\boldsymbol{y}_{k-1} = V_k T_{k,k-1}\boldsymbol{y}_{k-1}, \tag{31}$$

and, by (25) and (22), we have

$$\boldsymbol{x}_k = V_k Q_{k,k-1}\boldsymbol{z}_{k-1}.$$

It follows that

$$Q_{k,k-1}\boldsymbol{z}_{k-1} = T_{k,k-1}\boldsymbol{y}_{k-1}. \tag{32}$$

Multiplying this equation by $Q_{k,k-1}^T$ yields, in view of (19), that

$$\boldsymbol{z}_{k-1} = Q_{k,k-1}^T T_{k,k-1}\boldsymbol{y}_{k-1} = \bar{R}_{k-1}\boldsymbol{y}_{k-1}. \tag{33}$$

Application of (30), (12), (33) and (29), in order, yields

$$\boldsymbol{b}^T A^{-1}\boldsymbol{x}_k = \boldsymbol{b}^T V_{k-1}\boldsymbol{y}_{k-1} = \|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1^T \boldsymbol{y}_{k-1} = \|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1^T \bar{R}_{k-1}^{-1}\boldsymbol{z}_{k-1} = \boldsymbol{z}_{k-1}^T \boldsymbol{z}_{k-1}. \tag{34}$$

It follows from (25) that $\boldsymbol{x}_k^T\boldsymbol{x}_k = \boldsymbol{z}_{k-1}^T\boldsymbol{z}_{k-1}$. This observation and (34) show that equation (5) can be written in the form

$$\boldsymbol{e}_k^T\boldsymbol{e}_k = \boldsymbol{r}_k^T A^{-2}\boldsymbol{r}_k = \boldsymbol{b}^T A^{-2}\boldsymbol{b} - \boldsymbol{z}_{k-1}^T\boldsymbol{z}_{k-1}. \tag{35}$$

The term $\boldsymbol{z}_{k-1}^T\boldsymbol{z}_{k-1}$ is straightforward to evaluate from (29). Section 3 describes how easily computable upper and lower bounds, or estimates, of $\boldsymbol{b}^T A^{-2}\boldsymbol{b}$ can be derived by using Gauss-type quadrature rules. In this manner, we obtain easily computable upper and lower bounds, or estimates, of the norm of $\boldsymbol{e}_k$. Details are described in Section 3.

Assume for the moment that n steps of the Lanczos process have been carried out to yield the Lanczos decomposition $AV_n = V_nT_n$, analogous to (11). Using the QR-factorization (18) of T_n and the property (12) yields

$$\begin{aligned}\boldsymbol{b}^T A^{-2}\boldsymbol{b} &= \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T V_n^T A^{-2} V_n \tilde{\boldsymbol{e}}_1 = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T T_n^{-2} \tilde{\boldsymbol{e}}_1 \\ &= \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T R_n^{-1} R_n^{-T} \tilde{\boldsymbol{e}}_1.\end{aligned}$$

Substituting this expression into (35) and using (29) shows that

$$\boldsymbol{e}_k^T \boldsymbol{e}_k = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T R_n^{-1} R_n^{-T} \tilde{\boldsymbol{e}}_1 - \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T \bar{R}_{k-1}^{-1} \bar{R}_{k-1}^{-T} \tilde{\boldsymbol{e}}_1. \tag{36}$$

The right-hand side of (36) is analogous to expressions for the A-norm of the error $\boldsymbol{e}_k$ discussed in [10, 11, 14].

2.2. Updating formulas for the iterative method. We describe how the computation of the iterates $\boldsymbol{x}_k$ defined by (25) can be organized so that storage of only a few n-vectors is required.

Let the matrix T_k in (11) have the entries

$$T_k = \begin{bmatrix} \alpha_1 & \beta_1 & & & & & 0 \\ \beta_1 & \alpha_2 & \beta_2 & & & & \\ & \beta_2 & \alpha_3 & & & & \\ & & & \ddots & & & \\ & & \ddots & \ddots & \beta_{k-2} & & \\ & & & \beta_{k-2} & \alpha_{k-1} & \beta_{k-1} \\ 0 & & & & \beta_{k-1} & \alpha_k \end{bmatrix} \in \mathbb{R}^{k\times k}, \tag{37}$$

where according to the discussion following equation (12) we may assume that the β_j are nonvanishing. This property of the β_j secures that the eigenvalues of T_k are distinct. Introduce the spectral factorization

$$\begin{aligned} T_k &= W_k \Theta_k W_k^T, \qquad W_k \in \mathbb{R}^{k\times k}, \qquad W_k^T W_k = I_k, \\ \Theta_k &= \mathrm{diag}[\theta_1^{(k)}, \theta_2^{(k)}, \ldots, \theta_k^{(k)}], \qquad \theta_1^{(k)} < \theta_2^{(k)} < \cdots < \theta_k^{(k)}. \end{aligned} \tag{38}$$

The QR-factorization (18) of T_k is computed by applying $k-1$ Givens rotations

$$G_k^{(j)} := \begin{bmatrix} I_{j-1} & & & \\ & c_j & s_j & \\ & -s_j & c_j & \\ & & & I_{k-j-1} \end{bmatrix} \in \mathbb{R}^{k\times k}, \quad c_j^2 + s_j^2 = 1, \quad s_j \geq 0, \tag{39}$$

to T_k, i.e.,

$$R_k := G_k^{(k-1)} G_k^{(k-2)} \cdots G_k^{(1)} T_k, \qquad Q_k := G_k^{(1)T} G_k^{(2)T} \cdots G_k^{(k-1)T}; \tag{40}$$

see, e.g., [12, Chapter 5] for a discussion on Givens rotations. In our iterative method the matrix Q_k is not explicitly formed; instead we use the representation (40). Since T_k is tridiagonal, the upper triangular matrix R_k has nonvanishing entries on the diagonal and the two adjacent superdiagonals only.

The matrix T_k in (37) is determined by k steps of the Lanczos process. After an additional step, we obtain the Lanczos decomposition

$$AV_{k+1} = V_{k+1}T_{k+1} + \boldsymbol{f}_{k+1}\tilde{\boldsymbol{e}}_{k+1}^T, \tag{41}$$

analogous to (11). For future reference, we remark that the last subdiagonal entry of the symmetric tridiagonal matrix T_{k+1} may be computed by

$$\beta_k := \|\boldsymbol{f}_k\| \tag{42}$$

already after completion of k Lanczos steps.

The matrix T_{k+1} has the QR-factorization

$$T_{k+1} = Q_{k+1}R_{k+1}, \tag{43}$$

whose factors can be computed from Q_k and R_k in a straightforward manner. We have

$$\begin{aligned} Q_{k+1} &= \begin{bmatrix} Q_k & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 \end{bmatrix} G_{k+1}^{(k)T} \in \mathbb{R}^{(k+1)\times(k+1)}, \\ Q_{k+1,k} &= \begin{bmatrix} Q_k & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 \end{bmatrix} G_{k+1,k}^{(k)T} \in \mathbb{R}^{(k+1)\times k}, \end{aligned} \tag{44}$$

where $G_{k+1}^{(k)}$ is defined by (39) and $G_{k+1,k}^{(k)} \in \mathbb{R}^{(k+1)\times k}$ is made up of the first k columns of $G_{k+1}^{(k)}$.

We obtain updating formulas for computing the triangular matrix R_{k+1} in (43) from the matrix R_k in (40) by expressing these matrices in terms of their columns

$$R_k = [\boldsymbol{r}_1^{(k)}, \boldsymbol{r}_2^{(k)}, \ldots, \boldsymbol{r}_k^{(k)}], \qquad R_{k+1} = [\boldsymbol{r}_1^{(k+1)}, \boldsymbol{r}_2^{(k+1)}, \ldots, \boldsymbol{r}_k^{(k+1)}, \boldsymbol{r}_{k+1}^{(k+1)}].$$

Comparing (18) and (43) yields

$$\boldsymbol{r}_j^{(k+1)} = \begin{bmatrix} \boldsymbol{r}_j^{(k)} \\ 0 \end{bmatrix}, \qquad 1 \le j < k, \tag{45}$$

and

$$\begin{aligned} \boldsymbol{r}_k^{(k+1)} &= G_{k+1}^{(k)} \begin{bmatrix} \boldsymbol{r}_k^{(k)} \\ \beta_k \end{bmatrix}, \\ \boldsymbol{r}_{k+1}^{(k+1)} &= G_{k+1}^{(k)} G_{k+1}^{(k-1)} T_{k+1} \tilde{\boldsymbol{e}}_{k+1}. \end{aligned} \tag{46}$$

Thus, the entries of all the matrices $R_1, R_2, \ldots, R_{k+1}$ can be computed in only $O(k)$ arithmetic floating-point operations.

The matrix $\bar{R}_k = [\bar{r}_{j\ell}^{(k)}]_{j,\ell=1}^k$ defined by (19) is the leading principal submatrix of R_{k+1} of order k and agrees with $R_k = [r_{j\ell}^{(k)}]_{j,\ell=1}^k$ except for the last diagonal entry. Equation (46) and the fact that β_k is nonvanishing yield

$$\bar{r}_{kk}^{(k)} > r_{kk}^{(k)} \ge 0, \tag{47}$$

and when T_k is nonsingular, we have $r_{kk}^{(k)} > 0$.

We turn to the computation of the columns of

$$\tilde{V}_{k+1} = [\tilde{\boldsymbol{v}}_1^{(k+1)}, \tilde{\boldsymbol{v}}_2^{(k+1)}, \ldots, \tilde{\boldsymbol{v}}_{k+1}^{(k+1)}] := V_{k+1}Q_{k+1} \tag{48}$$

from those of the matrix $\tilde{V}_k$, where V_{k+1} is determined by the Lanczos decomposition (41) and Q_{k+1} is given by (44). Substituting (44) into the right-hand side of (48) yields

$$\begin{aligned} \tilde{V}_{k+1} &= [V_k, \boldsymbol{v}_{k+1}]Q_{k+1} = [\tilde{V}_k, \boldsymbol{v}_{k+1}]G_{k+1}^{(k)T} \\ &= [\bar{V}_{k-1}, c_k\tilde{\boldsymbol{v}}_k^{(k)} + s_k\boldsymbol{v}_{k+1}, -s_k\tilde{\boldsymbol{v}}_k^{(k)} + c_k\boldsymbol{v}_{k+1}]. \end{aligned} \tag{49}$$

Thus, the first $k-1$ columns of the matrix $\tilde{V}_{k+1}$ are the columns of $\bar{V}_{k-1}$. The columns $\tilde{\boldsymbol{v}}_k^{(k+1)}$ and $\tilde{\boldsymbol{v}}_{k+1}^{(k+1)}$ of $\tilde{V}_{k+1}$ are linear combinations of the last columns of $\tilde{V}_k$ and V_{k+1}.

Assume that the solution $\boldsymbol{z}_{k-1}$ of the linear system (29) is available. Since the matrix $\bar{R}_k$ is upper triangular and $\bar{R}_{k-1}$ is the leading principal submatrix of order $k-1$ of $\bar{R}_k$, the computation of the solution $\boldsymbol{z}_k = [\zeta_1, \zeta_2, \ldots, \zeta_k]^T$ of

$$\bar{R}_k^T \boldsymbol{z}_k = \|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1 \tag{50}$$

is easy. We have

$$\boldsymbol{z}_k = \begin{bmatrix} \boldsymbol{z}_{k-1} \\ \zeta_k \end{bmatrix}, \qquad \zeta_k = -(\bar{r}_{k-2,k}^{(k)}\zeta_{k-2} + \bar{r}_{k-1,k}^{(k)}\zeta_{k-1})/\bar{r}_{kk}^{(k)}. \tag{51}$$

Hence, only the last column of the matrix $\bar{R}_k$ is required.

We are now in a position to compute $\boldsymbol{x}_{k+1}$ from $\boldsymbol{x}_k$. Equations (25) and (49) yield

$$\boldsymbol{x}_{k+1} = \bar{V}_k\boldsymbol{z}_k = \bar{V}_{k-1}\boldsymbol{z}_{k-1} + \zeta_k\tilde{\boldsymbol{v}}_k^{(k+1)} = \boldsymbol{x}_k + \zeta_k\tilde{\boldsymbol{v}}_k^{(k+1)},$$

where we have used that $\tilde{\boldsymbol{v}}_k^{(k+1)}$ is the last column of $\bar{V}_k$. Note that only the last few columns of V_k and $\tilde{V}_k$ have to be stored in order to update the approximate solution $\boldsymbol{x}_k$.

3. Quadrature rules of Gauss-type for error estimation. This section describes how to bound or compute estimates of the matrix functional (8) by approximating the Stieltjes integral representation (9) by quadrature rules of Gauss-type. A nice discussion on the application of Gauss quadrature rules to the evaluation of upper and lower bounds of certain matrix functionals is presented by Golub and Meurant [9]. Related discussions can also be found in [2, 4, 11].

3.1. Gauss quadrature rules. Let f be a $2k$ times continuously differentiable function defined on the interval $[\lambda_1, \lambda_n]$, which contains the support of the measure ω. The k-point Gauss quadrature rule associated with ω for the computation of an approximation of the integral (10) is given by

$$\mathcal{G}_k(f) := \sum_{j=1}^{k} f(\theta_j^{(k)})\omega_j^{(k)}, \qquad \omega_j^{(k)} := \|\boldsymbol{b}\|^2(\tilde{\boldsymbol{e}}_1^T W_k \tilde{\boldsymbol{e}}_j)^2, \tag{52}$$

where the $\theta_j^{(k)}$ and W_k are defined by (38). The nodes and weights of the Gauss rule are uniquely determined by the requirement

$$\mathcal{G}_k(p) = \mathcal{I}(p), \qquad \forall p \in \Pi_{2k-1}, \tag{53}$$

where $\mathcal{I}$ is defined by (10). We also will use the representation

$$\mathcal{G}_k(f) = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T f(T_k) \tilde{\boldsymbol{e}}_1. \tag{54}$$

The equivalence of (52) and (54) is shown in [9] and follows by substituting the spectral factorization (38) into (54). The integration error

$$\mathcal{E}_k(f) := \mathcal{I}(f) - \mathcal{G}_k(f)$$

can be expressed as

$$\mathcal{E}_k(f) = \frac{f^{(2k)}(\tilde{\theta}^{(k)})}{(2k)!} \int_{-\infty}^{\infty} \prod_{\ell=1}^{k} (t - \theta_\ell^{(k)})^2 d\omega(t), \tag{55}$$

for some $\tilde{\theta}^{(k)}$ in the interval $[\lambda_1, \lambda_n]$, where $f^{(2k)}$ denotes the derivative of order $2k$ of the function f; see, e.g., [9] or [18, Section 3.6] for details.

In the remainder of this section, we will assume that f is given by (8) and that the matrix A is positive definite. Then $f^{(2k)}(t) > 0$ for $t > 0$, and the constant $\tilde{\theta}^{(k)}$ in (55) is positive. It follows from (55) that $\mathcal{E}_k(f) > 0$, and therefore

$$\mathcal{G}_k(f) < \mathcal{I}(f) = F(A) = \boldsymbol{b}^T A^{-2} \boldsymbol{b}, \tag{56}$$

where $F(A)$ is defined by (8).

The representation (54) of the Gauss quadrature rule can be simplified by using the QR-factorization (18) of T_k when f is given by (8),

$$\mathcal{G}_k(f) = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T T_k^{-2} \tilde{\boldsymbol{e}}_1 = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T R_k^{-1} R_k^{-T} \tilde{\boldsymbol{e}}_1 = \|\boldsymbol{b}\|^2 \|R_k^{-T} \tilde{\boldsymbol{e}}_1\|^2. \tag{57}$$

It is easy to evaluate the right-hand side of (57) when the solution $\boldsymbol{z}_{k-1}$ of (29) is available. Let $\tilde{\boldsymbol{z}}_k \in \mathbb{R}^k$ satisfy

$$R_k^T \tilde{\boldsymbol{z}}_k = \|\boldsymbol{b}\| \tilde{\boldsymbol{e}}_1. \tag{58}$$

Then

$$\mathcal{G}_k(f) = \tilde{\boldsymbol{z}}_k^T \tilde{\boldsymbol{z}}_k. \tag{59}$$

Since all entries $r_{j\ell}^{(k)}$ of R_k and $\bar{r}_{j\ell}^{(k)}$ of $\bar{R}_k$ are the same, except for $r_{kk}^{(k)} \neq \bar{r}_{kk}^{(k)}$, the solution of (58) is given by

$$\tilde{\boldsymbol{z}}_k = \begin{bmatrix} \boldsymbol{z}_{k-1} \\ \tilde{\zeta}_k \end{bmatrix}, \qquad \tilde{\zeta}_k = -(\bar{r}_{k-2,k}^{(k)} \zeta_{k-2} + \bar{r}_{k-1,k}^{(k)} \zeta_{k-1}) / r_{kk}^{(k)}. \tag{60}$$

Substituting inequality (56) into (35) (with k replaced by $k+1$) and using the representation (59) yields

$$\boldsymbol{e}_{k+1}^T \boldsymbol{e}_{k+1} > \tilde{\boldsymbol{z}}_k^T \tilde{\boldsymbol{z}}_k - \boldsymbol{z}_k^T \boldsymbol{z}_k = \tilde{\zeta}_k^2 - \zeta_k^2, \tag{61}$$

where the equality follows from (51) and (60). A comparison of (51) and (60) yields, in view of the inequality (47), that $|\tilde{\zeta}_k| \geq |\zeta_k|$, and therefore the right-hand side of (61) is nonnegative. Moreover, if $\tilde{\zeta}_k \neq 0$, then $|\tilde{\zeta}_k| > |\zeta_k|$, and we obtain

$$\|\boldsymbol{e}_{k+1}\| > \sqrt{\tilde{\zeta}_k^2 - \zeta_k^2} > 0. \tag{62}$$

Thus, Gauss quadrature rules give easily computable lower bounds for the error in the approximate solutions generated by the iterative method when applied to linear systems of equations with a symmetric positive definite matrix.

3.2. Anti-Gauss quadrature rules. Let the matrix A be symmetric and positive definite. If the smallest eigenvalue λ_1 of A were explicitly known, then an upper bound of (56) could be computed by a $(k+1)$-point Gauss-Radau quadrature rule with a fixed node between λ_1 and the origin; see Golub and Meurant [9, 10] for details. The computed bound typically improves the further away from the origin we can allocate the fixed node. However, accurate lower bounds for λ_1 are, in general, not available. We therefore propose to use anti-Gauss quadrature rules to compute estimates of the error that generally are of opposite sign as $\mathcal{E}_k(f)$.

Anti-Gauss rules were introduced by Laurie [13], and their application to the evaluation of matrix functionals was explored in [4]. Let f be a smooth function. Analogously to the representation (54) of the k-point Gauss rule, the $(k+1)$-point anti-Gauss quadrature rule associated with ω for the computation of an approximation of the integral (10) is given by

$$\breve{\mathcal{G}}_{k+1}(f) := \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T f(\breve{T}_{k+1}) \tilde{\boldsymbol{e}}_1, \tag{63}$$

where

$$\breve{T}_{k+1} = \begin{bmatrix} \alpha_1 & \beta_1 & & & & & \mathbf{0} \\ \beta_1 & \alpha_2 & \beta_2 & & & & \\ & \beta_2 & \alpha_3 & & & & \\ & & & \ddots & & & \\ & & \ddots & \ddots & \beta_{k-1} & & \\ & & & \beta_{k-1} & \alpha_k & \sqrt{2}\beta_k & \\ \mathbf{0} & & & & \sqrt{2}\beta_k & \alpha_{k+1} \end{bmatrix} \in \mathbb{R}^{(k+1)\times(k+1)}. \tag{64}$$

Thus, $\breve{T}_{k+1}$ is obtained from T_{k+1} by multiplying the last off-diagonal entries by $\sqrt{2}$. We note that the determination of $\breve{T}_{k+1}$ requires application of $k+1$ steps of the Lanczos process; cf. (11).

The $(k+1)$-point anti-Gauss rule is characterized by the requirement that the integration error

$$\breve{\mathcal{E}}_{k+1}(f) := \mathcal{I}(f) - \breve{\mathcal{G}}_{k+1}(f)$$

satisfies

$$\breve{\mathcal{E}}_{k+1}(p) = -\mathcal{E}_k(p), \qquad \forall p \in \Pi_{2k+1},$$

which can be written in the equivalent form

$$\breve{\mathcal{G}}_{k+1}(p) = (2\mathcal{I} - \mathcal{G}_k)(p), \qquad \forall p \in \Pi_{2k+1}. \tag{65}$$

Assume for the moment that we can carry out n steps of the Lanczos process without break down. This yields an orthonormal basis $\{\boldsymbol{v}_j\}_{j=1}^n$ of $\mathbb{R}^n$ and an associated sequence of polynomials $\{s_j\}_{j=0}^{n-1}$ defined by (13) that satisfy (15). Expanding the function f on the spectrum of A, denoted by $\lambda(A)$, in terms of the polynomials s_j yields

$$f(t) = \sum_{j=0}^{n-1} \eta_j s_j(t), \qquad t \in \lambda(A), \tag{66}$$

where $\eta_j = (f, s_j)$, with the inner product defined by (14).

In view of $\mathcal{I}(s_j) = 0$ for $j > 0$ and (53), it follows from (66) that

$$\mathcal{I}(f) = \eta_0 \mathcal{I}(s_0) = \eta_0 \mathcal{G}_k(s_0). \tag{67}$$

Therefore, applying the Gauss rule $\mathcal{G}_k$ and anti-Gauss rule $\breve{\mathcal{G}}_{k+1}$ to (66), using (53), (65) and (67), yields for $n \geq 2k+2$ that

$$\mathcal{G}_k(f) = \mathcal{I}(f) + \sum_{j=2k}^{n-1} \eta_j \mathcal{G}_k(s_j), \tag{68}$$

$$\begin{aligned}
\breve{\mathcal{G}}_{k+1}(f) &= \sum_{j=0}^{n-1} \eta_j \breve{\mathcal{G}}_{k+1}(s_j) = \sum_{j=0}^{2k+1} \eta_j (2\mathcal{I} - \mathcal{G}_k)(s_j) + \sum_{j=2k+2}^{n-1} \eta_j \breve{\mathcal{G}}_{k+1}(s_j) \\
&= \sum_{j=0}^{2k+1} \eta_j 2\mathcal{I}(s_j) - \sum_{j=0}^{2k+1} \eta_j \mathcal{G}_k(s_j) + \sum_{j=2k+2}^{n-1} \eta_j \breve{\mathcal{G}}_{k+1}(s_j) \\
&= \mathcal{I}(f) - \eta_{2k}\mathcal{G}_k(s_{2k}) - \eta_{2k+1}\mathcal{G}_k(s_{2k+1}) + \sum_{j=2k+2}^{n-1} \eta_j \breve{\mathcal{G}}_{k+1}(s_j).
\end{aligned} \tag{69}$$

Assume that the coefficients η_j converge rapidly to zero with increasing index. Then the leading terms in the expansions (68) and (69) dominate the error, i.e.,

$$\begin{array}{rclcl}
\mathcal{E}_k(f) & = & (\mathcal{I} - \mathcal{G}_k)(f) & \approx & -\eta_{2k}\mathcal{G}_k(s_{2k}) - \eta_{2k+1}\mathcal{G}_k(s_{2k+1}), \\
\breve{\mathcal{E}}_{k+1}(f) & = & (\mathcal{I} - \breve{\mathcal{G}}_{k+1})(f) & \approx & \eta_{2k}\mathcal{G}_k(s_{2k}) + \eta_{2k+1}\mathcal{G}_k(s_{2k+1}),
\end{array} \tag{70}$$

where $\approx$ stands for "approximately equal to." This leads us to expect that, in general, the errors $\mathcal{E}_k(f)$ and $\breve{\mathcal{E}}_{k+1}(f)$ are of opposite sign and of roughly the same magnitude.

In the remainder of this subsection, we let f be defined by (8) and discuss the evaluation of anti-Gauss rules for this particular integrand. Introduce the QR-factorization

$$\breve{T}_{k+1} = \breve{Q}_{k+1}\breve{R}_{k+1}, \quad \breve{Q}_{k+1}, \breve{R}_{k+1} \in \mathbb{R}^{(k+1)\times(k+1)}, \quad \breve{Q}_{k+1}^T \breve{Q}_{k+1} = I_{k+1}, \tag{71}$$

where $\breve{R}_{k+1} = [\breve{r}_{j\ell}^{(k+1)}]_{j,\ell=1}^{k+1}$ is upper triangular. Using the representation (63), we obtain, analogously to (57), that

$$\breve{\mathcal{G}}_{k+1}(f) = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T \breve{T}_{k+1}^{-2} \tilde{\boldsymbol{e}}_1 = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T \breve{R}_{k+1}^{-1} \breve{R}_{k+1}^{-T} \tilde{\boldsymbol{e}}_1 = \|\boldsymbol{b}\|^2 \|\breve{R}_{k+1}^{-T} \tilde{\boldsymbol{e}}_1\|^2. \tag{72}$$

Since by (56) we have $\mathcal{E}_k(f) > 0$, equation (70) suggests that, typically, $\breve{\mathcal{E}}_{k+1}(f) < 0$. Thus, we expect that for many symmetric positive definite matrices A, right-hand side vectors $\boldsymbol{b}$ and values of k, the inequality

$$\breve{\mathcal{G}}_{k+1}(f) > \mathcal{I}(f) = F(A) = \boldsymbol{b}^T A^{-2} \boldsymbol{b} \tag{73}$$

holds, where f and F are given by (8).

Let $\breve{\boldsymbol{z}}_{k+1}$ satisfy

$$\breve{R}_{k+1}^T \breve{\boldsymbol{z}}_{k+1} = \|\boldsymbol{b}\| \tilde{\boldsymbol{e}}_1. \tag{74}$$

Then it follows from (72) that

$$\breve{\mathcal{G}}_{k+1}(f) = \breve{\boldsymbol{z}}_{k+1}^T \breve{\boldsymbol{z}}_{k+1}. \tag{75}$$

The matrix $\breve{R}_{k+1}$ can be determined when $k+1$ Lanczos steps have been completed, and so can the approximate solution $\boldsymbol{x}_{k+1}$ of (1). Substituting (73) into (35) (with k replaced by $k+1$) and using the representation (75) suggests that the inequality

$$\boldsymbol{e}_{k+1}^T \boldsymbol{e}_{k+1} < \breve{\boldsymbol{z}}_{k+1}^T \breve{\boldsymbol{z}}_{k+1} - \boldsymbol{z}_k^T \boldsymbol{z}_k \tag{76}$$

holds for many symmetric positive definite matrices A, right-hand side vectors $\boldsymbol{b}$ and values of k.

We evaluate the right-hand side of (76) by using the close relation between the upper triangular matrices $\breve{R}_{k+1}$ and $\bar{R}_k$. Assume that $\bar{R}_k$ is nonsingular and that $\beta_{k+1} \neq 0$. It is easy to see that the $k \times k$ leading principal submatrix of $\breve{R}_{k+1}$ agrees with $\bar{R}_k$ except for its last diagonal entry. A comparison of (74) with (29) (with $k-1$ replaced by k) shows that

$$\breve{\boldsymbol{z}}_{k+1} = \begin{bmatrix} \boldsymbol{z}_{k-1} \\ \breve{\zeta}_k^{(k+1)} \\ \breve{\zeta}_{k+1}^{(k+1)} \end{bmatrix},$$

where

$$\begin{aligned} \breve{\zeta}_k^{(k+1)} &= -(\bar{r}_{k-2,k}^{(k)} \zeta_{k-2} + \bar{r}_{k-1,k}^{(k)} \zeta_{k-1}) / \breve{r}_{kk}^{(k+1)}, \\ \breve{\zeta}_{k+1}^{(k+1)} &= -(\breve{r}_{k-1,k+1}^{(k+1)} \zeta_{k-1} + \breve{r}_{k,k+1}^{(k+1)} \breve{\zeta}_k^{(k+1)}) / \breve{r}_{k+1,k+1}^{(k+1)}, \end{aligned}$$

and the ζ_j are entries of $\boldsymbol{z}_{k-1}$. Thus,

$$\breve{\boldsymbol{z}}_{k+1}^T \breve{\boldsymbol{z}}_{k+1} - \boldsymbol{z}_k^T \boldsymbol{z}_k = (\breve{\zeta}_{k+1}^{(k+1)})^2 + (\breve{\zeta}_k^{(k+1)})^2 - \zeta_k^2.$$

Substitution of this identity into (76) yields

$$\|\boldsymbol{e}_{k+1}\| < \sqrt{(\breve{\zeta}_{k+1}^{(k+1)})^2 + (\breve{\zeta}_k^{(k+1)})^2 - \zeta_k^2}. \tag{77}$$

According to the above discussion, we expect the argument of the square-root to be positive and the inequality to hold for many symmetric positive definite matrices A, right-hand side vectors $\boldsymbol{b}$ and values of k. We refer to the right-hand side of (77) as an upper estimate of the norm of the error $\boldsymbol{e}_{k+1}$. However, we point out that the inequality (77) might be violated for some values of k. This is illustrated in the numerical examples of Section 4.

3.3. Gauss-Radau quadrature rules. Throughout this section we assume that the matrix A is nonsingular and indefinite. Thus, there is an index m such that the eigenvalues (7) of A satisfy

$$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_m < 0 < \lambda_{m+1} \leq \cdots \leq \lambda_n. \tag{78}$$

The application of Gauss quadrature rules (52) to estimate the norm of the error in approximate solutions $\boldsymbol{x}_k$ might not be possible for all values of k when A is indefinite, because for some $k > 0$ one of the nodes $\theta_j^{(k)}$ of the Gauss rule (52) may be at the origin, and the integrand f given by (8) is not defined there. In fact, numerical difficulties may arise also when one of the nodes $\theta_j^{(k)}$ is very close to the origin. We circumvent this problem by modifying the integrand and estimating the norm of the error in the computed approximate solutions by Gauss-Radau quadrature rules associated with the measure ω and with a fixed node at the origin. Note that since the matrix A is indefinite, the origin is inside the smallest interval containing the spectrum of A. Some of the desired Gauss-Radau rules therefore might not exist. We will return to this issue below.

Let f be a smooth function on a sufficiently large interval that contains $\lambda(A)$ in its interior. We may, for instance, think of f as analytic. The $(k+1)$-point Gauss-Radau quadrature rule associated with the measure ω and with a fixed node $\hat{\theta}_1$ at the origin for the integration of f is of the form

$$\hat{\mathcal{G}}_{k+1}(f) := \sum_{j=1}^{k+1} f(\hat{\theta}_j^{(k+1)})\hat{\omega}_j^{(k+1)}. \tag{79}$$

It is characterized by the requirements that

$$\hat{\mathcal{G}}_{k+1}(p) = \mathcal{I}(p), \qquad \forall p \in \Pi_{2k} \qquad \text{and} \qquad \hat{\theta}_1^{(k+1)} = 0.$$

The nodes and weights in (79) are given by formulas analogous to those for the nodes and weights of standard Gauss rules (52). Introduce the symmetric tridiagonal matrix

$$\hat{T}_{k+1} = \begin{bmatrix} \alpha_1 & \beta_1 & & & & & \Large 0 \\ \beta_1 & \alpha_2 & \beta_2 & & & & \\ & \beta_2 & \alpha_3 & & & & \\ & & & \ddots & & & \\ & & \ddots & \ddots & \beta_{k-1} & & \\ & & & \beta_{k-1} & \alpha_k & \beta_k & \\ \Large 0 & & & & \beta_k & \hat{\alpha}_{k+1} \end{bmatrix} \in \mathbb{R}^{(k+1)\times(k+1)}, \tag{80}$$

where

$$\hat{\alpha}_{k+1} := \beta_k^2 \tilde{\boldsymbol{e}}_k^T T_k^{-1} \tilde{\boldsymbol{e}}_k$$

and T_k is given by (37). In view of the discussion on the computation of β_k, see (42), all entries of the matrix $\hat{T}_{k+1}$ can be computed after k Lanczos steps have been completed, provided that the matrix T_k is invertible. Since A is indefinite, we cannot exclude that T_k is singular. However, because of the interlacing property of the eigenvalues of the matrices T_k and T_{k+1}, it follows that if T_k is singular, then T_{k+1} is not. Thus, the desired $(k+1)$-point Gauss-Radau rules can be determined for at least every other value of k.

Define the spectral factorization

$$\begin{aligned} \hat{T}_{k+1} &= \hat{W}_{k+1}\hat{\Theta}_{k+1}\hat{W}_{k+1}^T, \quad \hat{W}_{k+1} \in \mathbb{R}^{(k+1)\times(k+1)}, \quad \hat{W}_{k+1}^T\hat{W}_{k+1} = I_{k+1}, \\ \hat{\Theta}_{k+1} &= \mathrm{diag}[\hat{\theta}_1^{(k+1)}, \hat{\theta}_2^{(k+1)}, \ldots, \hat{\theta}_{k+1}^{(k+1)}], \quad 0 = \hat{\theta}_1^{(k+1)} < |\hat{\theta}_2^{(k+1)}| \le \ldots \le |\hat{\theta}_{k+1}^{(k+1)}|. \end{aligned}$$

The eigenvalues $\hat{\theta}_j^{(k+1)}$ are distinct and may be positive or negative. The nodes in the Gauss-Radau quadrature rule (79) are the eigenvalues $\hat{\theta}_j^{(k+1)}$ and the weights are given by

$$\hat{\omega}_j^{(k+1)} := \|\boldsymbol{b}\|^2 (\tilde{\boldsymbol{e}}_1^T \hat{W}_{k+1} \tilde{\boldsymbol{e}}_j)^2;$$

see Golub and Meurant [9] for details. Analogously to (54), the quadrature rule (79) also can be represented by

$$\hat{\mathcal{G}}_{k+1}(f) = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T f(\hat{T}_{k+1}) \tilde{\boldsymbol{e}}_1. \tag{81}$$

Let for the moment f be a function that is analytic on an interval that contains all eigenvalues of A and all Gauss-Radau nodes $\hat{\theta}_j^{(k+1)}$, and satisfies

$$f(t) := \begin{cases} 1/t^2, & t \in \lambda(A) \cup \{\hat{\theta}_j^{(k+1)}\}_{j=2}^{k+1}, \\ 0, & t = 0. \end{cases} \tag{82}$$

Then

$$\mathcal{I}(f) = \boldsymbol{b}^T A^{-2} \boldsymbol{b} = \boldsymbol{b}^T (A^\dagger)^2 \boldsymbol{b}$$

and the representations (79) and (81) yield

$$\hat{\mathcal{G}}_{k+1}(f) = \sum_{j=2}^{k+1} (\hat{\theta}_j^{(k+1)})^{-2} \hat{\omega}_j^{(k+1)} = \|\boldsymbol{b}\|^2 \tilde{\boldsymbol{e}}_1^T (\hat{T}_{k+1}^\dagger)^2 \tilde{\boldsymbol{e}}_1 = \|\boldsymbol{b}\|^2 \|\hat{T}_{k+1}^\dagger \tilde{\boldsymbol{e}}_1\|^2, \tag{83}$$

where $M^\dagger$ denotes the Moore-Penrose pseudoinverse of the matrix M.

PROPOSITION 3.1. *Let the index m be determined by (78). Then the nonvanishing eigenvalues $\hat{\theta}_j^{(k+1)}$ of the Gauss-Radau matrix $\hat{T}_{k+1}$ satisfy*

$$\hat{\theta}_j^{(k+1)} \leq \lambda_m \quad or \quad \hat{\theta}_j^{(k+1)} \geq \lambda_{m+1}, \qquad 2 \leq j \leq k+1.$$

Proof. The result follows by combining Lemma 5.2 and Lemma 5.3 of [3]. ◻

The proposition secures that none of the nonvanishing Gauss-Radau nodes is closer to the origin than the eigenvalue of A of smallest magnitude. This property does not hold for nodes in Gauss rules (52). Therefore, the symmetric tridiagonal matrices (37) associated with Gauss rules may be near-singular, even when A is well conditioned. Near-singularity of the tridiagonal matrices (37) makes the computed error estimates sensitive to propagated round-off errors, and may cause the computed estimates to be of poor quality. This is illustrated in Examples 3 and 4 of Section 4.

The error $(\mathcal{I} - \hat{\mathcal{G}}_{k+1})(f)$ can be expressed by a formula similar to (55). However, the derivatives of the integrand f change sign on the interval $[\lambda_1, \lambda_n]$ and the sign of the error cannot be determined from this formula. The Gauss-Radau rule only provides estimates of the error in the computed approximate solutions. The computed examples of Section 4 show these estimates to be close to the norm of the error in the computed approximate solutions. This is typical for our experience from a large number of computed examples.

We turn to the evaluation of Gauss-Radau rules (83). Define the QR-factorization

$$\hat{T}_{k+1} = Q_{k+1} \hat{R}_{k+1}, \quad Q_{k+1}, \hat{R}_{k+1} \in \mathbb{R}^{(k+1)\times(k+1)}, \quad Q_{k+1}^T Q_{k+1} = I_{k+1}, \tag{84}$$

where $\hat{R}_{k+1} = [\hat{r}_{j\ell}^{(k+1)}]_{j,\ell=1}^{k+1}$ is upper triangular. Since $\hat{T}_{k+1}$ is singular, the entry $\hat{r}_{k+1,k+1}^{(k+1)}$ vanishes. Note that the matrix Q_{k+1} in (84) is the same as in (43). Moreover, the leading $k \times k$ principal submatrix of $\hat{R}_{k+1}$ is given by the matrix $\bar{R}_k$ in (50).

Let $\boldsymbol{q}_{k+1}^{(k+1)}$ denote the last column of Q_{k+1}. Then

$$\boldsymbol{q}_{k+1}^{(k+1)T}\hat{T}_{k+1} = \boldsymbol{q}_{k+1}^{(k+1)T}Q_{k+1}\hat{R}_{k+1} = \tilde{\boldsymbol{e}}_{k+1}^T\hat{R}_{k+1} = \boldsymbol{0}^T.$$

By symmetry of $\hat{T}_{k+1}$ it follows that

$$\hat{T}_{k+1}\boldsymbol{q}_{k+1}^{(k+1)} = \boldsymbol{0},$$

i.e., $\boldsymbol{q}_{k+1}^{(k+1)}$ spans the null space of $\hat{T}_{k+1}$ and is orthogonal to the range of $\hat{T}_{k+1}$. In particular, $I_{k+1} - \boldsymbol{q}_{k+1}^{(k+1)}\boldsymbol{q}_{k+1}^{(k+1)T}$ is the orthogonal projector onto the range of $\hat{T}_{k+1}$.

We evaluate the right-hand side of (83) by using the QR-factorization (84) as follows. The vector $\|\boldsymbol{b}\|\hat{T}_{k+1}^{\dagger}\tilde{\boldsymbol{e}}_1$ is the solution of minimal norm of the least-squares problem

$$\min_{\boldsymbol{y}_{k+1}\in\mathbb{R}^{k+1}} \|\hat{T}_{k+1}\boldsymbol{y}_{k+1} - \|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1\|. \tag{85}$$

We may replace the vector $\|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1$ in (85) by its orthogonal projection onto the range of $\hat{T}_{k+1}$ without changing the solution of the least-squares problem. Thus, $\|\boldsymbol{b}\|\hat{T}_{k+1}^{\dagger}\tilde{\boldsymbol{e}}_1$ also is the solution of minimal norm of the least-squares problem

$$\min_{\boldsymbol{y}_{k+1}\in\mathbb{R}^{k+1}} \|\hat{T}_{k+1}\boldsymbol{y}_{k+1} - (I_{k+1} - \boldsymbol{q}_{k+1}^{(k+1)}\boldsymbol{q}_{k+1}^{(k+1)T})\|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1\|. \tag{86}$$

Substituting $\hat{T}_{k+1} = \hat{T}_{k+1}^T = \hat{R}_{k+1}^T Q_{k+1}^T$ into (86) and letting $\hat{\boldsymbol{y}}_{k+1} = Q_{k+1}^T\boldsymbol{y}_{k+1}$ yields the consistent linear system of equations

$$\hat{R}_{k+1}^T\hat{\boldsymbol{y}}_{k+1} = (I_{k+1} - \boldsymbol{q}_{k+1}^{(k+1)}\boldsymbol{q}_{k+1}^{(k+1)T})\|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1. \tag{87}$$

Let $\hat{\boldsymbol{y}}_{k+1}$ denote the minimal norm solution of (87). Then $\hat{\boldsymbol{y}}_{k+1} = \|\boldsymbol{b}\|Q_{k+1}^T\hat{T}_{k+1}^{\dagger}\tilde{\boldsymbol{e}}_1$ and therefore

$$\|\hat{\boldsymbol{y}}_{k+1}\| = \|\boldsymbol{b}\|\|\hat{T}_{k+1}^{\dagger}\tilde{\boldsymbol{e}}_1\|. \tag{88}$$

Since $\hat{r}_{k+1,k+1}^{(k+1)} = 0$ and $\hat{r}_{jj}^{(k+1)} > 0$ for $1 \le j \le k$, the minimal norm solution of (87) is of the form

$$\hat{\boldsymbol{y}}_{k+1} = \begin{bmatrix} \bar{\boldsymbol{y}}_k \\ 0 \end{bmatrix}, \qquad \bar{\boldsymbol{y}}_k \in \mathbb{R}^k.$$

The vector $\bar{\boldsymbol{y}}_k$ satisfies the linear system of equations obtained by removing the last row and column of the matrix and the last entry of the right-hand side in (87), i.e.,

$$\bar{R}_k\bar{\boldsymbol{y}}_k = \|\boldsymbol{b}\|\tilde{\boldsymbol{e}}_1 - \|\boldsymbol{b}\|\bar{\boldsymbol{q}}_k\boldsymbol{q}_{k+1}^{(k+1)T}\tilde{\boldsymbol{e}}_1,$$

where $\bar{\boldsymbol{q}}_k \in \mathbb{R}^k$ consists of the first k entries of $\boldsymbol{q}_{k+1}^{(k+1)}$. Thus,

$$\bar{\boldsymbol{y}}_k = \boldsymbol{z}_k + \bar{\boldsymbol{z}}_k, \tag{89}$$

where $\boldsymbol{z}_k$ solves (50) and $\bar{\boldsymbol{z}}_k$ satisfies

$$\bar{R}_k^T\bar{\boldsymbol{z}}_k = -\|\boldsymbol{b}\|\bar{\boldsymbol{q}}_k\boldsymbol{q}_{k+1}^{(k+1)T}\tilde{\boldsymbol{e}}_1. \tag{90}$$

A recursion formula for the vector $\boldsymbol{q}_{k+1}^{(k+1)}$ can be derived easily. It follows from the representation (44) of the matrix Q_{k+1} that

$$\boldsymbol{q}_{k+1}^{(k+1)} = Q_{k+1}\tilde{\boldsymbol{e}}_{k+1} = \begin{bmatrix} Q_k & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} G_{k+1}^{(k)T}\tilde{\boldsymbol{e}}_{k+1} = \begin{bmatrix} -s_k\boldsymbol{q}_k^{(k)} \\ c_k \end{bmatrix}, \tag{91}$$

where $\boldsymbol{q}_k^{(k)}$ denotes the last column of Q_k. Repeated application of equation (91) for increasing values of k makes it possible to compute the vectors $\boldsymbol{q}_2^{(2)}, \boldsymbol{q}_3^{(3)}, \dots, \boldsymbol{q}_{k+1}^{(k+1)}$ in about $k^2/2$ arithmetic floating-point operations.

The solutions of the linear systems (90) can be evaluated by a recursion formula based on (91) for increasing values of k as follows. Equation (91) yields that

$$\begin{aligned} \boldsymbol{q}_{k+1}^{(k+1)T}\tilde{\boldsymbol{e}}_1 &= -s_k\boldsymbol{q}_k^{(k)T}\tilde{\boldsymbol{e}}_1, \\ \bar{\boldsymbol{q}}_k &= -s_k\boldsymbol{q}_k^{(k)} \end{aligned} \tag{92}$$

and

$$\bar{\boldsymbol{q}}_{k+1} = -s_{k+1}\begin{bmatrix} \bar{\boldsymbol{q}}_k \\ c_k \end{bmatrix}, \tag{93}$$

where the vector $\bar{\boldsymbol{q}}_{k+1}$ consists of the $k+1$ first entries of $\boldsymbol{q}_{k+2}^{(k+2)}$, the last column of Q_{k+2}. Assume that the solution $\bar{\boldsymbol{z}}_k$ of (90) is available. We would like to compute the vector $\bar{\boldsymbol{z}}_{k+1} = [\bar{\zeta}_1^{(k+1)}, \bar{\zeta}_2^{(k+1)}, \dots, \bar{\zeta}_{k+1}^{(k+1)}]^T$ that satisfies

$$\bar{R}_{k+1}^T\bar{\boldsymbol{z}}_{k+1} = -\|\boldsymbol{b}\|\bar{\boldsymbol{q}}_{k+1}\boldsymbol{q}_{k+2}^{(k+2)T}\tilde{\boldsymbol{e}}_1. \tag{94}$$

Substituting (92) and (93) into (94) yields

$$\bar{R}_{k+1}^T\bar{\boldsymbol{z}}_{k+1} = -\|\boldsymbol{b}\|s_{k+1}^2\begin{bmatrix} \bar{\boldsymbol{q}}_k \\ c_k \end{bmatrix}\boldsymbol{q}_{k+1}^{(k+1)T}\tilde{\boldsymbol{e}}_1,$$

which shows that

$$\begin{aligned} \bar{\boldsymbol{z}}_{k+1} &= \begin{bmatrix} s_{k+1}^2\bar{\boldsymbol{z}}_k \\ \bar{\zeta}_{k+1}^{(k+1)} \end{bmatrix}, \\ \bar{\zeta}_{k+1}^{(k+1)} &= -(\|\boldsymbol{b}\|s_{k+1}^2c_k\boldsymbol{q}_{k+1}^{(k+1)T}\tilde{\boldsymbol{e}}_1 + \bar{r}_{k-1,k+1}^{(k+1)}\bar{\zeta}_{k-1}^{(k+1)} + \bar{r}_{k,k+1}^{(k+1)}\bar{\zeta}_k^{(k+1)})/\bar{r}_{k+1,k+1}^{(k+1)}. \end{aligned}$$

Thus, assuming that the matrix $\bar{R}_{k+1}$ is available, all the vectors $\bar{\boldsymbol{z}}_1, \bar{\boldsymbol{z}}_2, \dots, \bar{\boldsymbol{z}}_{k+1}$ can be computed in $O(k^2)$ arithmetic floating-point operations.

Having computed the solutions of (50) and (90), the above development, and in particular equations (88) and (89), show that we can evaluate the $(k+1)$-point Gauss-Radau rule (83) with integrand (82) according to

$$\hat{\mathcal{G}}_{k+1}(f) = \|\boldsymbol{z}_k + \bar{\boldsymbol{z}}_k\|^2.$$

Substituting this approximation of $\boldsymbol{b}^TA^{-2}\boldsymbol{b}$ into (35) yields

$$\begin{aligned} \boldsymbol{e}_k^T\boldsymbol{e}_k &= |\boldsymbol{b}^TA^{-2}\boldsymbol{b} - \boldsymbol{z}_{k-1}^T\boldsymbol{z}_{k-1}| \\ &\approx |\|\boldsymbol{z}_k + \bar{\boldsymbol{z}}_k\|^2 - \boldsymbol{z}_{k-1}^T\boldsymbol{z}_{k-1}| \\ &= |\bar{\boldsymbol{z}}_k^T(2\boldsymbol{z}_k + \bar{\boldsymbol{z}}_k) + \zeta_k^2|, \end{aligned}$$

where the last equality follows from (51). This suggests the approximation

$$\|\boldsymbol{e}_k\| \approx |\bar{\boldsymbol{z}}_k^T(2\boldsymbol{z}_k + \bar{\boldsymbol{z}}_k) + \zeta_k^2|^{1/2}. \tag{95}$$

We note that the approximate solution $\boldsymbol{x}_k$ of (1) and the right-hand side of (95) can be evaluated after k Lanczos steps have been carried out and the last subdiagonal entry of the Gauss-Radau matrix (80) has been determined by (42). Computed examples in the following section indicate that the approximation (95) typically gives accurate estimates of the norm of the error.

4. Computed examples. We describe four examples that illustrate the performance of the iterative method, the error bound and the error estimates. All computations were carried out on an XP1000 Alpha workstation in Matlab with about 16 significant digits. In all examples we chose the initial approximate solution $\boldsymbol{x}_0 = \boldsymbol{0}$ and terminated the iterations as soon as

$$\|\boldsymbol{e}_k\| < \epsilon, \tag{96}$$

with $\epsilon := 1 \cdot 10^{-10}$ or $\epsilon := 1 \cdot 10^{-11}$. These values of ϵ are likely to be smaller than values of interest in many application. Our choices of ϵ demonstrates the possibility of computing accurate solutions and error estimates. In fact, the error bounds and estimates perform well also for values of ϵ smaller than $1 \cdot 10^{-11}$.

We determined the matrices in the linear systems in Examples 1–3 in the following fashion. Let

$$A := U_n\Lambda_nU_n^T, \quad \Lambda_n = \mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_n], \quad U_n \in \mathbb{R}^{n\times n}, \quad U_n^TU_n = I_n, \tag{97}$$

where the eigenvector matrix U_n either is the $n \times n$ identity matrix I_n or a random orthogonal matrix determined by orthogonalizing the columns of an $n \times n$ real matrix with random entries. The matrix A is diagonal when $U_n = I_n$ and dense when U_n is a random orthogonal matrix. We remark that the matrices T_k and V_k in the Lanczos decomposition (11) depend on the choice of U_n. Moreover, propagated round-off errors, due to round-offs introduced during matrix-vector product evaluations with the matrix A, may depend on the matrix U_n.

Example 1. Let $n := 1000$ and assume that the diagonal entries of the matrix Λ_n in (97) are given by $\lambda_j = 5j$. We first let U_n be a random orthogonal matrix. Then the matrix A defined by (97) is symmetric positive definite and dense. The right-hand side vector $\boldsymbol{b}$ is chosen so that $\boldsymbol{x} = \frac{1}{10}[1, 1, \ldots, 1]^T$ solves (1). We terminate the iterations as soon as (96) is satisfied with $\epsilon = 1 \cdot 10^{-11}$.

Figure 1(a) shows the 10-logarithm of $\|\boldsymbol{e}_k\|$ (solid curve), the 10-logarithm of the lower bound of $\|\boldsymbol{e}_k\|$ computed by Gauss quadrature (62) (dash-dotted curve), and the 10-logarithm of the upper estimate of $\|\boldsymbol{e}_k\|$ computed by anti-Gauss quadrature (77) (dashed curve) as functions of the number of iterations k. After the first 50 iterations, the computed lower bounds and upper estimates can be seen to be quite close to the norm of the error in the computed approximate solutions.

The closeness between the lower bound (62), upper estimate (77), and the norm of the error of the computed approximate solutions is also illustrated in Figures 1(b) and 1(c). The former figure displays $(\tilde{\zeta}_k^2 - \zeta_k^2)^{1/2} - \|\boldsymbol{e}_k\|$ (solid curve) and $(((\breve{\zeta}_k^{(k)})^2 + (\breve{\zeta}_{k-1}^{(k)})^2 - \zeta_{k-1}^2)^{1/2} - \|\boldsymbol{e}_k\|$ (dash-dotted curve) as functions of k. These quantities are seen to converge to zero as k increases. To shed some light on the rate of convergence, Figure 1(c) shows the relative differences $((\tilde{\zeta}_k^2 - \zeta_k^2)^{1/2} - \|\boldsymbol{e}_k\|)/\|\boldsymbol{e}_k\|$ and $((((\breve{\zeta}_k^{(k)})^2 + (\breve{\zeta}_{k-1}^{(k)})^2 - \zeta_{k-1}^2)^{1/2} - \|\boldsymbol{e}_k\|)/\|\boldsymbol{e}_k\|$, both of which converge to zero as k increases.

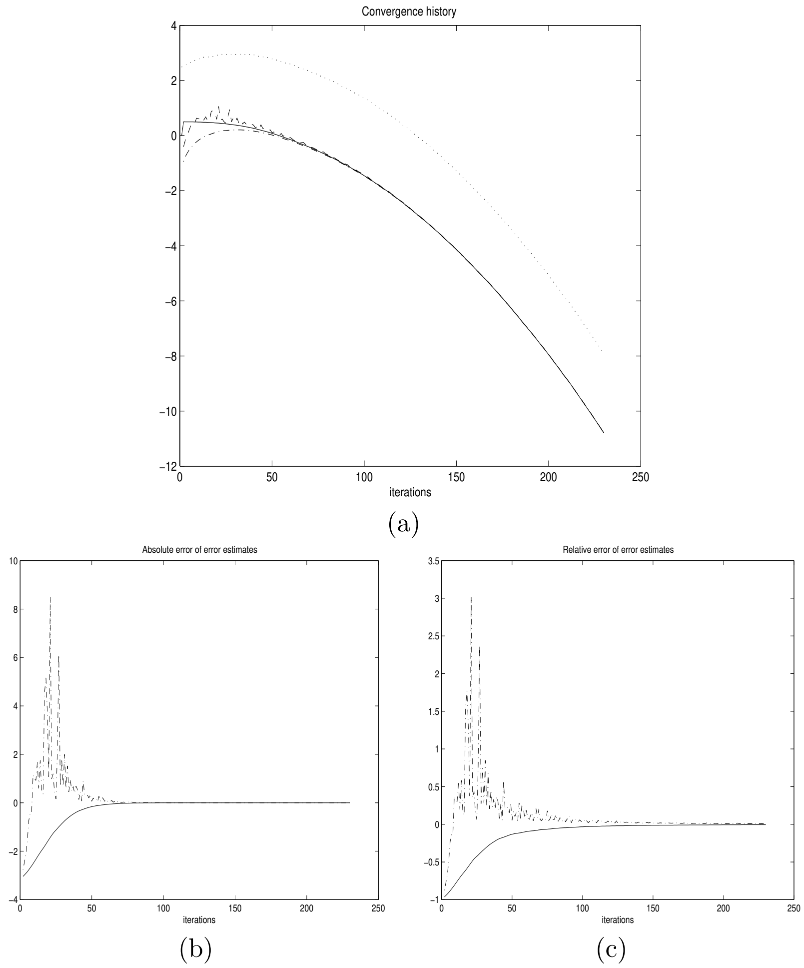


FIG. 1. *Example 1: Symmetric positive definite dense matrix. Figure (a) shows the* 10*-logarithm of the norm of the error (solid curve), of the Gauss bound (62) (dash-dotted curve), of the anti-Gauss upper estimate (77) (dashed curve) and of the norm of the residual error (dotted curve). Figure (b) displays the error in the Gauss bound (solid curve) and anti-Gauss upper estimate (dash-dotted curve). Figure (c) shows the relative error in the Gauss bound (solid curve) and anti-Gauss upper estimate (dash-dotted curve).*

Figure 1(a) also shows the 10-logarithm of the norm of the residual error (3) as a function of k (dotted curve). The norm of the residual error is about a factor $1 \cdot 10^3$ larger than the norm of the error in the corresponding approximate solution. If we would like to stop the iterations when the error in the computed approximate solution is below a certain tolerance, then we can terminate the computations much sooner if we base the stopping criterion on the formulas (62) and (77) than on the norm of the residual error.

We now replace the random orthogonal matrix U_n in the definition (97) of the matrix A by I_n. The matrix A obtained is diagonal and has the same spectrum as the matrix used for the computations shown in Figure 1. The right-hand side vector $\boldsymbol{b}$ is chosen so that $\boldsymbol{x} = \frac{1}{10}[1, 1, \ldots, 1]^T$ solves (1). The performance of the iterative method applied to this linear system is displayed by Figure 2, which is analogous to

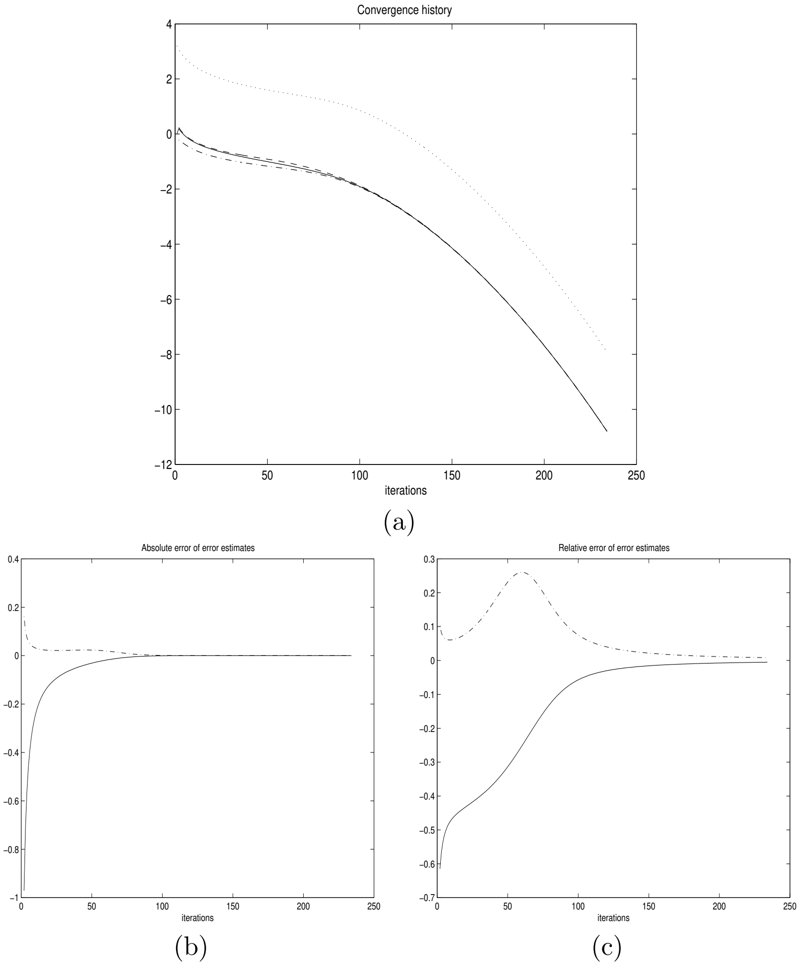


FIG. 2. *Example 1: Symmetric positive definite diagonal matrix. Figure (a) shows the* 10-*logarithm of the norm of the error (solid curve), of the Gauss bound (62) (dash-dotted curve), of the anti-Gauss upper estimate (77) (dashed curve) and of the norm of the residual error (dotted curve). Figure (b) displays the error in the Gauss bound (solid curve) and anti-Gauss upper estimate (dash-dotted curve). Figure (c) shows the relative error in the Gauss bound (solid curve) and anti-Gauss upper estimate (dash-dotted curve).*

Figure 1.

Figures 1 and 2 show the Gauss and anti-Gauss rules to give good lower bounds and upper estimates of the norm of the error in the computed approximate solutions, with the lower bounds and upper estimates being closer to the norm of the error when $U_n = I_n$ than when U_n was chosen to be a random orthogonal matrix. This example illustrates that the quality of the computed error bounds and estimates may depend on the eigenvector matrix of A. $\Box$

Example 2. Let the matrix $A \in \mathbb{R}^{48\times 48}$ in the linear system (1) be of the form (97) with U_{48} a random orthogonal matrix and Λ_{48} defined by

$$\lambda_i := c + \frac{i-1}{47}(d-c)\rho^{48-i}, \quad i = 1, 2, \ldots, 48.$$

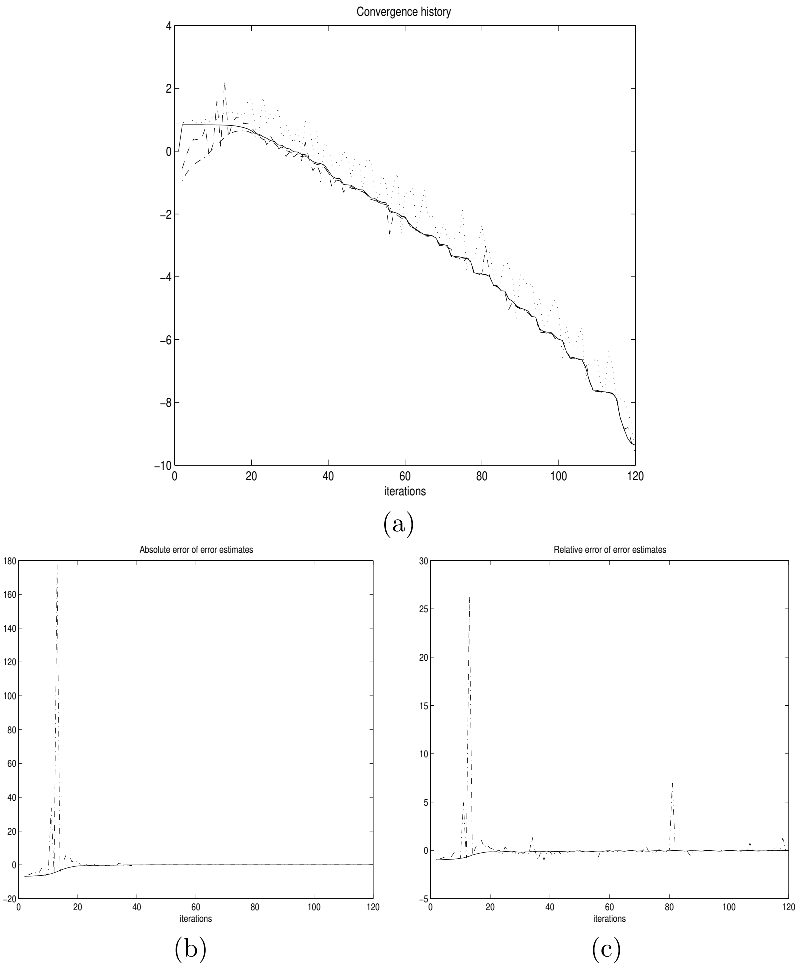
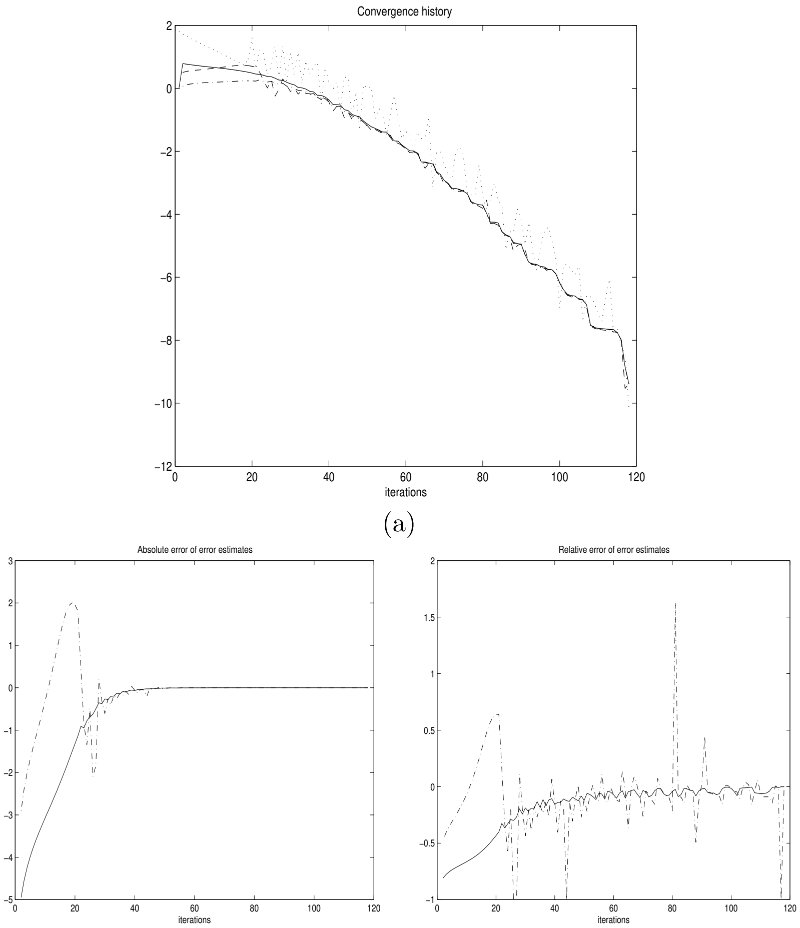


FIG. 3. *Example 2: Symmetric positive definite dense matrix. Figure (a) shows the* 10*-logarithm of the norm of the error (solid curve), of the Gauss bound (62) (dash-dotted curve), of the anti-Gauss upper estimate (77) (dashed curve) and of the norm of the residual error (dotted curve). Figure (b) displays the error in the Gauss bound (solid curve) and anti-Gauss upper estimate (dash-dotted curve). Figure (c) shows the relative error in the Gauss bound (solid curve) and anti-Gauss upper estimate (dash-dotted curve).*

Here $c := 0.1$, $d := 100$ and $\rho := 0.875$. Thus, A is symmetric, positive definite and dense. The right-hand side vector $\boldsymbol{b}$ is chosen so that $\boldsymbol{x} = [1, 1, \ldots, 1]^T$ solves the linear system (1). We terminate the iterations as soon as (96) is satisfied with $\epsilon = 1 \cdot 10^{-10}$.

Figure 3 is analogous to Figure 1 and shows the performance of the iterative method, of the lower error bound (62) and of the upper error estimate (77). The error bound (62) and error estimate (77) are seen to be close to the norm of the error in the computed approximate solutions. The "spikes" in Figures 3(b) and 3(c) correspond to anti-Gauss rules associated with ill-conditioned tridiagonal matrices (64). Ill-conditioning of the tridiagonal matrices (64) can cause loss of accuracy in the computed error estimates.

Now replace the random orthogonal matrix U_{48} in the definition (97) of the matrix



(a)

(b) (c)

FIG. 4. *Example 2: Symmetric positive definite diagonal matrix. Figure (a) shows the* 10-*logarithm of the norm of the error (solid curve), of the Gauss bound (62) (dash-dotted curve), of the anti-Gauss upper estimate (77) (dashed curve) and of the norm of the residual error (dotted curve). Figure (b) displays the error in the Gauss bound (solid curve) and anti-Gauss upper estimate (dash-dotted curve). Figure (c) shows the relative error in the Gauss bound (solid curve) and anti-Gauss upper estimate (dash-dotted curve).*

A by the identity matrix I_{48}. The matrix A so defined is diagonal and has the same spectrum as the matrix used for the computations shown in Figure 3. The right-hand side vector $\boldsymbol{b}$ is chosen so that $\boldsymbol{x} = [1, 1, \ldots, 1]^T$ solves (1). This linear system has previously been used in computed examples in [10, 11, 14] with a stopping criterion, based on the A-norm instead of the Euclidean norm, with $\epsilon = 1 \cdot 10^{-10}$. We therefore use the same value of ϵ in the present example. The performance of the iterative method, as well as of the error bounds and estimates, are shown in Figure 4.

Figures 3 and 4 display that the lower bounds and upper estimates of the norm of the error in the computed approximate solutions are closer to the norm of the error when $U_{48} = I_{48}$ than when U_{48} was chosen to be a random orthogonal matrix. Thus, similarly as in Example 1, the quality of the error bounds and estimates depends on the eigenvector matrix of A. $\square$

The following two examples are concerned with linear systems of equations with symmetric indefinite matrices. For such matrices, the convex hull of the spectrum contains the origin, and some Gauss rules (52) may have a node in the interval between the largest negative and the smallest positive eigenvalues, where the matrix has no eigenvalues. The presence of a node close to the origin can give inaccurate estimates of the norm of the error in the computed approximate solution. This is illustrated by Figures 5 and 6. This difficulty is circumvented by Gauss-Radau quadrature rules, cf. Proposition 3.1.

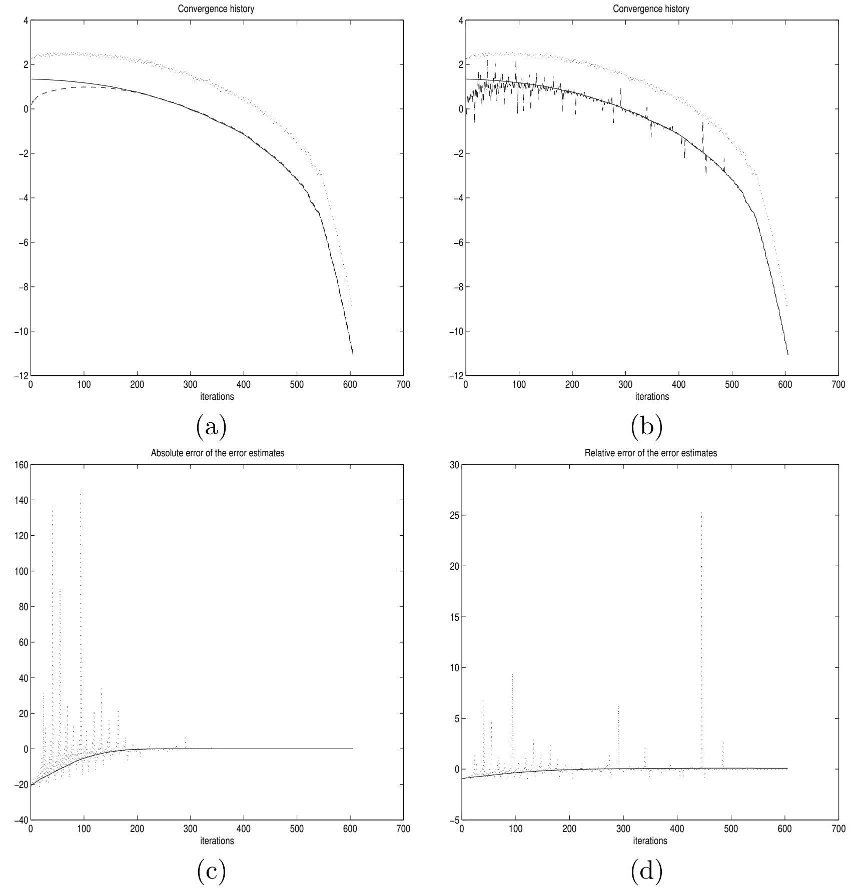


FIG. 5. *Example 3: Symmetric indefinite dense matrix. Figure (a) shows the* 10*-logarithm of the norm of the error (solid curve), of the Gauss-Radau estimate (95) (dashed curve) and of the norm of the residual error (dotted curve). Figure (b) displays the* 10*-logarithm of the norm of the error (solid curve), of the Gauss estimate (62) (dashed curve) and of the norm of the residual error (dotted curve). Figure (c) shows the error in the Gauss-Radau estimate (solid curve) and Gauss estimate (dotted curve). Figure (d) displays the relative error in the Gauss-Radau estimate (solid curve) and Gauss estimate (dotted curve).*

Example 3. Let the matrix A in (1) be of order 491 and of the form (97), where U_{491} is a random orthogonal matrix and the entries of the diagonal matrix Λ_{491} are given by

$$\lambda_i = \begin{cases} -150 + (i-1), & i = 1, \dots, 141, \\ i - 141, & i = 142, \dots, 491. \end{cases}$$

Then A is a dense matrix with eigenvalues in the interval $[-150, 350]$. We determine the right-hand side vector $\boldsymbol{b}$ so that $\boldsymbol{x} = [1, 1, \ldots, 1]^T$ solves the linear system (1). The iterations are terminated as soon as (96) is satisfied with $\epsilon = 1 \cdot 10^{-11}$.

Figure 5(a) shows the 10-logarithm of the error in the computed approximate solutions (solid curve), the 10-logarithm of the error estimate determined by Gauss-Radau quadrature (95) (dashed curve), and the 10-logarithm of the norm of the residual error (dotted curve). The error estimates computed by Gauss-Radau quadrature can be seen to be quite close to the norm of the error in the computed approximate solutions.

Figure 5(b) is obtained from Figure 5(a) by replacing the curve for the Gauss-Radau estimates (95) with a curve that displays error estimates computed by Gauss quadrature (62). Thus, the dashed curve of Figure 5(b) displays the 10-logarithm of the right-hand side of (62). Note that since A is indefinite, formula (55) for the integration error does not reveal the sign of the error and inequality (56) is not guaranteed to hold. The Gauss rules only give estimates of the norm of the error in the computed approximate solutions. The "spikes" of the dashed curve are caused by nodes of Gauss rules being very close to the origin.

Figure 5(c) is analogous to Figure 3(b). The solid curve displays the error in the Gauss-Radau estimates $|\bar{\boldsymbol{z}}_k^T(2\boldsymbol{z}_k + \bar{\boldsymbol{z}}_k) + \zeta_k^2|^{1/2} - \|\boldsymbol{e}_k\|$, cf. (95), and the dashed curve shows the error in the Gauss estimates $(\tilde{\zeta}_k^2 - \zeta_k^2)^{1/2} - \|\boldsymbol{e}_k\|$. Figure 5(d) displays the corresponding relative errors, i.e., $(|\bar{\boldsymbol{z}}_k^T(2\boldsymbol{z}_k + \bar{\boldsymbol{z}}_k) + \zeta_k^2|^{1/2} - \|\boldsymbol{e}_k\|)/\|\boldsymbol{e}_k\|$ (solid curve) and $((\tilde{\zeta}_k^2 - \zeta_k^2)^{1/2} - \|\boldsymbol{e}_k\|)/\|\boldsymbol{e}_k\|$ (dashed curve). The Gauss-Radau estimates are seen to be more reliable than the Gauss estimates. $\Box$

Example 4. Let $A \in \mathbb{R}^{200\times 200}$ be defined by $A := B^2 - \mu I_{200}$, where B is the standard 3-point discretization of the 1-dimensional Laplacian and $\mu := \sqrt{3}$. Thus, B^2 is a pentadiagonal matrix; a typical row has the nonvanishing entries $\{1, -4, 6, -4, 1\}$. Then A has 77 negative eigenvalues and condition number 3.9×10^3. The right-hand side vector $\boldsymbol{b}$ is chosen so that $\boldsymbol{x} = [1, 1, \ldots, 1]^T$ solves the linear system (1). We terminate the iterations as soon as the stopping criterion (96) is satisfied with $\epsilon = 1 \cdot 10^{-11}$.

Figures 6(a)-(d) are analogous to Figures 5(a)-(d). The error estimates obtained by Gauss-Radau rules are quite accurate, while the estimates determined by Gauss rules oscillate widely during the first 77 iterations. After these initial iterations both Gauss-Radau and Gauss rules provide accurate error estimates. $\Box$

5. Conclusion. This paper describes an iterative method for the solution of linear systems of equations with a symmetric nonsingular matrix. The iterative method is designed to allow the computation of bounds or estimates of the error in the computed approximate solutions. Computed examples show that the computed bounds and estimates are close to the norm of the actual errors in the computed approximate solutions.

Acknowledgements. We would like to thank Walter Gautschi for comments. This work was concluded while D.C. and L.R. were visiting the University of Bologna. They would like to thank Fiorella Sgallari for making this visit possible and enjoyable.

REFERENCES

[1] C. Brezinski, Error estimates for the solution of linear systems, SIAM J. Sci. Comput., 21 (2000), pp. 764–781.
[2] D. Calvetti, G. H. Golub and L. Reichel, A computable error bound for matrix functionals, J. Comput. Appl. Math., 103 (1999), pp. 301–306.

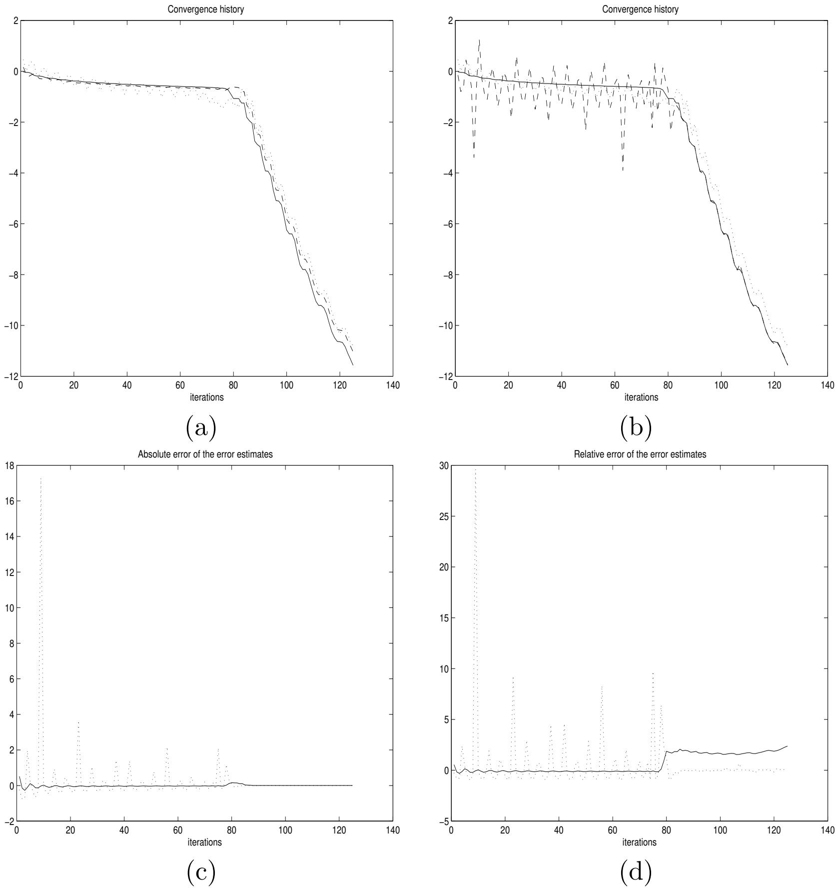


FIG. 6. *Example 4: Symmetric indefinite banded matrix. Figure (a) shows the* 10*-logarithm of the norm of the error (solid curve), of the Gauss-Radau estimate (95) (dashed curve) and of the norm of the residual error (dotted curve). Figure (b) displays the* 10*-logarithm of the norm of the error (solid curve), of the Gauss estimate (62) (dashed curve) and of the norm of the residual error (dotted curve). Figure (c) shows the error in the Gauss-Radau estimate (solid curve) and Gauss estimate (dotted curve). Figure (d) displays the relative error in the Gauss-Radau estimate (solid curve) and Gauss estimate (dotted curve).*

[3] D. Calvetti and L. Reichel, An adaptive Richardson iteration method for indefinite linear systems, Numer. Algorithms, 12 (1996), pp. 125–149.

[4] D. Calvetti, L. Reichel and F. Sgallari, Application 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.

[5] D. Calvetti, L. Reichel and D. C. Sorensen, An implicitly restarted Lanczos method for large symmetric eigenvalue problems, Elec. Trans. Numer. Anal., 2 (1994), pp. 1–21.

[6] G. Dahlquist, S. C. Eisenstat and G. H. Golub, Bounds for the error of linear systems of equations using the theory of moments, J. Math. Anal. Appl., 37 (1972), pp. 151–166.

[7] G. Dahlquist, G. H. Golub and S. G. Nash, Bounds for the error in linear systems, in Semi-Infinite Programming, ed. R. Hettich, Lecture Notes in Control and Computer Science # 15, Springer, Berlin, 1979, pp. 154–172.

[8] B. Fischer, Polynomial Based Iteration Methods for Symmetric Linear Systems, Teubner-Wiley, New York, 1996.

[9] 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.

[10] G. H. Golub and G. Meurant, Matrices, moments and quadrature II: How to compute the norm

of the error in iterative methods, BIT, 37 (1997), pp. 687–705.
[11] G. H. Golub and Z. Strakos, Estimates in quadratic formulas, Numer. Algorithms, 8 (1994), pp. 241–268.
[12] G. H. Golub and C. F. Van Loan, Matrix Computations, 3rd ed., Johns Hopkins University Press, Baltimore, 1996.
[13] D. P. Laurie, Anti-Gaussian quadrature formulas, Math. Comp., 65 (1996), pp. 739–747.
[14] G. Meurant, The computation of bounds for the norm of the error in the conjugate gradient algorithm, Numer. Algorithms, 16 (1997), pp. 77–87.
[15] G. Meurant, Numerical experiments in computing bounds for the norm of the error in the preconditioned conjugate gradient algorithm, Numer. Algorithms, to appear.
[16] C. C. Paige and M. A. Saunders, Solution of sparse indefinite systems of linear equations, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.
[17] Y. Saad, Iterative Methods for Sparse Linear Systems, PWS, Boston, 1996.
[18] J. Stoer and R. Bulirsch, Introduction to Numerical Analysis, 2nd ed., Springer, New York, 1993.