Electronic Transactions on Numerical Analysis.
Volume 13, pp. 119-147, 2002.

ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

THE INTERPLAY BETWEEN CLASSICAL ANALYSIS AND (NUMERICAL) LINEAR ALGEBRA — A TRIBUTE TO GENE H. GOLUB*

WALTER GAUTSCHI†

Dedicated in friendship, and with high esteem, to Gene H. Golub on his 70th birthday

Abstract. Much of the work of Golub and his collaborators uses techniques of linear algebra to deal with problems in analysis, or employs tools from analysis to solve problems arising in linear algebra. Instances are described of such interdisciplinary work, taken from quadrature theory, orthogonal polynomials, and least squares problems on the one hand, and error analysis for linear algebraic systems, element-wise bounds for the inverse of matrices, and eigenvalue estimates on the other hand.

Key words. Gauss-type quadratures, eigenvalue/vector characterizations, orthogonal polynomials, modification algorithms, polynomials orthogonal on several intervals, least squares problem, Lanczos algorithm, bounds for matrix functionals, iterative methods.

AMS subject classifications. 65D32, 33C45, 65D10, 15A45, 65F10.

1. Introduction.

It has been a privilege for me to have known Gene Golub for so many years and to have been able to see his very extensive work unfold. What intrigues me most about his work — at least the part I am familiar with — is the imaginative use made of linear algebra in problems originating elsewhere. Much of Golub's work, indeed, can be thought of as lying on the interface between classical analysis and linear algebra. The interface, to be sure, is directional: a problem posed in analysis may be solved with the help of linear algebra, or else, a linear algebra problem solved with tools from analysis. Instances of the former type occur in quadrature problems, orthogonal polynomials, and least squares problems, while examples of the latter type arise in error estimates for the solution of linear algebraic systems, element-wise bounds for the inverse of a matrix, and in eigenvalue estimates of interest in iterative methods.

It will not be possible here to pursue all the ramifications of this interesting interplay between different disciplines, but we try to bring across some of the main ideas and will refer to the literature for variations and extensions.

2. Quadrature.

Integration with respect to some given measure $\mathrm{d}\lambda$ on the real line $\mathbb{R}$ is certainly a topic that belongs to analysis, and so is the evaluation or approximation of integrals $\int_{\mathbb{R}} f(t)\mathrm{d}\lambda(t)$. If one follows Gauss, one is led to orthogonal polynomials relative to the measure $\mathrm{d}\lambda$, which is another vast area of classical analysis. How does linear algebra enter in all of this? It was in 1969 when the connection between Gauss quadrature rules and the algebraic eigenvalue problem was, if not discovered, then certainly exploited in the now classical and widely cited paper [33]. We begin with giving a brief account of this work, and then discuss various extensions thereof made subsequently.

*Received August 14, 2002. Accepted for publication August 20, 2002. Recommended by L. Reichel. Expanded version of a lecture presented in a special session honoring Professor Gene H. Golub at the Latsis Symposium 2002 on *Iterative Solvers for Large Linear Systems, celebrating 50 years of the conjugate gradient method*, held at the Swiss Federal Institute of Technology in Zurich, February 18–21, 2002.

†Department of Computer Sciences, Purdue University, West Lafayette, Indiana 47907-1398. E-mail: wxg@cs.purdue.edu

ETNA
Kent State University
etna@mcs.kent.edu

2.1. Gauss quadrature. Assume $\mathrm{d}\lambda$ is a positive measure on $\mathbb{R}$, all (or sufficiently many) of whose moments

$$\mu_r = \int_{\mathbb{R}} t^r \mathrm{d}\lambda(t), \quad r = 0, 1, 2, \dots, \tag{2.1}$$

exist with $\mu_0 > 0$. The n-point *Gauss quadrature rule* for $\mathrm{d}\lambda$ is

$$\int_{\mathbb{R}} f(t)\mathrm{d}\lambda(t) = \sum_{\nu=1}^{n} \lambda_\nu f(\tau_\nu) + R_n(f), \tag{2.2}$$

where $\lambda_\nu = \lambda_\nu^{(n)}$, $\tau_\nu = \tau_\nu^{(n)}$ depend on n and $\mathrm{d}\lambda$, and $R_n(f) = 0$ whenever f is a polynomial of degree $\leq 2n-1$,

$$R_n(f) = 0, \quad f \in \mathbb{P}_{2n-1}. \tag{2.3}$$

This is the maximum degree possible. If $f^{(2n)}$ is continuous on the support of $\mathrm{d}\lambda$ and has constant sign, then

$$R_n(f) > 0 \quad \text{if } \operatorname{sgn} f^{(2n)} = 1, \tag{2.4}$$

with the inequality reversed if sgn $f^{(2n)} = -1$.

The connection between Gauss quadrature and orthogonal polynomials is well known. If $\pi_k(\,\cdot\,) = \pi_k(\,\cdot\,; \mathrm{d}\lambda)$, $k = 0, 1, 2, \dots$, denotes the system of (monic) polynomials orthogonal with respect to the measure $\mathrm{d}\lambda$,

$$\int_{\mathbb{R}} \pi_k(t)\pi_\ell(t)\mathrm{d}\lambda(t) \begin{cases} = 0 & \text{if } k \neq \ell, \\ > 0 & \text{if } k = \ell, \end{cases} \tag{2.5}$$

then $\tau_1, \tau_2, \dots, \tau_n$ are the zeros of $\pi_n(\,\cdot\,; \mathrm{d}\lambda)$, and the λ_ν can be expressed in terms of the orthogonal polynomials as well. The former are all distinct and contained in the interior of the support interval of $\mathrm{d}\lambda$, the latter all positive. What is important here is the well-known fact that the orthogonal polynomials satisfy a three-term recurrence relation,

$$\begin{gathered} \pi_{k+1}(t) = (t - \alpha_k)\pi_k(t) - \beta_k \pi_{k-1}(t), \quad k = 0, 1, 2, \dots, \\ \pi_{-1}(t) = 0, \quad \pi_0(t) = 1, \end{gathered} \tag{2.6}$$

with well-determined real coefficients $\alpha_k = \alpha_k(\mathrm{d}\lambda)$ and $\beta_k = \beta_k(\mathrm{d}\lambda) > 0$. In terms of these, one defines the *Jacobi matrix*

$$\boldsymbol{J}(\mathrm{d}\lambda) = \begin{bmatrix} \alpha_0 & \sqrt{\beta_1} & & & & 0 \\ \sqrt{\beta_1} & \alpha_1 & \sqrt{\beta_2} & & & \\ & \sqrt{\beta_2} & \alpha_2 & \sqrt{\beta_3} & & \\ & & \ddots & \ddots & \ddots & \\ 0 & & & & & \end{bmatrix}, \tag{2.7}$$

in general an infinite symmetric tridiagonal matrix. Its leading principal minor matrix of order n will be denoted by

$$\boldsymbol{J}_n(\mathrm{d}\lambda) = [\boldsymbol{J}(\mathrm{d}\lambda)]_{[1:n,1:n]}. \tag{2.8}$$

ETNA
Kent State University
etna@mcs.kent.edu

Here, then, is the connection between the Gauss quadrature formula (2.2) and the algebraic eigenvalue problem: *the Gauss nodes* τ_ν *are the eigenvalues of* $\boldsymbol{J}_n(\mathrm{d}\lambda)$, *whereas the Gauss weights* λ_ν *are*

$$\lambda_\nu = \mu_0 \boldsymbol{v}_{\nu,1}^2, \tag{2.9}$$

where $\boldsymbol{v}_{\nu,1}$ *is the first component of the normalized eigenvector* $\boldsymbol{v}_\nu$ *corresponding to the eigenvalue* τ_ν. The eigenvalue charaterization of the nodes τ_ν is an easy consequence of the recurrence relation (2.6) and has been known for some time prior to the 1960s. The characterization (2.9) of the weights λ_ν is more intricate and seems to have first been observed in 1962 by Wilf [45, Ch.2, Exercise 9], or even previously, around 1954, by Goertzel [46]; it has also been used by the physicist Gordon in [34, p. 658]. The merit of Golub's work in [33] is to have clearly realized the great computational potential of this result and in fact to have developed a stable and efficient computational procedure based on the QL algorithm.

It is useful to note that the quadrature sum in (2.2) for smooth functions f can be written in terms of $\boldsymbol{J}_n = \boldsymbol{J}_n(\mathrm{d}\lambda)$ as

$$\sum_{\nu=1}^{n} \lambda_\nu f(\tau_\nu) = \mu_0 \boldsymbol{e}_1^T f(\boldsymbol{J}_n)\boldsymbol{e}_1, \quad \boldsymbol{e}_1^T = [1, 0, \ldots, 0] \in \mathbb{R}^n. \tag{2.10}$$

This follows readily from the spectral decomposition of $\boldsymbol{J}_n$ and (2.9). Also, for the remainder $R_n(f)$ in (2.2) one has (cf. [21, p. 291, (vii)])

$$|R_n(f)| \le \frac{\|f^{(2n)}\|_\infty}{(2n)!} \int_{\mathbb{R}} \pi_n^2(t)\mathrm{d}\lambda(t) = \frac{\|f^{(2n)}\|_\infty}{(2n)!} \beta_0\beta_1 \cdots \beta_n, \tag{2.11}$$

provided $f^{(2n)}$ is continuous on the support $\mathrm{supp}(\mathrm{d}\lambda)$ of $\mathrm{d}\lambda$. The ∞-norm of $f^{(2n)}$ is the maximum of $|f^{(2n)}|$ on $\mathrm{supp}(\mathrm{d}\lambda)$, and (2.11) holds regardless of whether or not $f^{(2n)}$ has constant sign.

The Jacobi matrix $\boldsymbol{J}_n(\mathrm{d}\lambda)$, and with it the Gauss quadrature rule, is uniquely determined by the first $2n$ moments $\mu_0, \mu_1, \ldots, \mu_{2n-1}$ of the measure $\mathrm{d}\lambda$. The *Chebyshev algorithm* (cf. [21, §2.3]) is a vehicle for passing directly from these $2n$ moments to the $2n$ recursion coefficients α_k, β_k, $k = 0, 1, \ldots, n-1$. Although numerically unstable, the procedure can be carried out in symbolic computation to arbitrary precision. (A Maple 5 script named `cheb.mws` can be found on the internet at `http://www.cs.purdue.edu/archives/2001/wxg/codes`.)

2.2. Gauss-Radau and Gauss-Lobatto quadrature. If the support of $\mathrm{d}\lambda$ is a finite interval $[a, b]$, the Gauss quadrature formula can be modified by requiring that one or both of the endpoints of $[a, b]$ be quadrature nodes. This gives rise to *Gauss-Radau* resp. *Gauss-Lobatto* formulae. Interestingly, both these formulae allow again a characterization in terms of eigenvalue problems; this was shown by Golub in [23].

2.2.1. Gauss-Radau quadrature. If $\tau_0 = a$ is the prescribed node, the $(n+1)$-point Gauss-Radau formula is

$$\int_a^b f(t)\mathrm{d}\lambda(t) = \lambda_0^a f(a) + \sum_{\nu=1}^{n} \lambda_\nu^a f(\tau_\nu^a) + R_n^a(f), \tag{2.12}$$

where the remainder now vanishes for polynomials of degree $\le 2n$,

$$R_n^a(f) = 0, \quad f \in \mathbb{P}_{2n}. \tag{2.13}$$

ETNA
Kent State University
etna@mcs.kent.edu

Define a *modified Jacobi matrix* of order $n+1$ by

$$\boldsymbol{J}_{n+1}^{R,a}(\mathrm{d}\lambda)=\begin{bmatrix}\boldsymbol{J}_n(\mathrm{d}\lambda) & \sqrt{\beta_n}\boldsymbol{e}_n\\ \sqrt{\beta_n}\boldsymbol{e}_n^T & \alpha_n^R\end{bmatrix}, \tag{2.14}$$

where $\boldsymbol{J}_n(\mathrm{d}\lambda)$ is the same matrix as in (2.8), $\beta_n=\beta_n(\mathrm{d}\lambda)$ as in (2.6),

$$\alpha_n^R=a-\beta_n\,\frac{\pi_{n-1}(a)}{\pi_n(a)}, \tag{2.15}$$

with $\pi_m(\,\cdot\,)=\pi_m(\,\cdot\,;\mathrm{d}\lambda)$, and $\boldsymbol{e}_n^T=[0,0,\ldots,1]$ the nth canonical basis vector of $\mathbb{R}^n$. Then, *the nodes in* (2.12) (*including* $\tau_0=a$) *are the eigenvalues of* $\boldsymbol{J}_{n+1}^{R,a}(\mathrm{d}\lambda)$, *and the weights* λ_ν^a, $\nu=0,1,\ldots,n$, *are again given by* (2.9) *in terms of the respective normalized eigenvectors*. An analogous result holds for the Gauss-Radau formula with prescribed node $\tau_{n+1}=b$,

$$\int_a^b f(t)\mathrm{d}\lambda(t)=\sum_{\nu=1}^{n}\lambda_\nu^b f(\tau_\nu^b)+\lambda_{n+1}^b f(b)+R_n^b(f). \tag{2.16}$$

The only change is replacing a in (2.15) by b, giving rise to a modified Jacobi matrix $\boldsymbol{J}_{n+1}^{R,b}(\mathrm{d}\lambda)$. Both quadrature sums in (2.12) and (2.16) allow a matrix representation analogous to (2.10), with $\boldsymbol{J}_n$ replaced by $\boldsymbol{J}_{n+1}^{R,a}$ resp. $\boldsymbol{J}_{n+1}^{R,b}$ and the dimension of $\boldsymbol{e}_1$ increased by 1.

The remainders R_n^a, R_n^b of the two Gauss-Radau formulae have the useful property

$$R_n^a(f)>0,\quad R_n^b(f)<0\quad \text{if}\ \ \operatorname{sgn} f^{(2n+1)}=1 \text{ on } [a,b], \tag{2.17}$$

with the inequalities reversed if $\operatorname{sgn} f^{(2n+1)}=-1$. This means that one of the two Gauss-Radau approximations is a lower bound, and the other an upper bound for the exact value of the integral.

It now takes $2n+1$ moments $\mu_0,\mu_1,\ldots,\mu_{2n}$ to obtain $\boldsymbol{J}_{n+1}^{R,a}(\mathrm{d}\lambda)$, $\boldsymbol{J}_{n+1}^{R,b}(\mathrm{d}\lambda)$ and the $(n+1)$-point Gauss-Radau formulae. Chebyshev's algorithm will provide the recursion coefficients needed to generate $\boldsymbol{J}_n(\mathrm{d}\lambda)$ in (2.14), β_n, and the ratio of orthogonal polynomials in (2.15).

The case of a discrete measure $\mathrm{d}\lambda_N$ supported on N points t_k with $a\le t_1<t_2<\cdots<t_N\le b$, and having positive jumps w_k^2 at t_k,

$$\int_{\mathbb{R}} f(t)\mathrm{d}\lambda_N(t):=\sum_{k=1}^{N}w_k^2 f(t_k), \tag{2.18}$$

is of some interest in applications. For one thing, the Gauss-Radau formulae (2.12), (2.16) (and, for that matter, the Gauss formula (2.2) as well), provide "compressions" of the sum $S=\sum_{k=1}^N w_k^2 f(t_k)$, i.e., approximations of S by a sum with fewer terms if $n<N-1$. When f is a polynomial of degree $\le 2n$, the compressed sums in fact have the same value as the original sum. More importantly, the formula (2.12) with $n<N$ together with the companion formula (2.16) furnish upper and lower bounds of S if $f^{(2n+1)}<0$ on $[a,b]$. Applications of this will be made in §§5.2–5.4. Chebyshev's algorithm can again be used to generate (2.12) and (2.16) from the moments of $\mathrm{d}\lambda_N$. There is also a numerically stable alternative to Lanczos's algorithm (cf. §5.1), due to Gragg and Harrod [35], generating the Jacobi matrix $\boldsymbol{J}_n(\mathrm{d}\lambda_N)$ directly from the quantities w_k and t_k in (2.18).

ETNA
Kent State University
etna@mcs.kent.edu

2.2.2. Gauss-Lobatto quadrature. Written as an $(n+2)$-point formula, the Gauss-Lobatto quadrature rule is

$$\int_a^b f(t)\mathrm{d}\lambda(t) = \lambda_0 f(a) + \sum_{\nu=1}^{n} \lambda_\nu f(\tau_\nu) + \lambda_{n+1} f(b) + R_n^{a,b}(f), \tag{2.19}$$

and has the exactness property

$$R_n^{a,b}(f) = 0, \quad f \in \mathbb{P}_{2n+1}, \tag{2.20}$$

and the sign property

$$R_n^{a,b}(f) < 0 \quad \text{if } \operatorname{sgn} f^{(2n+2)} = 1 \text{ on } [a,b], \tag{2.21}$$

with the inequality reversed if $\operatorname{sgn} f^{(2n+2)} = -1$. The appropriate modification of the Jacobi matrix is

$$\boldsymbol{J}_{n+2}^L(\mathrm{d}\lambda) = \begin{bmatrix} \boldsymbol{J}_{n+1}(\mathrm{d}\lambda) & \sqrt{\beta_{n+1}^L}\boldsymbol{e}_{n+1} \\ \sqrt{\beta_{n+1}^L}\boldsymbol{e}_{n+1}^T & \alpha_{n+1}^L \end{bmatrix}, \tag{2.22}$$

with notations similar as in (2.14). Here, α_{n+1}^L, β_{n+1}^L are defined as the solution of the 2×2 linear system

$$\begin{bmatrix} \pi_{n+1}(a) & \pi_n(a) \\ \pi_{n+1}(b) & \pi_n(b) \end{bmatrix} \begin{bmatrix} \alpha_{n+1}^L \\ \beta_{n+1}^L \end{bmatrix} = \begin{bmatrix} a\pi_{n+1}(a) \\ b\pi_{n+1}(b) \end{bmatrix}. \tag{2.23}$$

Then *the nodes of* (2.19) (including $\tau_0 = a$ and $\tau_{n+1} = b$) *are the eigenvalues of* $\boldsymbol{J}_{n+2}^L(\mathrm{d}\lambda)$ *and the weights* λ_ν, $\nu = 0, 1, \ldots, n, n+1$, *once again are given by* (2.9) *in terms of the respective normalized eigenvectors*. Hence, (2.10) again holds, with $\boldsymbol{J}_n$ replaced by $\boldsymbol{J}_{n+2}^L$ and $\boldsymbol{e}_1$ having dimension $n+2$.

2.3. Gauss quadrature with multiple nodes. The Gauss-Radau and Gauss-Lobatto formulae may be generalized by allowing an arbitrary number of prescribed nodes, even of arbitrary multiplicities, outside, or on the boundary, of the support interval of $\mathrm{d}\lambda$. (Those of even multiplicities may also be inside the support interval.) The remaining "free" nodes are either simple or of *odd* multiplicity. The quadrature rule in question, therefore, has the form

$$\int_{\mathbb{R}} f(t)\mathrm{d}\lambda(t) = \sum_{\nu=1}^{n} \sum_{\sigma=0}^{2s_\nu} \lambda_\nu^{(\sigma)} f^{(\sigma)}(\tau_\nu) + \sum_{\mu=1}^{m} \sum_{\rho=0}^{r_\mu - 1} \kappa_\mu^{(\rho)} f^{(\rho)}(u_\mu) + R_{n,m}(f), \tag{2.24}$$

where τ_ν are the free nodes and u_μ the prescribed ones, and the formula is required to have maximum degree of exactness $2\left(n + \sum_\nu s_\nu\right) + \sum_\mu r_\mu - 1$. This has a long history, going back to Christoffel (all $s_\nu = 0$ and $r_\mu = 1$) and including among its contributors Turán ($m = 0$, $s_\nu = s$ for all ν), Chakalov, Popoviciu, and Stancu (cf. [19, §2.2]).

The prescribed nodes u_μ give rise to the polynomial

$$u(t) = \omega \prod_{\mu=1}^{m} (t - u_\mu)^{r_\mu},$$

where $\omega = \pm 1$ is chosen such that $u(t) \geq 0$ for t on the support of $\mathrm{d}\lambda$. For the formula (2.24) to have maximum algebraic degree of exactness, the free nodes τ_ν ("Gauss nodes") must be chosen to satisfy

$$\int_{\mathbb{R}} \prod_{\nu=1}^{n} (t-\tau_\nu)^{2s_\nu+1} t^k u(t)\mathrm{d}\lambda(t) = 0, \quad k = 0, 1, \ldots, n-1.$$

By far the simplest scenario is the one in which $s_\nu = 0$ for all ν. In this case, τ_ν are the zeros of the polynomial $\pi_n(\,\cdot\,; u\mathrm{d}\lambda)$ of degree n orthogonal with respect to the (positive) measure $u\mathrm{d}\lambda$. This gives rise to the *problem of modification*: given the Jacobi matrix of the measure $\mathrm{d}\lambda$, find the Jacobi matrix of the modified measure $u\mathrm{d}\lambda$. An elegant solution of this problem involves genuine techniques from linear algebra; this will be described in §3. The weights $\lambda_\nu = \lambda_\nu^{(0)}$ are computable similarly as in (2.9) for ordinary Gauss quadrature, namely [27, §6]

$$\lambda_\nu = \mu_0 \boldsymbol{v}_{\nu,1}^2 / u(\tau_\nu), \quad \nu = 1, 2, \ldots, n,$$

where $\mu_0 = \int_{\mathbb{R}} u(t)\mathrm{d}\lambda(t)$ and $\boldsymbol{v}_{\nu,1}$ is the first component of the normalized eigenvector of $\boldsymbol{J}_n(u\mathrm{d}\lambda)$ corresponding to the eigenvalue τ_ν. For the computation of the remaining weights $\kappa_\mu^{(\rho)}$ in (2.24), see [37].

The case of multiple Gauss nodes ($s_\nu > 0$) is a good deal more complicated, requiring the iterative solution of a system of nonlinear equations for the τ_ν and the solution of linear algebraic systems for the weights $\lambda_\nu^{(\sigma)}$, $\kappa_\mu^{(\rho)}$; see, e.g., [27, §5] and [22].

2.4. Gauss-Kronrod quadrature. The quadrature rules discussed so far are products of the 19th century (except for the multiple-node Gauss rules). Let us turn now to a truly 20th-century product — the *Gauss-Kronrod formula*

$$\int_{\mathbb{R}} f(t)\mathrm{d}\lambda(t) = \sum_{\nu=1}^{n} \lambda_\nu^K f(\tau_\nu^G) + \sum_{\mu=1}^{n+1} \lambda_\mu^{*K} f(\tau_\mu^K) + R_n^K(f), \tag{2.25}$$

where τ_ν^G are the nodes of the n-point Gauss formula for $\mathrm{d}\lambda$, and the $n+1$ remaining nodes, called *Kronrod nodes*, as well as all $2n+1$ weights λ_ν^K, λ_μ^{*K} are determined by requiring maximum degree of exactness $3n+1$, i.e.,

$$R_n^K(f) = 0, \quad f \in \mathbb{P}_{3n+1}. \tag{2.26}$$

This was proposed by Kronrod [39] in the 1960s in the special case $\mathrm{d}\lambda(t) = \mathrm{d}t$ on $[-1, 1]$ as an economical way of estimating the error of the n-point Gauss-Legendre quadrature rule. The formula (2.25) nowadays is widely used in automatic and adaptive quadrature routines ([43], [17]).

Remarkably enough, there is an eigenvalue/vector characterization similar to those in §§2.1, 2.2 also for Gauss-Kronrod quadrature rules. This was discovered in 1997 by Laurie [40]. He assumes that there exists a positive Gauss-Kronrod formula (i.e., $\lambda_\nu^K > 0$, $\lambda_\mu^{*K} > 0$, and $\tau_\nu^K \in \mathbb{R}$), which need not be the case in general. (Indeed, the Kronrod nodes and all weights may well be complex.) The modified Jacobi matrix is now a symmetric tridiagonal matrix of order $2n+1$ and has the form

$$\boldsymbol{J}_{2n+1}^K(\mathrm{d}\lambda) = \begin{bmatrix} \boldsymbol{J}_n(\mathrm{d}\lambda) & \sqrt{\beta_n}\boldsymbol{e}_n & \boldsymbol{0} \\ \sqrt{\beta_n}\boldsymbol{e}_n^T & \alpha_n & \sqrt{\beta_{n+1}}\boldsymbol{e}_1^T \\ \boldsymbol{0} & \sqrt{\beta_{n+1}}\boldsymbol{e}_1 & \boldsymbol{J}_n^* \end{bmatrix} \tag{2.27}$$

ETNA
Kent State University
etna@mcs.kent.edu

with notation similar as before and $\boldsymbol{J}_n^*$ a symmetric tridiagonal matrix. The structure of $\boldsymbol{J}_n^*$ differs according as n is even or odd. For definiteness, suppose that n is even. Then

$$\boldsymbol{J}_n^* = \begin{bmatrix} \boldsymbol{J}_{[n+1:3n/2]}(\mathrm{d}\lambda) & \sqrt{\beta_n^*}\boldsymbol{e}_{n/2} \\ \sqrt{\beta_n^*}\boldsymbol{e}_{n/2}^T & \boldsymbol{J}^K_{[(3n+2)/2:2n]} \end{bmatrix} \quad (n \text{ even}), \tag{2.28}$$

where $\boldsymbol{J}_{[p:q]}(\mathrm{d}\lambda)$ denotes the principal minor matrix of $\boldsymbol{J}(\mathrm{d}\lambda)$ that has diagonal elements $\alpha_p, \alpha_{p+1}, \ldots, \alpha_q$, and similarly for $\boldsymbol{J}^K_{[p:q]}$. Thus, the upper left square block of $\boldsymbol{J}_n^*$ (of order $n/2$) may be assumed known, and the rest, including the constant β_n^*, is to be determined. Laurie devised an algorithm that determines the unknown elements of $\boldsymbol{J}_n^*$ in such a way that *the Gauss nodes* τ_ν^G *and Kronrod nodes* τ_μ^K *are the eigenvalues of* $\boldsymbol{J}^K_{2n+1}(\mathrm{d}\lambda)$ *and the weights are given by*

$$\begin{aligned} \lambda_\nu^K &= \mu_0[\boldsymbol{u}^K_{\nu,1}]^2, \quad \nu = 1, \ldots, n; \\ \lambda_\mu^{*K} &= \mu_0[\boldsymbol{u}^K_{n+\mu,1}]^2, \quad \mu = 1, \ldots, n, n+1, \end{aligned} \tag{2.29}$$

where $\boldsymbol{u}_1^K, \boldsymbol{u}_2^K, \ldots, \boldsymbol{u}_{2n+1}^K$ *are the normalized eigenvectors of* $\boldsymbol{J}^K_{2n+1}(\mathrm{d}\lambda)$ *corresponding to the eigenvalues* $\tau_1^G, \ldots, \tau_n^G$; $\tau_1^K, \ldots, \tau_{n+1}^K$, *and* $\boldsymbol{u}^K_{1,1}, \boldsymbol{u}^K_{2,1}, \ldots, \boldsymbol{u}^K_{2n+1,1}$ *their first components*. Moreover, $\boldsymbol{J}_n^*$ *in* (2.27) *has the same eigenvalues as* $\boldsymbol{J}_n(\mathrm{d}\lambda)$, i.e., the Gauss nodes $\tau_1^G, \ldots, \tau_n^G$.

If the Gauss nodes τ_ν^G are already known, as is often the case, there is some redundancy in Laurie's algorithm, inasmuch as it regenerates them all. In a joint paper with Calvetti, Gragg, and Reichel [8], Golub removes this redundancy by focusing directly on the Kronrod nodes. The basic idea is to observe that the trailing matrix $\boldsymbol{J}_n^*$ in (2.27) as well as the leading matrix $\boldsymbol{J}_{[n+1:3n/2]}(\mathrm{d}\lambda)$ in (2.28) (again with n assumed even) have their own sets of orthogonal polynomials and respective Gauss quadrature rules, the measures of which, however, are unknown. Since the eigenvalues of $\boldsymbol{J}_n^*$ are the same as those of $\boldsymbol{J}_n(\mathrm{d}\lambda)$, the former Gauss rule has nodes τ_ν^G, $\nu = 1, 2, \ldots, n$, and positive weights λ_ν^*, say, while the latter has certain nodes $\tilde{\tau}_\kappa$ and weights $\tilde{\lambda}_\kappa$, $\kappa = 1, 2, \ldots, n/2$. Let the matrices of normalized eigenvectors of $\boldsymbol{J}_n(\mathrm{d}\lambda)$ and $\boldsymbol{J}_n^*$ be $\boldsymbol{v} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n]$ and $\boldsymbol{v}^* = [\boldsymbol{v}_1^*, \boldsymbol{v}_2^*, \ldots, \boldsymbol{v}_n^*]$, respectively. The new algorithm will make use of the *last* components $\boldsymbol{v}_{1,n}, \boldsymbol{v}_{2,n}, \ldots, \boldsymbol{v}_{n,n}$ of the eigenvectors in $\boldsymbol{v}$ (assumed known) and the *first* components $\boldsymbol{v}^*_{1,1}, \boldsymbol{v}^*_{2,1}, \ldots, \boldsymbol{v}^*_{n,1}$ of those in $\boldsymbol{v}^*$. The latter, according to (2.9), are related to the Gauss weights λ_ν^* through

$$[\boldsymbol{v}^*_{\nu,1}]^2 = \lambda_\nu^*, \quad \nu = 1, 2, \ldots, n,$$

where the underlying measure is assumed normalized to have total mass 1, and one computes

$$\lambda_\nu^* = \sum_{\kappa=1}^{n/2} \ell_\nu(\tilde{\tau}_\kappa)\tilde{\lambda}_\kappa, \quad \nu = 1, 2, \ldots, n,$$

in terms of the second Gauss rule (for $\boldsymbol{J}_{[n+1:3n/2]}(\mathrm{d}\lambda)$) and the elementary Lagrange interpolation polynomials ℓ_ν associated with the nodes $\tau_1^G, \tau_2^G, \ldots, \tau_n^G$. Therefore,

$$\boldsymbol{v}^*_{\nu,1} = \sqrt{\lambda_\nu^*}, \quad \nu = 1, 2, \ldots, n. \tag{2.30}$$

Now let

$$\boldsymbol{J}_n^* = \boldsymbol{v}^*\boldsymbol{D}\boldsymbol{v}^{*T}, \quad \boldsymbol{D} = \mathrm{diag}\,(\tau_1^G, \tau_2^G, \ldots, \tau_n^G) \tag{2.31}$$

ETNA
Kent State University
etna@mcs.kent.edu

be the spectral decomposition of $\boldsymbol{J}_n^*$, and define

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{v} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 & \boldsymbol{0}^T \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{v}^* \end{bmatrix}, \tag{2.32}$$

a matrix of order $2n+1$. From (2.27), one gets

$$\boldsymbol{V}^T(\boldsymbol{J}_{2n+1}^K(\mathrm{d}\lambda) - \lambda \boldsymbol{I})\boldsymbol{V} = \begin{bmatrix} \boldsymbol{D} - \lambda\boldsymbol{I} & \sqrt{\beta_n}\boldsymbol{v}^T\boldsymbol{e}_n & \boldsymbol{0} \\ \sqrt{\beta_n}\boldsymbol{e}_n^T\boldsymbol{v} & \alpha_n - \lambda & \sqrt{\beta_{n+1}}\boldsymbol{e}_1^T\boldsymbol{v}^* \\ \boldsymbol{0} & \sqrt{\beta_{n+1}}\boldsymbol{v}^{*T}\boldsymbol{e}_1 & \boldsymbol{D} - \lambda\boldsymbol{I} \end{bmatrix},$$

where the matrix on the right is a diagonal matrix plus a Swiss cross containing the known elements $\boldsymbol{e}_n^T\boldsymbol{v}$ and the elements $\boldsymbol{e}_1^T\boldsymbol{v}^*$ that were computed in (2.30). A further (cosmetic) orthogonal similarity transformation involving a permutation and a sequence of Givens rotations can be applied to yield

$$\tilde{\boldsymbol{V}}^T(\boldsymbol{J}_{2n+1}^K(\mathrm{d}\lambda) - \lambda \boldsymbol{I})\tilde{\boldsymbol{V}} = \left[\begin{array}{c|cc} \boldsymbol{D} - \lambda\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \hline \boldsymbol{0} & \boldsymbol{D} - \lambda\boldsymbol{I} & \boldsymbol{c} \\ \boldsymbol{0}^T & \boldsymbol{c}^T & \alpha_n - \lambda \end{array}\right], \tag{2.33}$$

where $\tilde{\boldsymbol{V}}$ is the transformed matrix $\boldsymbol{V}$ and $\boldsymbol{c}$ a vector containing the entries in positions $n+1$ to $2n$ of the transformed vector $[\sqrt{\beta_n}\boldsymbol{e}_n^T\boldsymbol{v}, \sqrt{\beta_{n+1}}\boldsymbol{e}_1^T\boldsymbol{v}^*, \alpha_n]$. Eq. (2.33) now reveals that one set of eigenvalues of $\boldsymbol{J}_{2n+1}^K(\mathrm{d}\lambda)$ is $\{\tau_1^G, \tau_2^G, \ldots, \tau_n^G\}$, while the remaining eigenvalues are those of the trailing block in (2.33). From

$$\begin{bmatrix} \boldsymbol{D} - \lambda\boldsymbol{I} & \boldsymbol{c} \\ \boldsymbol{c}^T & \alpha_n - \lambda \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{c}^T(\boldsymbol{D} - \lambda\boldsymbol{I})^{-1} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{D} - \lambda\boldsymbol{I} & \boldsymbol{c} \\ \boldsymbol{0}^T & -f(\lambda) \end{bmatrix},$$

where

$$f(\lambda) = \lambda - \alpha_n + \sum_{\nu=1}^{n} \frac{c_\nu^2}{\tau_\nu^G - \lambda}, \quad \boldsymbol{c}^T = [c_1, c_2, \ldots, c_n], \tag{2.34}$$

it follows that the remaining eigenvalues, i.e., the Kronrod nodes, are the zeros of $f(\lambda)$. It is evident from (2.34) that they interlace with the Gauss nodes τ_ν^G. The normalized eigenvectors $\boldsymbol{u}_1^K, \boldsymbol{u}_2^K, \ldots, \boldsymbol{u}_{2n+1}^K$ required to compute the weights λ_ν^K, λ_μ^{*K} via (2.29) can be computed from the columns of $\boldsymbol{V}$ by keeping track of the orthogonal transformations.

3. Orthogonal polynomials. The connection between orthogonal polynomials and Jacobi matrices (cf. §2.1) gives rise to several interesting problems:

(a) Given the Jacobi matrix for the measure $\mathrm{d}\lambda$, find the Jacobi matrix for the modified measure $\mathrm{d}\lambda_{\mathrm{mod}} = r\mathrm{d}\lambda$, where r is either a polynomial or a rational function.

(b) Given the Jacobi matrices for two measures $\mathrm{d}\lambda_1$ and $\mathrm{d}\lambda_2$, find the Jacobi matrix for $\mathrm{d}\lambda = \mathrm{d}\lambda_1 + \mathrm{d}\lambda_2$.

(c) Let $[c_j, d_j]$ be a finite set of intervals, disjoint or not, and $\mathrm{d}\lambda_j$ a positive measure on $[c_j, d_j]$. Let $\mathrm{d}\lambda(t) = \sum_j \chi_{[c_j,d_l]}(t)\mathrm{d}\lambda_j(t)$, where $\chi_{[c_j,d_j]}$ is the characteristic function of the interval $[c_j, d_j]$,

$$\chi_{[c_j,d_j]}(t) = \begin{cases} 1 \text{ if } t \in [c_j, d_j], \\ 0 \text{ otherwise.} \end{cases}$$

Knowing the Jacobi matrices $\boldsymbol{J}^{(j)}$ for $\mathrm{d}\lambda_j$, find the Jacobi matrix for $\mathrm{d}\lambda$.

The problem in (b) is discussed in [13], where three algorithms are developed for its solution. We do not attempt to describe them here, since they are rather technical and not easily summarized. Suffice it to say that linear algebra figures prominently in all three of these algorithms. A special case of Problem (a) — modification of the measure by a polynomial factor — a problem discussed in [27] and [38], is considered in §§3.2, 3.3. It is related to a classical theorem of Christoffel (cf., e.g., [19, p. 85]), which expresses the orthogonal polynomials for the modified measure in determinantal form in terms of the orthogonal polynomials of the original measure. For algorithmic and computational purposes, however, the use of Jacobi matrices is vastly superior. The case of rational r, in particular $r(t) = (t - x)^{-1}$ with real x outside the support of $\mathrm{d}\lambda$, and $r(t) = [(t - x)^2 + y^2]^{-1}$, $y > 0$, is treated in [15], where algorithms are developed that are similar to those in [20] but are derived in a different manner. Problem (c) is dealt with in §3.4. Note that Problem (b) is a special case of Problem (c).

We begin with an integral representation of Jacobi matrices and then in turn describe modification of the measure by a linear, quadratic, and higher-degree polynomial, and solution procedures for Problem (c).

3.1. Integral representation of the Jacobi matrix. Let $\tilde{\pi}_0, \tilde{\pi}_1, \tilde{\pi}_2, \ldots$ be the system of *orthonormal* polynomials with respect to to the measure $\mathrm{d}\lambda$, that is,

$$\tilde{\pi}_k(t) = \frac{\pi_k(t)}{\|\pi_k\|}, \quad \|\pi_k\|^2 = \int_{\mathbb{R}} \pi_k^2(t)\mathrm{d}\lambda, \tag{3.1}$$

with π_k as in (2.5), (2.6). They satisfy the recurrence relation

$$\begin{gathered} \sqrt{\beta_{k+1}}\tilde{\pi}_{k+1}(t) = (t - \alpha_k)\tilde{\pi}_k(t) - \sqrt{\beta_k}\tilde{\pi}_{k-1}(t), \quad k = 0, 1, 2, \ldots, \\ \tilde{\pi}_{-1}(t) = 0, \quad \tilde{\pi}_0(t) = 1/\sqrt{\beta_0}, \end{gathered} \tag{3.2}$$

with recursion coefficients α_k, β_k as in (2.6) and $\beta_0 = \int_{\mathbb{R}} \mathrm{d}\lambda(t)\ (= \mu_0)$. From (3.2) and the orthonormality of the polynomials $\tilde{\pi}_k$ one easily checks that

$$\int_{\mathbb{R}} t\tilde{\pi}_k(t)\tilde{\pi}_\ell(t)\mathrm{d}\lambda(t) = \begin{cases} 0 & \text{if } |k - \ell| > 1, \\ \sqrt{\beta_{k+1}} & \text{if } |k - \ell| = 1, \\ \alpha_k & \text{if } k = \ell. \end{cases} \tag{3.3}$$

This allows us to represent the Jacobi matrix $\boldsymbol{J} = \boldsymbol{J}_n(\mathrm{d}\lambda)$ of order n (cf. (2.8)) in integral form as

$$\boldsymbol{J} = \int_{\mathbb{R}} t\boldsymbol{p}(t)\boldsymbol{p}^T(t)\mathrm{d}\lambda(t), \tag{3.4}$$

where

$$\boldsymbol{p}^T(t) = [\tilde{\pi}_0(t), \tilde{\pi}_1(t), \ldots, \tilde{\pi}_{n-1}(t)]. \tag{3.5}$$

ETNA
Kent State University
etna@mcs.kent.edu

Orthogonality, on the other hand, is expressible as

$$\int_{\mathbb{R}} \boldsymbol{p}(t)\boldsymbol{p}^T(t)\mathrm{d}\lambda(t) = \boldsymbol{I}, \tag{3.6}$$

where $\boldsymbol{I} = \boldsymbol{I}_n$ is the unit matrix of order n, and the first n recurrence relations in (3.2) can be given the form

$$t\boldsymbol{p}(t) = \boldsymbol{J}\boldsymbol{p}(t) + \sqrt{\beta_n}\tilde{\pi}_n(t)\boldsymbol{e}_n, \tag{3.7}$$

where $\boldsymbol{e}_n = [0, 0, \ldots, 1]^T \in \mathbb{R}^n$.

3.2. Modification by a linear factor. The problem to be studied is the effect on the Jacobi matrix $\boldsymbol{J}$ of modifying the (positive) measure $\mathrm{d}\lambda$ into a measure $\mathrm{d}\lambda_{\mathrm{mod}}$ defined by

$$\mathrm{d}\lambda_{\mathrm{mod}}(t) = \omega(t - c)\mathrm{d}\lambda(t), \tag{3.8}$$

where c is a real constant outside, or on the boundary, of the support interval of $\mathrm{d}\lambda$ and $\omega = \pm 1$ chosen such that the measure $\mathrm{d}\lambda_{\mathrm{mod}}$ is again positive. A solution of this problem has been given already by Galant [16] and was taken up again, and simplified, in [27].

The symmetric matrix $\omega(\boldsymbol{J} - c\boldsymbol{I})$ is positive definite since by the assumtions made regarding (3.8) all its eigenvalues are positive. It thus admits a Cholesky decomposition

$$\omega(\boldsymbol{J} - c\boldsymbol{I}) = \boldsymbol{L}\boldsymbol{L}^T, \tag{3.9}$$

where $\boldsymbol{L}$ is lower triangular and bidiagonal. By (3.4), (3.6), and (3.8) one has

$$\omega(\boldsymbol{J} - c\boldsymbol{I}) = \omega \int_{\mathbb{R}} (t - c)\boldsymbol{p}(t)\boldsymbol{p}^T(t)\mathrm{d}\lambda(t) = \int_{\mathbb{R}} \boldsymbol{p}(t)\boldsymbol{p}^T(t)\mathrm{d}\lambda_{\mathrm{mod}}(t).$$

This may be written as

$$\omega(\boldsymbol{J} - c\boldsymbol{I}) = \boldsymbol{L}\int_{\mathbb{R}} \boldsymbol{L}^{-1}\boldsymbol{p}(t)\boldsymbol{p}^T(t)\boldsymbol{L}^{-T}\mathrm{d}\lambda_{\mathrm{mod}}(t)\boldsymbol{L}^T,$$

which, since $\boldsymbol{L}^{-1}\omega(\boldsymbol{J} - c\boldsymbol{I})\boldsymbol{L}^{-T} = \boldsymbol{I}$ by (3.9), implies

$$\int_{\mathbb{R}} \boldsymbol{p}_{\mathrm{mod}}(t)\boldsymbol{p}_{\mathrm{mod}}^T(t)\mathrm{d}\lambda_{\mathrm{mod}}(t) = \boldsymbol{I}, \tag{3.10}$$

where

$$\boldsymbol{p}_{\mathrm{mod}}(t) = \boldsymbol{L}^{-1}\boldsymbol{p}(t). \tag{3.11}$$

This means that $\boldsymbol{p}_{\mathrm{mod}}$ are the orthonormal polynomials with respect to the measure $\mathrm{d}\lambda_{\mathrm{mod}}$. What is the corresponding Jacobi matrix $\boldsymbol{J}_{\mathrm{mod}}$?

First observe that from (3.7) one has

$$(t - c)\boldsymbol{p}(t) = (\boldsymbol{J} - c\boldsymbol{I})\boldsymbol{p}(t) + \sqrt{\beta_n}\tilde{\pi}_n(t)\boldsymbol{e}_n. \tag{3.12}$$

Using the analogues of (3.4), (3.6), one has, by (3.11) and (3.8),

$$\begin{aligned} \boldsymbol{J}_{\mathrm{mod}} - c\boldsymbol{I} &= \int_{\mathbb{R}} (t - c)\boldsymbol{p}_{\mathrm{mod}}(t)\boldsymbol{p}_{\mathrm{mod}}^T(t)\mathrm{d}\lambda_{\mathrm{mod}}(t) \\ &= \omega\boldsymbol{L}^{-1}\int_{\mathbb{R}} (t - c)^2\boldsymbol{p}(t)\boldsymbol{p}^T(t)\mathrm{d}\lambda(t)\boldsymbol{L}^{-T}. \end{aligned} \tag{3.13}$$

Multiplying (3.12) with its transpose, one gets

$$(t-c)^2\boldsymbol{p}(t)\boldsymbol{p}^T(t) = [(\boldsymbol{J}-c\boldsymbol{I})\boldsymbol{p}(t)+\sqrt{\beta_n}\tilde{\pi}_n(t)\boldsymbol{e}_n][\boldsymbol{p}^T(t)(\boldsymbol{J}-c\boldsymbol{I})+\sqrt{\beta_n}\tilde{\pi}_n(t)\boldsymbol{e}_n^T],$$

and observing that by (3.6)

$$\int_{\mathbb{R}}(\boldsymbol{J}-c\boldsymbol{I})\boldsymbol{p}(t)\boldsymbol{p}^T(t)(\boldsymbol{J}-c\boldsymbol{I})\mathrm{d}\lambda(t) = (\boldsymbol{J}-c\boldsymbol{I})^2$$

and by orthonormality

$$\int_{\mathbb{R}}(\boldsymbol{J}-c\boldsymbol{I})\boldsymbol{p}(t)\tilde{\pi}_n(t)\boldsymbol{e}_n^T\mathrm{d}\lambda(t) = 0, \quad \int_{\mathbb{R}}\tilde{\pi}_n^2(t)\mathrm{d}\lambda(t) = 1,$$

one finds

$$\int_{\mathbb{R}}(t-c)^2\boldsymbol{p}(t)\boldsymbol{p}^T(t)\mathrm{d}\lambda(t) = (\boldsymbol{J}-c\boldsymbol{I})^2+\beta_n\boldsymbol{e}_n\boldsymbol{e}_n^T.$$

Thus, by (3.13),

$$\boldsymbol{J}_{\mathrm{mod}}-c\boldsymbol{I} = \omega\boldsymbol{L}^{-1}\left((\boldsymbol{J}-c\boldsymbol{I})^2+\beta_n\boldsymbol{e}_n\boldsymbol{e}_n^T\right)\boldsymbol{L}^{-T}.$$

Substituting from (3.9) yields for the desired Jacobi matrix

$$\boldsymbol{J}_{\mathrm{mod}} = \frac{1}{\omega}\boldsymbol{L}^T\boldsymbol{L}+c\boldsymbol{I}+\omega\beta_n\boldsymbol{L}^{-1}\boldsymbol{e}_n\boldsymbol{e}_n^T\boldsymbol{L}^{-T}$$

and noting that $\boldsymbol{L}^{-1}\boldsymbol{e}_n = \boldsymbol{e}_n/(\boldsymbol{e}_n^T\boldsymbol{L}\boldsymbol{e}_n)$ finally

$$\boldsymbol{J}_{\mathrm{mod}} = \frac{1}{\omega}\boldsymbol{L}^T\boldsymbol{L}+c\boldsymbol{I}+\gamma\boldsymbol{e}_n\boldsymbol{e}_n^T, \quad \gamma = \omega\beta_n/(\boldsymbol{e}_n^T\boldsymbol{L}\boldsymbol{e}_n)^2. \tag{3.14}$$

Equations (3.9) and (3.14) allow the following interpretation: *The matrix* $\boldsymbol{J}_1 := \boldsymbol{J}_{\mathrm{mod}} - \gamma\boldsymbol{e}_n\boldsymbol{e}_n^T$ *is the result of one step of the symmetric LR algorithm with shift* c,

$$\boldsymbol{J}-c\boldsymbol{I} = \frac{1}{\omega}\boldsymbol{L}\boldsymbol{L}^T, \quad \boldsymbol{J}_1 = \frac{1}{\omega}\boldsymbol{L}^T\boldsymbol{L}+c\boldsymbol{I}. \tag{3.15}$$

Note that $\boldsymbol{J}_1$ differs from $\boldsymbol{J}_{\mathrm{mod}}$ only by one element in the lower right-hand corner. We could get rid of it by deleting the last row and last column of $\boldsymbol{J}_1$. This would yield the desired Jacobi matrix of order $n-1$. If we are interested in the Jacobi matrix $\boldsymbol{J}_{n,\mathrm{mod}}$ of order n, we can *apply the symmetric LR algorithm* (with shift c) *to* $\boldsymbol{J}_{n+1}(\mathrm{d}\lambda)$ *and then obtain* $\boldsymbol{J}_{n,\mathrm{mod}}$ *by discarding the last row and last column in the resulting matrix.*

3.3. Modification by a quadratic and higher-degree factor. Modification by a quadratic factor $(t-c_1)(t-c_2)$ essentially amounts to two applications of (3.15),

$$\boldsymbol{J}-c_1\boldsymbol{I} = \frac{1}{\omega_1}\boldsymbol{L}_1\boldsymbol{L}_1^T, \quad \boldsymbol{J}_1 = \frac{1}{\omega_1}\boldsymbol{L}_1^T\boldsymbol{L}_1+c_1\boldsymbol{I} \tag{3.16}$$

followed by

$$\boldsymbol{J}_1-c_2\boldsymbol{I} = \frac{1}{\omega_2}\boldsymbol{L}_2\boldsymbol{L}_2^T, \quad \boldsymbol{J}_2 = \frac{1}{\omega_2}\boldsymbol{L}_2^T\boldsymbol{L}_2+c_2\boldsymbol{I}. \tag{3.17}$$

From the second and third of these equations one gets

$$\boldsymbol{L}_2\boldsymbol{L}_2^T = \omega_2\left(\frac{1}{\omega_1}\boldsymbol{L}_1^T\boldsymbol{L}_1 + (c_1 - c_2)\boldsymbol{I}\right).$$

In particular, if $c_1 = c_2 = c$ and $\omega_2/\omega_1 = 1$, then

$$\boldsymbol{L}_2\boldsymbol{L}_2^T = \boldsymbol{L}_1^T\boldsymbol{L}_1. \tag{3.18}$$

Let

$$\boldsymbol{Q} = \boldsymbol{L}_1^{-T}\boldsymbol{L}_2, \quad \boldsymbol{R} = \boldsymbol{L}_2^T\boldsymbol{L}_1^T. \tag{3.19}$$

Then, using (3.18), one computes

$$\begin{aligned}\boldsymbol{Q}^T\boldsymbol{Q} &= \boldsymbol{L}_2^T\boldsymbol{L}_1^{-1}\boldsymbol{L}_1^{-T}\boldsymbol{L}_2 = \boldsymbol{L}_2^T(\boldsymbol{L}_1^T\boldsymbol{L}_1)^{-1}\boldsymbol{L}_2\\ &= \boldsymbol{L}_2^T(\boldsymbol{L}_2\boldsymbol{L}_2^T)^{-1}\boldsymbol{L}_2 = \boldsymbol{L}_2^T\boldsymbol{L}_2^{-T}\boldsymbol{L}_2^{-1}\boldsymbol{L}_2 = \boldsymbol{I},\end{aligned}$$

so that $\boldsymbol{Q}$ is orthogonal. As a product of two upper triangular matrices, $\boldsymbol{R}$ is upper triangular. Since, again by (3.18), $\boldsymbol{QR} = \boldsymbol{L}_1^{-T}\boldsymbol{L}_2\boldsymbol{L}_2^T\boldsymbol{L}_1^T = \boldsymbol{L}_1^{-T}\boldsymbol{L}_1^T\boldsymbol{L}_1\boldsymbol{L}_1^T = \boldsymbol{L}_1\boldsymbol{L}_1^T$, the first equation of (3.16) can be written as

$$\boldsymbol{J} - c\boldsymbol{I} = \boldsymbol{QR} \tag{3.20}$$

and the second of (3.17) similarly as

$$\boldsymbol{J}_2 = \boldsymbol{RQ} + c\boldsymbol{I}. \tag{3.21}$$

Thus, $\boldsymbol{J}_2$ is obtained by one step of the QR algorithm with shift c. It is now clear how the modification

$$\mathrm{d}\lambda_{\mathrm{mod}}(t) = (t-c)^2\mathrm{d}\lambda(t) \tag{3.22}$$

is to be handled: *apply one step of the QR algorithm with shift c to the Jacobi matrix $\boldsymbol{J}_{n+2}(\mathrm{d}\lambda)$ of order $n+2$ and discard the last two rows and columns of the resulting matrix to obtain $\boldsymbol{J}_{n,\mathrm{mod}}$.*

More generally, a modification $\mathrm{d}\lambda_{\mathrm{mod}}(t) = (t-c)^{2m}\mathrm{d}\lambda(t)$ with an even power can be handled by m steps of the QR algorithm with shift c, discarding the appropriate number of rows and columns, and a modification $\mathrm{d}\lambda_{\mathrm{mod}}(t) = (t-c)^{2m+1}\mathrm{d}\lambda(t)$ by an odd power by means of m shifted QR steps followed by one step of the symmetric LR algorithm as in §3.2. In this way it is possible to accomplish the modification $\mathrm{d}\lambda_{\mathrm{mod}}(t) = r(t)\mathrm{d}\lambda(t)$ for any polynomial r with real roots and $r(t) \geq 0$ for t on the support of $\mathrm{d}\lambda$. Alternative methods, not necessarily requiring knowledge of the roots, are developed in [38].

3.4. Polynomials orthogonal on several intervals. Here we describe two solution procedures for Problem (c) based respectively on Stieltjes's procedure (cf. [21, §2.1]) and modified moments.

3.4.1. Solution by Stieltjes's procedure. Suppose we are interested in generating the Jacobi matrix $\boldsymbol{J} = \boldsymbol{J}_n(\mathrm{d}\lambda)$ of order n for the measure $\mathrm{d}\lambda(t) = \sum_j \chi_{[c_j,d_j]}(t)\,\mathrm{d}\lambda_j(t)$. It is well known that the recursion coefficients $\alpha_k = \alpha_k(\mathrm{d}\lambda)$, $\beta_k = \beta_k(\mathrm{d}\lambda)$ satisfy

$$\alpha_k = \frac{(t\pi_k,\pi_k)_{\mathrm{d}\lambda}}{(\pi_k,\pi_k)_{\mathrm{d}\lambda}}, \quad k = 0,1,\ldots,n-1, \tag{3.23}$$

$$\beta_k = \frac{(\pi_k, \pi_k)_{\mathrm{d}\lambda}}{(\pi_{k-1}, \pi_{k-1})_{\mathrm{d}\lambda}}, \quad k = 1, 2, \ldots, n-1, \tag{3.24}$$

where

$$(u, v)_{\mathrm{d}\lambda} = \int_{\mathbb{R}} u(t)v(t)\mathrm{d}\lambda(t) \tag{3.25}$$

is the inner product associated with $\mathrm{d}\lambda$. We also recall the basic recurrence relation (cf. (2.6))

$$\begin{gathered} \pi_{k+1}(t) = (t - \alpha_k)\pi_k(t) - \beta_k \pi_{k-1}(t), \quad k = 0, 1, \ldots, n-1, \\ \pi_{-1}(t) = 0, \quad \pi_0(t) = 1, \end{gathered} \tag{3.26}$$

satisfied by the (monic) orthogonal polynomials $\pi_k(\,\cdot\,) = \pi_k(\,\cdot\,; \mathrm{d}\lambda)$. For convenience, we let, as before,

$$\beta_0 = \int_{\mathbb{R}} \mathrm{d}\lambda(t). \tag{3.27}$$

Stieltjes's procedure consists in the following: Compute α_0 from (3.23) with $k = 0$ and β_0 from (3.27). Then use (3.26) with $k = 0$ to generate π_1. Go back to Eqs. (3.23), (3.24) and use them for $k = 1$ to obtain α_1, β_1. Then (3.26) is reapplied with $k = 1$ to get π_2, etc. This procedure, alternating between (3.23), (3.24) and (3.26), is continued until α_{n-1}, β_{n-1} are obtained.

The principal issue in this procedure is the computation of the inner products in (3.23), (3.24). Since they require integrating polynomials of degrees at most $2n-1$, one can use n-point Gauss quadrature

$$\int_{c_j}^{d_j} p(t)\mathrm{d}\lambda_j(t) = \sum_{\nu=1}^{n} \lambda_\nu^{(j)} p(\tau_\nu^{(j)}), \quad p \in \mathbb{P}_{2n-1}, \tag{3.28}$$

for the measure $\mathrm{d}\lambda_j$ on each constituent interval $[c_j, d_j]$ of $\mathrm{d}\lambda$. It has been observed in [14] that the explicit calculation of the Gauss nodes and weights is not required, but only matrix manipulations involving the Jacobi matrices $\boldsymbol{J}^{(j)}$ (of order n) for $\mathrm{d}\lambda_j$ (cf. (2.10)).

We illustrate this for the inner product

$$(t\pi_k, \pi_k)_{\mathrm{d}\lambda} = \int_{\mathbb{R}} t\pi_k^2(t)\mathrm{d}\lambda(t) = \sum_j \int_{c_j}^{d_j} t\pi_k^2(t)\mathrm{d}\lambda_j(t). \tag{3.29}$$

Denote $\beta_0^{(j)} = \int_{c_j}^{d_j} \mathrm{d}\lambda_j(t)$ and let

$$\boldsymbol{\zeta}_k^{(j)} = \pi_k(\boldsymbol{J}^{(j)})\boldsymbol{e}_1, \quad \boldsymbol{e}_1^T = [1, 0, \ldots, 0] \in \mathbb{R}^n. \tag{3.30}$$

Then, using (3.29), (3.28), and (2.10), one has

$$\begin{aligned} (t\pi_k, \pi_k)_{\mathrm{d}\lambda} &= \sum_j \lambda_\nu^{(j)} \tau_\nu^{(j)} \pi_k^2(\tau_\nu^{(j)}) \\ &= \sum_j \beta_0^{(j)} \boldsymbol{e}_1^T \boldsymbol{J}^{(j)} [\pi_k(\boldsymbol{J}^{(j)})]^2 \boldsymbol{e}_1 = \sum_j \beta_0^{(j)} \boldsymbol{e}_1^T \pi_k(\boldsymbol{J}^{(j)}) \boldsymbol{J}^{(j)} \pi_k(\boldsymbol{J}^{(j)}) \boldsymbol{e}_1, \end{aligned}$$

that is,

$$(3.31)\qquad (t\pi_k, \pi_k)_{\mathrm{d}\lambda} = \sum_j \beta_0^{(j)} \boldsymbol{\zeta}_k^{(j)T} \boldsymbol{J}^{(j)} \boldsymbol{\zeta}_k^{(j)}.$$

Similarly (in fact a bit simpler), one finds

$$(3.32)\qquad (\pi_k, \pi_k)_{\mathrm{d}\lambda} = \sum_j \beta_0^{(j)} \boldsymbol{\zeta}_k^{(j)T} \boldsymbol{\zeta}_k^{(j)}.$$

The updating of the $\boldsymbol{\zeta}_k^{(j)}$ required in Stieltjes's procedure follows immediately from (3.26),

$$(3.33)\qquad \boldsymbol{\zeta}_{k+1}^{(j)} = (\boldsymbol{J}^{(j)} - \alpha_k \boldsymbol{I})\boldsymbol{\zeta}_k^{(j)} - \beta_k \boldsymbol{\zeta}_{k-1}^{(j)},$$

where $\boldsymbol{I}$ is the unit matrix of order n and $\boldsymbol{\zeta}_{-1}^{(j)} = \mathbf{0}$.

3.4.2. Solution by the modified Chebyshev algorithm. The desired recursion coefficients $\alpha_k(\mathrm{d}\lambda)$, $\beta_k(\mathrm{d}\lambda)$, $k = 0, 1, \ldots, n-1$, can also be produced from the first $2n$ *modified moments*

$$(3.34)\qquad m_k = \int_{\mathbb{R}} p_k(t)\mathrm{d}\lambda(t), \quad k = 0, 1, \ldots, 2n-1,$$

where $\{p_k\}$ is a system of polynomials satisfying a three-term recurrence relation

$$(3.35)\qquad \begin{gathered} p_{k+1}(t) = (t - a_k)p_k(t) - b_k p_{k-1}(t), \quad k = 0, 1, \ldots, n-1, \\ p_{-1}(t) = 0, \quad p_0(t) = 1 \end{gathered}$$

with known coefficients a_k, b_k. A procedure accomplishing this is the *modified Chebyshev algorithm* (cf. [21, §2.4]); this works also if the polynomials $\{p_k\}$ satisfy an extended recurrence relation $p_{k+1}(t) = tp_k(t) - \sum_{j=0}^{k} c_{kj} p_j(t)$, and even if the measure $\mathrm{d}\lambda$ is indefinite (see, e.g., [26]). The computation of the modified moments (3.34) by Gauss quadrature is entirely analogous to the computation of inner products in §3.4.1. Letting now

$$(3.36)\qquad \boldsymbol{z}_k^{(j)} = p_k(\boldsymbol{J}^{(j)})\boldsymbol{e}_1,$$

one finds

$$(3.37)\qquad m_k = \sum_j \beta_0^{(j)} \boldsymbol{z}_k^{(j)T} \boldsymbol{e}_1.$$

Updating the vectors $\boldsymbol{z}_k^{(j)}$ can again be done via the recurrence relation (3.35),

$$(3.38)\qquad \boldsymbol{z}_{k+1}^{(j)} = (\boldsymbol{J}^{(j)} - a_k \boldsymbol{I})\boldsymbol{z}_k^{(j)} - b_k \boldsymbol{z}_{k-1}^{(j)}, \quad \boldsymbol{z}_{-1}^{(j)} = \mathbf{0}.$$

There is yet a third algorithm proposed in [14], which is based on a fast Cholesky decomposition. For this, we refer to the original source.

We remark that the modified Chebyshev algorithm provides an alternative way of solving Problem (a) for polynomial modifications $\mathrm{d}\lambda_{\mathrm{mod}}(t) = r(t)\mathrm{d}\lambda(t)$ (cf. [19, p. 123], [15]). Indeed, if $r \in \mathbb{P}_m$, then r can be expressed in terms of the polynomials p_k as

$$(3.39)\qquad r(t) = \sum_{j=0}^{m} c_j p_j(t).$$

If one assumes that $\{p_k\}$ are orthogonal relative to the measure $\mathrm{d}\lambda$, then the modified moments $m_k = \int_{\mathbb{R}} p_k(t)\mathrm{d}\lambda_{\mathrm{mod}}(t)$ are simply

$$m_k = \begin{cases} c_k \int_{\mathbb{R}} p_k^2(t)\mathrm{d}\lambda(t) \text{ if } k \le m, \\ 0 \text{ if } k > m. \end{cases} \tag{3.40}$$

The modified Chebyshev algorithm, if $m < k$, in fact simplifies, owing to the $k - m + 1$ zero modified moments in (3.40).

4. The least squares problem. The polynomial least squares problem P_N is as follows: Given N data points (t_k, y_k), $k = 1, 2, \dots, N$, where $t_1, t_2, \dots, t_N$ are mutually distinct points on the real line, and N positive weights w_k^2, find a polynomial $q^0 \in \mathbb{P}_{n-1}$, $n \le N$, such that

$$P_N: \quad \sum_{k=1}^{N} w_k^2 \left(y_k - q^0(t_k)\right)^2 \le \sum_{k=1}^{N} w_k^2 \left(y_k - q(t_k)\right)^2 \quad \text{for all } q \in \mathbb{P}_{n-1}.$$

With Problem P_N one associates the discrete inner product

$$(u, v)_{\mathrm{d}\lambda_N} = \int_{\mathbb{R}} u(t)v(t)\mathrm{d}\lambda_N := \sum_{k=1}^{N} w_k^2 u(t_k)v(t_k), \tag{4.1}$$

and the norm $\|u\|_{\mathrm{d}\lambda_N}^2 = (u, u)_{\mathrm{d}\lambda_N}$, in terms of which P_N can be written as

$$\|y - q^0\|_{\mathrm{d}\lambda_N}^2 \le \|y - q\|_{\mathrm{d}\lambda_N}^2 \quad \text{for all } q \in \mathbb{P}_{n-1}.$$

It is well known that the problem allows an elegant solution by means of the orthonormal polynomials $\tilde{\pi}_k(\,\cdot\,) = \tilde{\pi}_k(\,\cdot\,; \mathrm{d}\lambda_N)$. Recall that there are exactly N such polynomials, $\tilde{\pi}_0, \tilde{\pi}_1, \dots, \tilde{\pi}_{N-1}$; we define

$$\tilde{\pi}_N(t) = \frac{\prod_{k=1}^{N}(t - t_k)}{\|\pi_{N-1}\|}, \tag{4.2}$$

where π_{N-1} is the *monic* orthogonal polynomial.

4.1. Matrix formulation of the least squares problem and its solution. Let $\boldsymbol{J} = \boldsymbol{J}_N(\mathrm{d}\lambda_N)$ be the Jacobi matrix of order N for the measure $\mathrm{d}\lambda_N$ (cf. (2.8)) and $\boldsymbol{p}^T = [\tilde{\pi}_0, \tilde{\pi}_1, \dots, \tilde{\pi}_{N-1}]$ the vector of the N discrete orthonormal polynomials. Then, similarly as in (3.7),

$$t\boldsymbol{p}(t) = \boldsymbol{J}\boldsymbol{p}(t) + \tilde{\pi}_N(t)\boldsymbol{e}_N, \tag{4.3}$$

where $\tilde{\pi}_N(t)$ is defined as in (4.2). Note by (4.3) and (4.2) that the eigenvalues of $\boldsymbol{J}$ are the knots $t_1, t_2, \dots, t_N$. Thus, if $\boldsymbol{P} = [\boldsymbol{p}(t_1), \boldsymbol{p}(t_2), \dots, \boldsymbol{p}(t_N)]$, then

$$\boldsymbol{J}\boldsymbol{P} = \boldsymbol{P}\boldsymbol{\Lambda}, \quad \boldsymbol{\Lambda} = \mathrm{diag}(t_1, t_2, \dots, t_N). \tag{4.4}$$

As a consequence of dual orthogonality (cf. [41, §2.4.6]), one has

$$\boldsymbol{p}^T(t_k)\boldsymbol{p}(t_k) = w_k^{-2}, \quad k = 1, 2, \dots, N,$$

so that $w_k \boldsymbol{p}(t_k)$ are the normalized eigenvectors of $\boldsymbol{J}$. Thus, if

$$\boldsymbol{D} = \mathrm{diag}(w_1, w_2, \dots, w_N), \tag{4.5}$$

ETNA
Kent State University
etna@mcs.kent.edu

the matrix $\boldsymbol{PD}$ is orthogonal, and one has

$$\boldsymbol{P}^T\boldsymbol{P} = \boldsymbol{D}^{-2}, \quad \boldsymbol{PD}^2\boldsymbol{P}^T = \boldsymbol{I}. \tag{4.6}$$

Finally,

$$\boldsymbol{e}_1^T\boldsymbol{PD} = [w_1, w_2, \ldots, w_N]\tilde{\pi}_0. \tag{4.7}$$

Now let $\boldsymbol{t} = [t_1, t_2, \ldots, t_N]^T$, $\boldsymbol{y} = [y_1, y_2, \ldots, y_N]^T$, and let $q(t) = \boldsymbol{p}^T(t)\boldsymbol{c}$ be any polynomial of degree $N-1$ with coefficients $\boldsymbol{c} = [c_0, c_1, \ldots, c_{N-1}]^T$ in the basis of the orthonormal polynomials. One checks that $q(\boldsymbol{t}) = \boldsymbol{P}^T\boldsymbol{c}$. In terms of the Euclidean vector norm $\|\cdot\| = \|\cdot\|_{\mathbb{R}^N}$, the squared error in Problem P_N for the polynomial q, in view of (4.5), can be written as

$$\begin{aligned} \|y - q\|^2_{\mathrm{d}\lambda_N} &= \|\boldsymbol{D}\,(\boldsymbol{y} - q(\boldsymbol{t}))\,\|^2 \\ &= \|\boldsymbol{D}(\boldsymbol{y} - \boldsymbol{P}^T\boldsymbol{c})\|^2 = \|\boldsymbol{PD}\cdot\boldsymbol{D}(\boldsymbol{y} - \boldsymbol{P}^T\boldsymbol{c})\|^2 \\ &= \|\boldsymbol{PD}^2\boldsymbol{y} - \boldsymbol{c}\|^2, \end{aligned} \tag{4.8}$$

where the orthogonality of $\boldsymbol{PD}$ and the second relation in (4.6) have been used in the last two equations. Choosing $\boldsymbol{c} = \boldsymbol{PD}^2\boldsymbol{y}$ drives the error to zero and yields the interpolation polynomial of degree $N-1$. The solution of the least squares problem P_N, on the other hand, requires $\boldsymbol{c} = \begin{bmatrix} \boldsymbol{c}_n \\ \boldsymbol{0} \end{bmatrix}$, where $\boldsymbol{c}_n = [c_0, c_1, \ldots, c_{n-1}]^T$, and by (4.8) is equal to

$$q^0(t) = \boldsymbol{p}^T(t)\begin{bmatrix} \boldsymbol{c}_n \\ \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{c}_n = \boldsymbol{P}_{[1:n]}\boldsymbol{D}^2\boldsymbol{y}. \tag{4.9}$$

Here, $\boldsymbol{P}_{[1:n]}$ is the matrix formed with the first n rows of $\boldsymbol{P}$.

4.2. Updating and downdating the least squares solution. Suppose we adjoin to the N data points considered in §4.1 an additional point (t_{N+1}, y_{N+1}) and give it the weight w^2_{N+1}. How can the solution of the least squares problem P_N for the original N data points be used to obtain the solution of the least squares problem for the augmented set of $N+1$ data points? This is the problem of *updating* the least squares solution. There is an analogous problem of *downdating* whereby a single data point is deleted. An interesting treatment of these problems by matrix methods is given in [12].

We discuss here updating techniques only and refer to the cited reference for similar downdating techniques. In essence, the problem of updating can be considered as solved once we have constructed the Jacobi matrix $\boldsymbol{J}_{\mathrm{up}} = \boldsymbol{J}_{N+1}(\mathrm{d}\lambda_{N+1})$ of order $N+1$ for the augmented measure $\mathrm{d}\lambda_{N+1}$ from the Jacobi matrix $\boldsymbol{J} = \boldsymbol{J}_N(\mathrm{d}\lambda_N)$ for the original measure $\mathrm{d}\lambda_N$, the inner product for $\mathrm{d}\lambda_{N+1}$ being

$$(u, v)_{\mathrm{d}\lambda_{N+1}} = \int_{\mathbb{R}} u(t)v(t)\mathrm{d}\lambda_{N+1}(t) := \sum_{k=1}^{N+1} w_k^2 u(t_k)v(t_k). \tag{4.10}$$

Let (cf. (3.27))

$$\beta_0 = \int_{\mathbb{R}} \mathrm{d}\lambda_N(t), \quad \beta_{0,\mathrm{up}} = \int_{\mathbb{R}} \mathrm{d}\lambda_{N+1}(t), \tag{4.11}$$

so that

$$\tilde{\pi}_0 = 1/\sqrt{\beta_0}, \quad \tilde{\pi}_{0,\mathrm{up}} = 1/\sqrt{\beta_{0,\mathrm{up}}}. \tag{4.12}$$

There is a unique orthogonal matrix Q_{N+1} of order $N+1$ whose first row is prescribed to be

$$e_1^T \boldsymbol{Q}_{N+1} = \frac{1}{\sqrt{\beta_{0,\mathrm{up}}}}(\sqrt{\beta_0}\boldsymbol{e}_1^T + w_{N+1}\boldsymbol{e}_{N+1}^T) \tag{4.13}$$

and which accomplishes a similarity transformation of the matrix $\begin{bmatrix} \boldsymbol{J} & \boldsymbol{0} \\ \boldsymbol{0}^T & t_{N+1} \end{bmatrix}$ to tridiagonal form,

$$\boldsymbol{Q}_{N+1}\begin{bmatrix} \boldsymbol{J} & \boldsymbol{0} \\ \boldsymbol{0}^T & t_{N+1} \end{bmatrix}\boldsymbol{Q}_{N+1}^T = \boldsymbol{T}_{N+1}, \quad \boldsymbol{T}_{N+1} \ \text{tridiagonal} \tag{4.14}$$

(cf. [42, p. 113, (7-2-2)]). We claim that

$$\boldsymbol{J}_{\mathrm{up}} = \boldsymbol{T}_{N+1}. \tag{4.15}$$

To see this, recall that, with $\boldsymbol{Q} = \boldsymbol{P}\boldsymbol{D}$ (orthogonal), Eq. (4.4) implies $\boldsymbol{J} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^{-1} = \boldsymbol{Q}\boldsymbol{D}^{-1}\boldsymbol{\Lambda}\boldsymbol{D}\boldsymbol{Q}^{-1}$, hence

$$\boldsymbol{J} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T. \tag{4.16}$$

By (4.7), in view of the first relation in (4.12), there holds

$$\sqrt{\beta_0}\boldsymbol{e}_1^T\boldsymbol{Q} = [w_1, w_2, \ldots, w_N]. \tag{4.17}$$

The analogous relations for the augmented problem are

$$\boldsymbol{J}_{\mathrm{up}} = \boldsymbol{Q}_{\mathrm{up}}\boldsymbol{\Lambda}_{\mathrm{up}}\boldsymbol{Q}_{\mathrm{up}}^T, \quad \boldsymbol{\Lambda}_{\mathrm{up}} = \begin{bmatrix} \boldsymbol{\Lambda} & \boldsymbol{0} \\ \boldsymbol{0}^T & t_{N+1} \end{bmatrix}$$

and

$$\sqrt{\beta_{0,\mathrm{up}}}\boldsymbol{e}_1^T\boldsymbol{Q}_{\mathrm{up}} = [w_1, w_2, \ldots, w_N, w_{N+1}], \tag{4.18}$$

where $\boldsymbol{e}_1$ now has dimension $N+1$. Define

$$\boldsymbol{Q}^* = \boldsymbol{Q}_{\mathrm{up}}\begin{bmatrix} \boldsymbol{Q}^T & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 \end{bmatrix}.$$

Then

$$\begin{aligned}\boldsymbol{Q}^*\begin{bmatrix} \boldsymbol{J} & \boldsymbol{0} \\ \boldsymbol{0}^T & t_{N+1} \end{bmatrix}\boldsymbol{Q}^{*T} &= \boldsymbol{Q}_{\mathrm{up}}\begin{bmatrix} \boldsymbol{Q}^T & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 \end{bmatrix}\begin{bmatrix} \boldsymbol{J} & \boldsymbol{0} \\ \boldsymbol{0}^T & t_{N+1} \end{bmatrix}\begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 \end{bmatrix}\boldsymbol{Q}_{\mathrm{up}}^T \\ &= \boldsymbol{Q}_{\mathrm{up}}\begin{bmatrix} \boldsymbol{Q}^T\boldsymbol{J}\boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0}^T & t_{N+1} \end{bmatrix}\boldsymbol{Q}_{\mathrm{up}}^T,\end{aligned}$$

hence, by (4.16),

$$\boldsymbol{Q}^*\begin{bmatrix} \boldsymbol{J} & \boldsymbol{0} \\ \boldsymbol{0}^T & t_{N+1} \end{bmatrix}\boldsymbol{Q}^{*T} = \boldsymbol{Q}_{\mathrm{up}}\begin{bmatrix} \boldsymbol{\Lambda} & \boldsymbol{0} \\ \boldsymbol{0}^T & t_{N+1} \end{bmatrix}\boldsymbol{Q}_{\mathrm{up}}^T = \boldsymbol{Q}_{\mathrm{up}}\boldsymbol{\Lambda}_{\mathrm{up}}\boldsymbol{Q}_{\mathrm{up}}^T = \boldsymbol{J}_{\mathrm{up}}.$$

Furthermore, using (4.18),

$$\boldsymbol{e}_1^T\boldsymbol{Q}^* = \boldsymbol{e}_1^T\boldsymbol{Q}_{\mathrm{up}}\begin{bmatrix} \boldsymbol{Q}^T & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 \end{bmatrix} = \frac{1}{\sqrt{\beta_{0,\mathrm{up}}}}[w_1, \ldots, w_N, w_{N+1}]\begin{bmatrix} \boldsymbol{Q}^T & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 \end{bmatrix},$$

ETNA
Kent State University
etna@mcs.kent.edu

which by (4.7) and the first of (4.12) becomes

$$\begin{aligned}\boldsymbol{e}_1^T\boldsymbol{Q}^* &= \frac{1}{\sqrt{\beta_{0,\mathrm{up}}}}[\sqrt{\beta_0}\boldsymbol{e}_1^T\boldsymbol{Q} \;\; w_{N+1}]\begin{bmatrix}\boldsymbol{Q}^T & \boldsymbol{0}\\ \boldsymbol{0}^T & 1\end{bmatrix}\\ &= \frac{1}{\sqrt{\beta_{0,\mathrm{up}}}}[\sqrt{\beta_0}\boldsymbol{e}_1^T \;\; w_{N+1}] = \frac{1}{\sqrt{\beta_{0,\mathrm{up}}}}[\sqrt{\beta_0}\boldsymbol{e}_1^T + w_{N+1}\boldsymbol{e}_{N+1}^T].\end{aligned}$$

Thus, $\boldsymbol{Q}^*$ satisfies exactly the properties defining $\boldsymbol{Q}_{N+1}$, showing that indeed $\boldsymbol{J}_{\mathrm{up}} = \boldsymbol{T}_{N+1}$. The analogue of (4.3),

$$t\boldsymbol{p}_{\mathrm{up}}(t) = \boldsymbol{J}_{\mathrm{up}}\boldsymbol{p}_{\mathrm{up}}(t) + \tilde{\pi}_{N+1,\mathrm{up}}(t)\boldsymbol{e}_{N+1},$$

in combination with the second relation of (4.12) can now be used to generate the new discrete orthonormal polynomials, and with them the updated least squares solution by the analogue of (4.9).

Algorithmically, the transformation (4.14) can be implemented by a sequence of appropriate Givens rotations (cf. [12, Eq. (4.7)]). The updating technique described here is not the only possible one; for others, see [*ibid.*, §§4.3–4.7].

Since the solution for the one-point least squares problem P_1 is trivially $\tilde{\pi}_0 = 1/|w_1|$, $\boldsymbol{J} = [t_1]$, $\boldsymbol{c} = [|w_1|y_1]$, one can use the updating technique to build up the least squares solutions of P_N successively for $N = 2, 3, \ldots$ without necessarily having to store the entire data set.

5. Linear algebraic systems. Many linear algebra problems that involve a symmetric positive definite matrix $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ can be related to discrete orthogonal polynomials supported on the spectrum of $\boldsymbol{A}$. This provides the link between linear algebra and analysis. It may be appropriate, at this point, to recall that the use of discrete (and other) orthogonal polynomials in the context of linear algebra has been pioneered by Stiefel [44]; see also [36, §14].

For simplicity assume that $\boldsymbol{A}$ has distinct[1] eigenvalues λ_n,

$$0 < \lambda_N < \lambda_{N-1} < \cdots < \lambda_1, \tag{5.1}$$

and denote the respective (orthonormal) eigenvectors by $\boldsymbol{v}_n$,

$$\boldsymbol{A}\boldsymbol{v}_n = \lambda_n\boldsymbol{v}_n, \quad \boldsymbol{v}_n^T\boldsymbol{v}_m = \delta_{nm}, \quad n, m = 1, 2, \ldots, N. \tag{5.2}$$

(There should be no danger of confusing these λ's with the weights of the Gauss quadrature rule in (2.2).) Thus, with $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N]$, $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_N)$, there holds

$$\boldsymbol{A}\boldsymbol{V} = \boldsymbol{V}\boldsymbol{\Lambda}, \quad \boldsymbol{\Lambda} = \boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V}. \tag{5.3}$$

Now consider a discrete measure $\mathrm{d}\rho_N$ defined by

$$\int_{\mathbb{R}_+} f(t)\mathrm{d}\rho_N(t) := \sum_{k=1}^{N}\rho_k^2 f(\lambda_k), \tag{5.4}$$

where ρ_k^2 are positive weights, and assume, temporarily, that the measure $\mathrm{d}\rho_N$ is normalized,

$$\int_{\mathbb{R}_+} \mathrm{d}\rho_N(t) = 1. \tag{5.5}$$

[1]Otherwise, some terms in (5.4) below consolidate, so that N becomes smaller.

It is possible to generate the orthonormal polynomials $\tilde{\pi}_k(\,\cdot\,;\mathrm{d}\rho_N)$, $k = 0, 1, \ldots, N-1$, resp. the associated Jacobi matrix $\boldsymbol{J}_N = \boldsymbol{J}_N(\mathrm{d}\rho_N)$, entirely by matrix-vector multiplications involving the matrix $\boldsymbol{A}$ and by an initial vector

$$\boldsymbol{h}_0 = \sum_{k=1}^{N} \rho_k \boldsymbol{v}_k, \quad \|\boldsymbol{h}_0\| = 1, \tag{5.6}$$

whose components in the basis of the normalized eigenvectors are the (positive or negative) square roots of the weights ρ_k^2. (Here and in the following, $\|\cdot\|$ denotes the Euclidean vector norm.) A method accomplishing this is the *Lanczos algorithm*, which is briefly described in §5.1. The subsequent sections give applications of this algorithm when combined with quadrature methods.

5.1. The Lanczos algorithm. Let $\boldsymbol{h}_0$ be given as in (5.6), and define $\boldsymbol{h}_{-1} = \boldsymbol{0}$. The Lanczos algorithm is defined as follows:

$$\begin{array}{l} \text{for } j = 0, 1, \ldots, N-1 \text{ do} \\ \left[\begin{array}{l} \alpha_j = \boldsymbol{h}_j^T \boldsymbol{A} \boldsymbol{h}_j \\ \tilde{\boldsymbol{h}}_{j+1} = (\boldsymbol{A} - \alpha_j \boldsymbol{I})\boldsymbol{h}_j - \gamma_j \boldsymbol{h}_{j-1} \\ \gamma_{j+1} = \|\tilde{\boldsymbol{h}}_{j+1}\| \\ \boldsymbol{h}_{j+1} = \tilde{\boldsymbol{h}}_{j+1}/\gamma_{j+1} \end{array} \right. \end{array} \tag{5.7}$$

Note that γ_0 can be arbitrary, but is often defined, in accordance with (5.5), by $\gamma_0 = 1$, or, in accordance with (5.10) below, by $\gamma_0 = \beta_0$.

The vectors $\boldsymbol{h}_j$ so generated are orthonormal, as one checks by induction, and it is evident from (5.7) that $\{\boldsymbol{h}_j\}_{j=0}^n$, $n < N$, forms an orthonormal basis for the *Krylov space*

$$\mathcal{K}_n(\boldsymbol{A}, \boldsymbol{h}_0) = \mathrm{span}(\boldsymbol{h}_0, \boldsymbol{A}\boldsymbol{h}_0, \ldots, \boldsymbol{A}^n \boldsymbol{h}_0).$$

One also verifies by induction that

$$\boldsymbol{h}_j = p_j(\boldsymbol{A})\boldsymbol{h}_0, \tag{5.8}$$

where p_j is a polynomial of degree j satisfying the three-term recurrence relation

$$\begin{gathered} \gamma_{j+1} p_{j+1}(\lambda) = (\lambda - \alpha_j) p_j(\lambda) - \gamma_j p_{j-1}(\lambda), \\ j = 0, 1, \ldots, N-1, \\ p_{-1}(\lambda) = 0, \quad p_0(\lambda) = 1. \end{gathered} \tag{5.9}$$

We claim that $p_k(\,\cdot\,) = \tilde{\pi}_k(\,\cdot\,;\mathrm{d}\rho_N)$. Indeed, from the second relation in (5.3) one has

$$p_n(\boldsymbol{\Lambda}) = \boldsymbol{V}^T p_n(\boldsymbol{A})\boldsymbol{V},$$

hence, by (5.8),

$$\boldsymbol{h}_n = \boldsymbol{V} p_n(\boldsymbol{\Lambda})\boldsymbol{V}^T \boldsymbol{h}_0.$$

Orthonormality $\boldsymbol{h}_n^T \boldsymbol{h}_m = \delta_{nm}$ of the Lanczos vectors $\boldsymbol{h}_j$ then yields

$$\boldsymbol{h}_0^T \boldsymbol{V} p_n(\boldsymbol{\Lambda})\boldsymbol{V}^T \boldsymbol{V} p_m(\boldsymbol{\Lambda})\boldsymbol{V}^T \boldsymbol{h}_0 = \boldsymbol{h}_0^T \boldsymbol{V} p_n(\boldsymbol{\Lambda}) p_m(\boldsymbol{\Lambda})\boldsymbol{V}^T \boldsymbol{h}_0 = \delta_{nm},$$

ETNA
Kent State University
etna@mcs.kent.edu

which, since $\boldsymbol{V}^T\boldsymbol{h}_0 = \sum_{k=1}^N \rho_k \boldsymbol{e}_k$ by (5.6), implies

$$\sum_{k,\ell=1}^{N} \rho_k \boldsymbol{e}_k^T \operatorname{diag}\left(p_n(\lambda_1)p_m(\lambda_1), \ldots, p_n(\lambda_N)p_m(\lambda_N)\right) \rho_\ell \boldsymbol{e}_\ell$$
$$= \sum_{k,\ell=1}^{N} \rho_k\rho_\ell \boldsymbol{e}_k^T p_n(\lambda_\ell)p_m(\lambda_\ell)\boldsymbol{e}_\ell = \sum_{k=1}^{N} \rho_k^2 p_n(\lambda_k)p_m(\lambda_k) = \delta_{nm},$$

as claimed.

The recurrence relation (5.9), therefore, must be identical with the one in (3.2), i.e., $\gamma_j = \sqrt{\beta_j}$.

If the measure $\mathrm{d}\rho_N$ is not normalized and, as in (3.27), one puts

$$\beta_0 = \int_{\mathbb{R}_+} \mathrm{d}\rho_N(t), \tag{5.10}$$

then the recurrence relation still holds, except that one must define $p_0(\lambda) = 1/\sqrt{\beta_0}$.

5.2. Bounds for matrix functionals. Given $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ positive definite and f a function analytic on an interval containing the spectrum of $\boldsymbol{A}$, the problem to be considered is finding lower and upper bounds for the bilinear form

$$\boldsymbol{u}^T f(\boldsymbol{A})\boldsymbol{v}, \tag{5.11}$$

where $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^N$ are given vectors. The solution of this problem has many applications; some will be discussed in subsequent sections. For applications to constrained least squares problems for matrices, see [29], and [7] for applications to the evaluation of suitable regularization parameters in Tikhonov regularization. The case $f(t) = (\lambda - t)^{-1}$ with λ outside the spectrum of $\boldsymbol{A}$ is important in physical chemistry and solid state physics applications; for references, see [32, §1].

Let first $\boldsymbol{u} = \boldsymbol{v}$. With $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N]$ and $\boldsymbol{\Lambda}$ as defined in (5.2), (5.3), we let

$$\boldsymbol{u} = \sum_{k=1}^{N} \rho_k \boldsymbol{v}_k \tag{5.12}$$

and for simplicity assume $\rho_k \neq 0$ for all k. Then $\boldsymbol{u} = \boldsymbol{V}\boldsymbol{\rho}$, $\boldsymbol{\rho} = [\rho_1, \rho_2, \ldots, \rho_N]^T$, and $f(\boldsymbol{A}) = \boldsymbol{V} f(\boldsymbol{\Lambda})\boldsymbol{V}^T$. Therefore,

$$\boldsymbol{u}^T f(\boldsymbol{A})\boldsymbol{u} = \boldsymbol{\rho}^T \boldsymbol{V}^T \boldsymbol{V} f(\boldsymbol{\Lambda})\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{\rho} = \boldsymbol{\rho}^T f(\boldsymbol{\Lambda})\boldsymbol{\rho} = \sum_{k=1}^{N} \rho_k^2 f(\lambda_k),$$

that is,

$$\boldsymbol{u}^T f(\boldsymbol{A})\boldsymbol{u} = \int_{\mathbb{R}_+} f(t)\mathrm{d}\rho_N(t), \tag{5.13}$$

where $\mathrm{d}\rho_N$ is the discrete measure defined in (5.4). The desired bounds can be obtained by applying Gauss, Gauss-Radau, or Gauss-Lobatto quadrature to the integral in (5.13), provided the appropriate derivative of f has constant sign (cf. §§2.1,2.2). The Lanczos algorithm (cf. §5.1) applied with $\boldsymbol{h}_0 = \boldsymbol{u}/\|\boldsymbol{u}\|$ furnishes the necessary (discrete) orthogonal polynomials, resp. their recursion coefficients. For Gauss formulae, the quality of the bounds, even

ETNA
Kent State University
etna@mcs.kent.edu

when no specific information is known about the sign of derivatives of f, can be estimated in terms of the absolute values of these derivatives and the quantities $\gamma_j^2 = \beta_j$ generated during the Lanczos process (cf. [6]). One simply makes use of (2.11).

The case $\boldsymbol{u} \neq \boldsymbol{v}$ can be handled by using the polarization identity $\boldsymbol{u}^T f(\boldsymbol{A})\boldsymbol{v} = \frac{1}{4}(\boldsymbol{p}^T f(\boldsymbol{A})\boldsymbol{p} - \boldsymbol{q}^T f(\boldsymbol{A})\boldsymbol{q})$ where $\boldsymbol{p} = \boldsymbol{u} + \boldsymbol{v}$, $\boldsymbol{q} = \boldsymbol{u} - \boldsymbol{v}$ (cf. [2, §3.1.2], [3, p. 426], or, for a similar identity, [32, Eq. (3)]) and applying appropriate bounds to the first and second term of the identity. Alternatively, a "nonsymmetric" Lanczos process can be applied in conjunction with Gauss-Radau quadrature [30].

For the important function $f(t) = t^{-1}$ (see, e.g., (5.19) or (5.22) below), the case of an arbitrary nonsingular matrix $\boldsymbol{A}$ can be reduced to the case of a symmetric positive definite matrix by noting that

$$\boldsymbol{u}^T \boldsymbol{A}^{-1}\boldsymbol{v} = \boldsymbol{u}^T(\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{w}, \quad \boldsymbol{w} = \boldsymbol{A}^T\boldsymbol{v} \tag{5.14}$$

(cf. [2, §3.2], [3, p. 427]).

5.3. Error bounds. We consider now the system of linear algebraic equations

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{5.15}$$

with $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ symmetric and positive definite. Given any approximation $\boldsymbol{x}^*$ to the exact solution $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$, the object is to estimate the error $\boldsymbol{x} - \boldsymbol{x}^*$ in some norm. We begin with using the Euclidean vector norm $\|\cdot\|$.

Let $\boldsymbol{r}$ be the *residual* of $\boldsymbol{x}^*$,

$$\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^*. \tag{5.16}$$

Since $\boldsymbol{x} - \boldsymbol{x}^* = \boldsymbol{A}^{-1}\boldsymbol{r}$, we have

$$\|\boldsymbol{x} - \boldsymbol{x}^*\|^2 = \boldsymbol{r}^T\boldsymbol{A}^{-2}\boldsymbol{r}, \tag{5.17}$$

which is (5.11) with $\boldsymbol{u} = \boldsymbol{v} = \boldsymbol{r}$ and $f(t) = t^{-2}$. Here the derivatives are $f^{(2n)}(t) = (2n+1)!\,t^{-(2n+2)}$, $f^{(2n+1)}(t) = -(2n+2)!\,t^{-(2n+3)}$, so that

$$f^{(2n)}(t) > 0, \quad f^{(2n+1)}(t) < 0 \ \text{ for } t \in \mathbb{R}_+. \tag{5.18}$$

The n-point Gauss formula (with $n < N$) applied to the integral in (5.13) (with $f(t) = t^{-2}$) thus produces a lower bound for the squared error (5.17). If the spectrum of $\boldsymbol{A}$ can be enclosed in an interval $[a, b]$, $0 < a < b$, then the "left-sided" $(n+1)$-point Gauss-Radau formula yields an upper bound, and the "right-sided" formula a lower bound for (5.17). The Lanczos algorithm applied with $\boldsymbol{h}_0 = \boldsymbol{r}/\|\boldsymbol{r}\|$ yields the recursion coefficients for the orthogonal polynomials required for generating these quadrature rules.

If instead of the Euclidean norm one takes the A-norm $\|\boldsymbol{u}\|_A^2 = \boldsymbol{u}^T\boldsymbol{A}\boldsymbol{u}$ (cf. [31]), then

$$\|\boldsymbol{x} - \boldsymbol{x}^*\|_A^2 = \boldsymbol{r}^T\boldsymbol{A}^{-1}\boldsymbol{r}, \tag{5.19}$$

which is (5.11) with $\boldsymbol{u} = \boldsymbol{v} = \boldsymbol{r}$ and $f(t) = t^{-1}$. Since this function satisfies the same inequalities as in (5.18), the Gauss and Gauss-Radau formulae applied to the integral in (5.13) (with $f(t) = t^{-1}$) produce the same kind of bounds as in the case of the Euclidean norm. The difference between the N-point and n-point Gauss quadrature approximation equals $\|\boldsymbol{x} - \boldsymbol{x}_n\|_A^2/\|\boldsymbol{r}\|^2$, where $\boldsymbol{x}_n$ is the nth iterate of the conjugate gradient method started with $\boldsymbol{r}$ (cf. [32, Eq. (50)]). The conjugate gradient method, in fact, can be used not only as an alternative to the Lanczos algorithm to generate the recursion coefficients of the orthogonal

ETNA
Kent State University
etna@mcs.kent.edu

polynomials, but also to improve the approximation $\boldsymbol{x}^*$. The A-norm of the improved approximation can then be estimated from below and above (see [11, §5]). For analogous error estimates in the Euclidean norm, see [9].

The idea of using Gauss-Radau quadratures in combination with (5.18) to get error bounds for linear systems goes back to Dahlquist, Eisenstat, and Golub [10]. They also suggest a procedure based on linear programming when all eigenvalues are known (cf. [10, §2]). This requires knowledge of the moments μ_m of $\mathrm{d}\rho_N$, which by (5.13) are given by

$$\mu_m = \int_{\mathbb{R}_+} t^m \mathrm{d}\rho_N(t) = \boldsymbol{r}^T \boldsymbol{A}^m \boldsymbol{r}.$$

Thus, computing the first $2n+1$ moments $\mu_0, \mu_1, \ldots, \mu_{2n}$ amounts to generating the *Krylov sequence* $\boldsymbol{r}, \boldsymbol{A}\boldsymbol{r}, \ldots, \boldsymbol{A}^{2n}\boldsymbol{r}$ and computing the inner products of its members with $\boldsymbol{r}$. In view of

$$\|\boldsymbol{x} - \boldsymbol{x}^*\|^2 = \int_{\mathbb{R}_+} t^{-2} \mathrm{d}\rho_N(t) = \sum_{k=1}^{N} \rho_k^2 \lambda_k^{-2},$$

an upper bound can be found by solving the linear programming problem

$$\text{max!} \ \sum_{k=1}^{N} \gamma_k \lambda_k^{-2} \tag{5.20}$$

subject to the constraints

$$\begin{gathered} \sum_{k=1}^{N} \gamma_k \lambda_k^m = \mu_m, \quad m = 0, 1, \ldots, 2n, \\ \gamma_k \geq 0, \quad k = 1, 2, \ldots, N. \end{gathered} \tag{5.21}$$

Here, n can be any integer $< N$. A lower bound can similarly be obtained by replacing "max" in (5.20) by "min". The same procedure, with λ_k^{-2} in (5.20) replaced by λ_k^{-1}, works for the A-norm.

The ideas outlined above, and still other ideas from the theory of moments, are applied in [10] to obtain upper and lower bounds for the errors in the Jacobi iterative method. Bounds for matrix moments $\mu_m = \boldsymbol{r}^T \boldsymbol{A}^m \boldsymbol{r}$ are similarly obtained in [24].

5.4. The diagonal elements of $\boldsymbol{A}^{-1}$. Given $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ positive definite, the problem is to find bounds for the diagonal elements $(\boldsymbol{A}^{-1})_{ii}$ of $\boldsymbol{A}^{-1}$, $i = 1, 2, \ldots, N$. Here,

$$(\boldsymbol{A}^{-1})_{ii} = \boldsymbol{e}_i^T \boldsymbol{A}^{-1} \boldsymbol{e}_i, \tag{5.22}$$

where $\boldsymbol{e}_i$ is the ith canonical basis vector. This is (5.11) with $\boldsymbol{u} = \boldsymbol{v} = \boldsymbol{e}_i$ and $f(t) = t^{-1}$. As before, f satisfies the inequalities (5.18).

5.4.1. Lower bound from Gauss quadrature. By virtue of (2.4) and the first of (5.18), the n-point Gauss quadrature sum (cf. (2.10), where $\mu_0 = 1$) yields a lower bound for the integral, i.e.,

$$(\boldsymbol{A}^{-1})_{ii} = \int_{\mathbb{R}_+} t^{-1} \mathrm{d}\rho_N(t) > \boldsymbol{e}_1^T \boldsymbol{J}_n^{-1} \boldsymbol{e}_1, \quad \boldsymbol{e}_1^T = [1, 0, \ldots, 0] \in \mathbb{R}^n, \tag{5.23}$$

ETNA
Kent State University
etna@mcs.kent.edu

where $\boldsymbol{J}_n = \boldsymbol{J}_n(\mathrm{d}\rho_N)$. Consider $n = 2$; we apply two steps of the Lanczos algorithm with $\boldsymbol{h}_0 = \boldsymbol{e}_i$ to generate

$$\boldsymbol{J}_2 = \begin{bmatrix} \alpha_0 & \gamma_1 \\ \gamma_1 & \alpha_1 \end{bmatrix}.$$

According to (5.7) we have

$$\begin{aligned}
\alpha_0 &= a_{ii}, \\
\tilde{\boldsymbol{h}}_1 &= (\boldsymbol{A} - \alpha_0 \boldsymbol{I})\boldsymbol{e}_i = [a_{1i}, \ldots, a_{i-1,i}, 0, a_{i+1,i}, \ldots, a_{Ni}]^T, \\
\gamma_1 &= \sqrt{\sum_{k\neq i} a_{ki}^2} =: s_i, \\
\boldsymbol{h}_1 &= \tilde{\boldsymbol{h}}_1 / s_i, \\
\alpha_1 &= \frac{1}{s_i^2} \tilde{\boldsymbol{h}}_1^T \boldsymbol{A} \tilde{\boldsymbol{h}}_1 = \frac{1}{s_i^2} \sum_{k\neq i} \sum_{\ell\neq i} a_{k\ell} a_{ki} a_{\ell i}.
\end{aligned} \tag{5.24}$$

Since

$$\boldsymbol{J}_2^{-1} = \frac{1}{\alpha_0\alpha_1 - \gamma_1^2} \begin{bmatrix} \alpha_1 & -\gamma_1 \\ -\gamma_1 & \alpha_0 \end{bmatrix},$$

one has

$$\boldsymbol{e}_1^T \boldsymbol{J}_2^{-1} \boldsymbol{e}_1 = \frac{\alpha_1}{\alpha_0\alpha_1 - \gamma_1^2}, \tag{5.25}$$

and therefore, by (5.23) and (5.24),

$$(\boldsymbol{A}^{-1})_{ii} > \frac{\displaystyle\sum_{k\neq i}\sum_{\ell\neq i} a_{k\ell} a_{ki} a_{\ell i}}{\displaystyle a_{ii} \sum_{k\neq i}\sum_{\ell\neq i} a_{k\ell} a_{ki} a_{\ell i} - \left(\sum_{k\neq i} a_{ki}^2\right)^2}. \tag{5.26}$$

It should be noted that this bound, in contrast to those given below in §§5.4.2–5.4.3, does not require any information about the spectrum of $\boldsymbol{A}$.

5.4.2. Upper and lower bounds from Gauss-Radau quadrature. If the spectrum of $\boldsymbol{A}$ can be enclosed in the interval $[a, b]$, $0 < a < b$, then by the second of (5.18) and the first of (2.17) (with the inequality reversed) the "left-sided" $(n+1)$-point Gauss-Radau quadrature sum in (2.12) yields an upper bound, and similarly the "right-sided" quadrature sum in (2.16) a lower bound for the integral. Taking $n = 1$ in (2.14), (2.15), one gets

$$\boldsymbol{J}_2^{R,a}(\mathrm{d}\rho_N) = \begin{bmatrix} \alpha_0 & \gamma_1 \\ \gamma_1 & \alpha_1^R \end{bmatrix}, \quad \alpha_1^R = a + \frac{\gamma_1^2}{\alpha_0 - a},$$

where $\alpha_0 = a_{ii}$, $\gamma_1 = s_i$ from (5.24). Replacing here a by b yields $\boldsymbol{J}_2^{R,b}(\mathrm{d}\rho_N)$. From (5.25), where α_1 is replaced by α_1^R, a simple computation then gives

$$\frac{a_{ii} - b + s_i^2/b}{a_{ii}^2 - a_{ii}b + s_i^2} < (\boldsymbol{A}^{-1})_{ii} < \frac{a_{ii} - a + s_i^2/a}{a_{ii}^2 - a_{ii}a + s_i^2}. \tag{5.27}$$

ETNA
Kent State University
etna@mcs.kent.edu

5.4.3. Upper bound from Gauss-Lobatto quadrature. The $(n+2)$-point Gauss-Lobatto quadrature sum in (2.19), on account of (2.21) and the first of (5.18) (with n replaced by $n+1$), yields an upper bound for the integral. Taking $n=0$ in (2.22), one gets

$$\boldsymbol{J}_2^L(\mathrm{d}\rho_N) = \begin{bmatrix} \alpha_0 & \gamma_1^L \\ \gamma_1^L & \alpha_1^L \end{bmatrix},$$

where by (2.23) the quantities α_1^L and γ_1^L solve the 2×2 system

$$\begin{bmatrix} a-\alpha_0 & 1 \\ b-\alpha_0 & 1 \end{bmatrix} \begin{bmatrix} \alpha_1^L \\ (\gamma_1^L)^2 \end{bmatrix} = \begin{bmatrix} a(a-\alpha_0) \\ b(b-\alpha_0) \end{bmatrix}, \quad \alpha_0 = a_{ii}.$$

Carrying out the solution and using (5.25) with α_1, γ_1 replaced by α_1^L, γ_1^L, yields

$$(\boldsymbol{A}^{-1})_{ii} < \frac{a+b-a_{ii}}{ab}. \tag{5.28}$$

The results in §§5.4.1–5.4.3 are from [30, Thm. 5.1]. When $n>2$, the quadrature sum $\boldsymbol{e}_1^T\boldsymbol{J}_n^{-1}\boldsymbol{e}_1$ can be computed for all three quadrature rules in terms of quantities generated during the course of the Lanczos algorithm; see [30, Thm. 5.3]. For an application to Vičsek fractal Hamiltonian matrices, see [25].

5.4.4. The trace of $\boldsymbol{A}^{-1}$ and the determinant of $\boldsymbol{A}$. In principle, each method described in §§5.4.1–5.4.3 can be used to estimate the trace

$$\mathrm{tr}\,(\boldsymbol{A}^{-1}) = \sum_{i=1}^{N} (\boldsymbol{A}^{-1})_{ii} \tag{5.29}$$

of $\boldsymbol{A}^{-1}$ by applying the method to each term in the sum of (5.29), hence N times. For large sparse matrices there are, however, more efficient estimation procedures based on sampling and a Monte Carlo approach (cf. [2, §4]).

Alternatively, we may note that (cf. [1])

$$\mathrm{tr}\,(\boldsymbol{A}^{-1}) = \sum_{k=1}^{N} \lambda_k^{-1} = \int_{\mathbb{R}_+} t^{-1}\mathrm{d}\rho_N(t), \tag{5.30}$$

where $\mathrm{d}\rho_N$ is the discrete measure (5.4) with $\rho_k = 1$, $k=1,2,\ldots,N$. As in §5.4.2, we may then apply Gauss-Radau quadratures on an interval $[a,b]$ containing all eigenvalues λ_k to get lower and upper bounds. The only difference is that now $\mathrm{d}\rho_N$ is no longer normalized, in fact

$$\mu_0 = \int_{\mathbb{R}_+} \mathrm{d}\rho_N(t) = N, \tag{5.31}$$

and the Lanczos algorithm, in accordance with (5.6), is to be started with

$$\boldsymbol{h}_0 = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} \boldsymbol{v}_k.$$

Observing that

$$\begin{aligned} \mu_1 &= \int_{\mathbb{R}_+} t\,\mathrm{d}\rho_N(t) = \sum_{k=1}^{N} \lambda_k = \sum_{i=1}^{N} a_{ii} = \mathrm{tr}\,(\boldsymbol{A}), \\ \mu_2 &= \int_{\mathbb{R}_+} t^2\mathrm{d}\rho_N(t) = \sum_{k=1}^{N} \lambda_k^2 = \sum_{i,j=1}^{N} a_{ij}^2 = \|\boldsymbol{A}\|_F^2, \end{aligned} \tag{5.32}$$

from (5.7) one gets

$$\alpha_0 = \boldsymbol{h}_0^T \boldsymbol{A} \boldsymbol{h}_0 = \frac{1}{N} \sum_{k=1}^{N} \boldsymbol{v}_k^T \sum_{\ell=1}^{N} \boldsymbol{A} \boldsymbol{v}_\ell = \frac{1}{N} \sum_{k=1}^{N} \lambda_k,$$

that is,

$$\alpha_0 = \frac{1}{N}\, \mu_1. \tag{5.33}$$

Furthermore,

$$\tilde{\boldsymbol{h}}_1 = (\boldsymbol{A} - \alpha_0 \boldsymbol{I}) \boldsymbol{h}_0 = \frac{1}{\sqrt{N}} (\boldsymbol{A} - \alpha_0 \boldsymbol{I}) \sum_{k=1}^{N} \boldsymbol{v}_k = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} (\lambda_k - \alpha_0) \boldsymbol{v}_k,$$

and

$$\gamma_1^2 = \tilde{\boldsymbol{h}}_1^T \tilde{\boldsymbol{h}}_1 = \frac{1}{N} \sum_{k=1}^{N} (\lambda_k - \alpha_0) \boldsymbol{v}_k^T \sum_{\ell=1}^{N} (\lambda_\ell - \alpha_0) \boldsymbol{v}_\ell.$$

An elementary calculation yields

$$\gamma_1^2 = \frac{1}{N} \left(\mu_2 - \frac{1}{N}\, \mu_1^2\right). \tag{5.34}$$

The rest of the calculation is the same as in §5.4.2, except that, by (5.31), one has to include the factor $\mu_0 = N$ in (5.25). The result is

$$\frac{1}{b} \left(1 - \frac{\frac{1}{N}\mu_1^2 + N b^2}{\mu_2 - b\mu_1}\right) < \frac{1}{N}\, \mathrm{tr}\,(\boldsymbol{A}^{-1}) < \frac{1}{a} \left(1 - \frac{\frac{1}{N}\mu_1^2 + N a^2}{\mu_2 - a\mu_1}\right), \tag{5.35}$$

with μ_1, μ_2 given by (5.32). The same inequalities, in a different form, are derived in [1, Eq. (9)] by means of difference equations.

As far as the determinant $\det \boldsymbol{A}$ is concerned, we note that the trace is invariant to similarity transformations, so that by (5.3)

$$\mathrm{tr}\,(\ln \boldsymbol{A}) = \mathrm{tr}\,(\boldsymbol{V} \ln \boldsymbol{\Lambda} \boldsymbol{V}^T) = \mathrm{tr}\,(\ln \boldsymbol{\Lambda}) = \sum_{k=1}^{N} \ln \lambda_k = \ln \prod_{k=1}^{N} \lambda_k.$$

Since $\det \boldsymbol{A} = \prod_k \lambda_k$, this yields

$$\det \boldsymbol{A} = \exp(\mathrm{tr}\,(\ln \boldsymbol{A})). \tag{5.36}$$

Here, the trace of $\ln \boldsymbol{A}$ can be estimated as described for $\boldsymbol{A}^{-1}$, with the function $f(t) = t^{-1}$ replaced by $f(t) = \ln t$. This latter function has derivatives whose signs are opposite to those in (5.18), which gives rise to bounds whose types are opposite to those obtained in §§5.4.1–5.4.3.

Note that in place of $\boldsymbol{J}_2^{-1}$ in the quadrature sum (5.25), we now require $\ln \boldsymbol{J}_2$. This can be defined by linear interpolation at the eigenvalues $0 < \kappa_2 < \kappa_1$ of $\boldsymbol{J}_2$ (see, e.g., [18, Ch. 5]),

$$\ln \boldsymbol{J}_2 = \frac{1}{\kappa_1 - \kappa_2} [(\boldsymbol{J}_2 - \kappa_2 \boldsymbol{I}) \ln \kappa_1 + (\kappa_1 \boldsymbol{I} - \boldsymbol{J}_2) \ln \kappa_2].$$

In particular, therefore,

$$\boldsymbol{e}_1^T \ln \boldsymbol{J}_2 \boldsymbol{e}_1 = \frac{1}{\kappa_1 - \kappa_2} [(\alpha_0 - \kappa_2) \ln \kappa_1 + (\kappa_1 - \alpha_0) \ln \kappa_2], \tag{5.37}$$

where α_0 is given by the first of (5.24) resp. by (5.33).

ETNA
Kent State University
etna@mcs.kent.edu

5.5. Iterative methods. Consider again the system (5.15) with $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ symmetric and positive definite. Based on the splitting $\boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N}$, where $\boldsymbol{M}$ and $\boldsymbol{N}$ are symmetric and $\boldsymbol{M}$ positive definite, a large class of iterative methods for solving (5.15) is given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_{k-1} + \omega_{k+1}(\delta \boldsymbol{z}_k + \boldsymbol{x}_k - \boldsymbol{x}_{k-1}), \quad k = 0, 1, 2, \ldots, \quad \boldsymbol{x}_{-1} = \boldsymbol{0}, \tag{5.38}$$

where

$$\boldsymbol{M}\boldsymbol{z}_k = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_k. \tag{5.39}$$

In practice, $\boldsymbol{M}$ is chosen such that linear systems with this matrix as coefficient matrix can be easily solved. Depending on the choice of parameters, the iteration (5.38) includes such methods as the conjugate gradient, the Richardson second-order, and the Chebyshev semi-iterative method. For optimizing the speed of convergence of the two latter methods, it is important to have good estimates of the smallest and largest eigenvalues of $\boldsymbol{M}^{-1}\boldsymbol{N}$. Such estimates can be found via certain discrete orthogonal polynomials and the modified Chebyshev algorithm (cf. §3.4.2) generating them; see [28].

To analyze the speed of convergence of the iteration, there is no loss of generality in assuming, as we do, that $\boldsymbol{b} = \boldsymbol{0}$, and thus considering convergence of $\boldsymbol{x}_k$ resp. $\boldsymbol{z}_k$ to the zero vector.

Substituting $\boldsymbol{x}_k = -\boldsymbol{A}^{-1}\boldsymbol{M}\boldsymbol{z}_k$ obtained from (5.39) into (5.38) yields

$$\boldsymbol{z}_{k+1} = \omega_{k+1}\boldsymbol{B}\boldsymbol{z}_k + (1-\omega_{k+1})\boldsymbol{z}_{k-1}, \quad \boldsymbol{z}_{-1} = \boldsymbol{0}, \tag{5.40}$$

where

$$\boldsymbol{B} = \boldsymbol{I} - \delta\boldsymbol{M}^{-1}\boldsymbol{A}. \tag{5.41}$$

Since $\boldsymbol{B} = \boldsymbol{I} - \delta\boldsymbol{M}^{-1}(\boldsymbol{M}-\boldsymbol{N}) = (1-\delta)\boldsymbol{I} + \delta\boldsymbol{M}^{-1}\boldsymbol{N}$, the eigenvalues ν_n of $\boldsymbol{M}^{-1}\boldsymbol{N}$ are related to the eigenvalues λ_n of $\boldsymbol{B}$ by

$$\nu_n = 1 + \frac{1}{\delta}(\lambda_n - 1), \quad n = 1, 2, \ldots, N. \tag{5.42}$$

We may therefore focus attention on the eigenvalues of $\boldsymbol{B}$. Note that the eigenvalue problem $\boldsymbol{B}\boldsymbol{v} = \lambda\boldsymbol{v}$ for $\boldsymbol{B}$ is equivalent to the generalized eigenvalue problem

$$\boldsymbol{A}\boldsymbol{v} = \kappa\boldsymbol{M}\boldsymbol{v}, \quad \kappa = \frac{1-\lambda}{\delta}. \tag{5.43}$$

Since $\boldsymbol{M}$ is positive definite, the Cholesky decomposition $\boldsymbol{M} = \boldsymbol{L}\boldsymbol{L}^T$ will transform (5.43) into an ordinary eigenvalue problem for the symmetric matrix $\boldsymbol{L}^{-1}\boldsymbol{A}\boldsymbol{L}^{-T}$. It follows that (5.43), and therefore $\boldsymbol{B}$, has real eigenvalues and a complete set of M-orthogonal eigenvectors $\boldsymbol{v}_n$,

$$\boldsymbol{B}\boldsymbol{v}_n = \lambda_n\boldsymbol{v}_n, \;\; \boldsymbol{v}_n^T\boldsymbol{M}\boldsymbol{v}_m = \delta_{nm}, \quad n, m = 1, 2, \ldots, N. \tag{5.44}$$

From (5.40), one obtains by induction that

$$\boldsymbol{z}_k = p_k(\boldsymbol{B})\boldsymbol{z}_0, \quad k = 0, 1, 2, \ldots, \tag{5.45}$$

where p_k are polynomials of degree k satisfying

$$\begin{gathered} p_{k+1}(\lambda) = \omega_{k+1}\lambda p_k(\lambda) + (1-\omega_{k+1})p_{k-1}(\lambda), \\ p_{-1}(\lambda) = 0, \quad p_0(\lambda) = 1. \end{gathered} \tag{5.46}$$

ETNA
Kent State University
etna@mcs.kent.edu

With $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N]$ denoting the set of eigenvectors of $\boldsymbol{B}$, one has $\boldsymbol{B}\boldsymbol{V} = \boldsymbol{V}\boldsymbol{\Lambda}$, where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_N)$, hence $\boldsymbol{V}^T\boldsymbol{B}\boldsymbol{V} = \boldsymbol{\Lambda}$, $\boldsymbol{V}^T p_k(\boldsymbol{B})\boldsymbol{V} = p_k(\boldsymbol{\Lambda})$, and thus

$$p_k(\boldsymbol{B}) = \boldsymbol{V} p_k(\boldsymbol{\Lambda})\boldsymbol{V}^T. \tag{5.47}$$

The speed of convergence $\boldsymbol{z}_k \to \boldsymbol{0}$ in (5.45), therefore, is determined by the absolutely largest of the quantities $p_k(\lambda_n)$, $n = 1, 2, \ldots, N$.

If we expand $\boldsymbol{z}_0$ in the eigenvectors of B,

$$\boldsymbol{z}_0 = \sum_{i=1}^{N} \alpha_i \boldsymbol{v}_i, \tag{5.48}$$

then from (5.45) we get

$$\boldsymbol{z}_k = \sum_{i=1}^{N} \alpha_i p_k(\lambda_i)\boldsymbol{v}_i.$$

By the M-orthonormality (5.44) of the eigenvectors, the M-inner products of the iterates $\boldsymbol{z}_k$ become

$$\begin{aligned}\langle \boldsymbol{z}_n, \boldsymbol{z}_m\rangle_M := \boldsymbol{z}_n^T\boldsymbol{M}\boldsymbol{z}_m &= \sum_{i=1}^{N} \alpha_i p_n(\lambda_i)\boldsymbol{v}_i^T\boldsymbol{M}\sum_{j=1}^{N}\alpha_j p_m(\lambda_j)\boldsymbol{v}_j \\ &= \sum_{i,j=1}^{N} \alpha_i\alpha_j p_n(\lambda_i)p_m(\lambda_j)\boldsymbol{v}_i^T\boldsymbol{M}\boldsymbol{v}_j \\ &= \sum_{i=1}^{N} \alpha_i^2 p_n(\lambda_i)p_m(\lambda_i),\end{aligned}$$

that is,

$$\langle \boldsymbol{z}_n, \boldsymbol{z}_m\rangle_M = \int_{\mathbb{R}} p_n(\lambda)p_m(\lambda)\mathrm{d}\alpha_N(\lambda). \tag{5.49}$$

Here, $\mathrm{d}\alpha_N$ is a discrete measure supported on the eigenvalues λ_i of $\boldsymbol{B}$ and having jumps α_i^2 at λ_i.

Along with the measure $\mathrm{d}\alpha_N$ there come discrete orthogonal polynomials $\{\pi_k\}$,

$$\int_{\mathbb{R}} \pi_n(\lambda)\pi_m(\lambda)\mathrm{d}\alpha_N(\lambda) \left\{ \begin{array}{ll} = 0 & \text{if } n \neq m, \\ > 0 & \text{if } n = m, \end{array}\right. \qquad n, m = 0, 1, \ldots, N-1 \tag{5.50}$$

and Jacobi matrices $\boldsymbol{J}_k = \boldsymbol{J}_k(\mathrm{d}\alpha_N)$, $k = 1, 2, \ldots, N$. The extreme eigenvalues of $\boldsymbol{J}_k$, i.e., the extreme zeros of π_k, with increasing k, in general provide good approximations to the extreme eigenvalues of $\boldsymbol{B}$, hence by (5.42), to those of $\boldsymbol{M}^{-1}\boldsymbol{N}$.

In order to generate the matrices $\boldsymbol{J}_k$, one can use the modified Chebyshev algorithm (cf. §3.4.2), defining *modified moments* in terms of the polynomials p_k by

$$m_k = \langle \boldsymbol{z}_k, \boldsymbol{z}_0\rangle_M = \int_{\mathbb{R}} p_k(\lambda)\mathrm{d}\alpha_N(\lambda), \quad k = 0, 1, 2, \ldots. \tag{5.51}$$

The polynomials p_k indeed satisfy a three-term recurrence relation with known coefficients (cf. (5.46)). The first relation in (5.51) is used to compute the modified moments.

While the procedure described requires $2m$ modified moments to obtain $\boldsymbol{J}_m$, that is, $2m$ iterations of (5.38), there are special iterative methods, such as the Chebyshev semi-iterative method, where the same can be accomplished already after m iteration (cf. [28, §3]).

A similar method is developed in [4] to determine a few of the largest singular values of a large sparse matrix and the corresponding left and right singular vectors, and is extended in [5] to estimate complex eigenvalues of a large sparse nonsymmetric matrix in connection with an adaptive Chebyshev iterative method.

Acknowledgments. The author is indebted to Professors L. Reichel and Z. Strakoš for helpful comments on a draft version of this paper.

REFERENCES

[1] Z. Bai and G. H. Golub, *Bounds for the trace of the inverse and the determinant of symmetric positive definite matrices*, in The Heritage of P. L. Chebyshev: A Festschrift in Honor of the 70th Birthday of T. J. Rivlin, C. A. Micchelli, ed., Annals Numer. Math., 4 (1997), 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] M. Benzi and G. H. Golub, *Bounds for the entries of matrix functions with applications to preconditioning*, BIT, 39 (1999), pp. 417–438.
[4] M. Berry and G. Golub, *Estimating the largest singular values of large sparse matrices via modified moments*, Numer. Algorithms, 1 (1991), pp. 353–374.
[5] D. Calvetti, G. H. Golub, and L. Reichel, *An adaptive Chebyshev iterative method for nonsymmetric linear systems based on modified moments*, Numer. Math., 67 (1994), pp. 21–40.
[6] D. Calvetti, G. H. Golub, and L. Reichel, *A computable error bound for matrix functionals*, J. Comput. Appl. Math., 103 (1999), pp. 301–306.
[7] D. Calvetti, G. H. Golub, and L. Reichel, *Estimation of the L-curve via Lanczos bidiagonalization*, BIT, 39 (1999), pp. 603–619.
[8] D. Calvetti, G. H. Golub, W. B. Gragg, and L. Reichel, *Computation of Gauss-Kronrod quadrature rules*, Math. Comp., 69 (2000), 1035–1052.
[9] D. Calvetti, S. Morigi, L. Reichel, and F. Sgallari, *An iterative method with error estimators*, in Numerical Analysis 2000, Vol. 5, Quadrature and Orthogonal Polynomials, W. Gautschi, F. Marcellán, and L. Reichel, eds., J. Comput. Appl. Math., 127 (2001), pp. 93–119.
[10] 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.
[11] G. Dahlquist, G. H. Golub, and S. G. Nash, *Bounds for the error in linear systems*, in Semi-Infinite Programming (Proc. Workshop, Bad Honnef, 1978), R. Hettich, ed., Lecture Notes in Control and Information Sci., Vol. 15, Springer-Verlag, Berlin, 1979, pp. 154–172.
[12] S. Elhay, G. H. Golub, and J. Kautsky, *Updating and downdating of orthogonal polynomials with data fitting applications*, SIAM J. Matrix Anal. Appl., 12 (1991), pp. 327–353.
[13] S. Elhay, G. H. Golub, and J. Kautsky, *Jacobi matrices for sums of weight functions*, BIT, 32 (1992), pp. 143–166.
[14] B. Fischer and G. H. Golub, *On generating polynomials which are orthogonal over several intervals*, Math. Comp., 56 (1991), pp. 711–730.
[15] B. Fischer and G. H. Golub, *How to generate unknown orthogonal polynomials out of known orthogonal polynomials*, J. Comput. Appl. Math., 43 (1992), pp. 99–115.
[16] D. Galant, *An implementation of Christoffel's theorem in the theory of orthogonal polynomials*, Math. Comp., 25 (1971), pp. 111–113.
[17] W. Gander and W. Gautschi, *Adaptive quadrature — revisited*, BIT, 40 (2000), pp. 84–101.
[18] F. R. Gantmacher, *The theory of matrices*, Vol. 1, AMS Chelsea Publishing, Providence, RI, 1998. [Translated from the Russian by K. A. Hirsch. Reprint of the 1959 translation.]
[19] W. Gautschi, *A survey of Gauss-Christoffel quadrature formulae*, in E. B. Christoffel (Aachen/Monschau, 1979), P. L. Butzer and F. Fehér, eds., Birkhäuser, Basel, 1981, pp. 72–147.
[20] W. Gautschi, *An algorithmic implementation of the generalized Christoffel theorem*, in Numerical Integration, G. Hämmerlin, ed., Internat. Ser. Numer. Math., Vol. 57, 1982, Birkhäuser, Basel, pp. 89–106.
[21] W. Gautschi, *On generating orthogonal polynomials*, SIAM J. Sci. Statist. Comput., 3 (1982), pp. 289–317.
[22] W. Gautschi and G. V. Milovanović, *s-orthogonality and construction of Gauss-Turán-type quadrature formulae*, J. Comput. Appl. Math., 86 (1997), pp. 205–218.
[23] G. H. Golub, *Some modified matrix eigenvalue problems*, SIAM Rev., 15 (1973), pp. 318–334.
[24] G. H. Golub, *Bounds for matrix moments*, Rocky Mountain J. Math., 4 (1974), pp. 207–211.

[25] G. H. GOLUB, *Matrix computation and the theory of moments*, in Proceedings of the International Congress of Mathematicians (Zürich, 1994), S. D. Chatterji, ed., Vol. 2, Birkhäuser, Basel, 1995, pp. 1440–1448. [See also Bull. Belg. Math. Soc. Simon Stevin 1996, suppl.,1–9.]
[26] G. H. GOLUB AND M. H. GUTKNECHT, *Modified moments for indefinite weight functions*, Numer. Math., 57 (1990), pp. 607–624.
[27] G. H. GOLUB AND J. KAUTSKY, *Calculation of Gauss quadratures with multiple free and fixed knots*, Numer. Math., 41 (1983), pp. 147–163.
[28] G. H. GOLUB AND M. D. KENT, *Estimates of eigenvalues for iterative methods*, Math. Comp., 53 (1989), pp. 619–626.
[29] G. H. GOLUB AND U. VON MATT, *Quadratically constrained least squares and quadratic problems*, Numer. Math., 59 (1991), pp. 561–580.
[30] G. H. GOLUB AND G. MEURANT, *Matrices, moments and quadrature*, in Numerical Analysis 1993 (Dundee, 1993), Pitman Res. Notes Math. Ser., Vol. 303, Longman Sci. Tech., Harlow, 1994.
[31] G. H. GOLUB AND G. MEURANT, *Matrices, moments and quadrature. II. How to compute the norm of the error in iterative methods*, BIT, 37 (1994), pp. 687–705.
[32] G. H. GOLUB AND Z. STRAKOŠ, *Estimates in quadratic formulas*, Numer. Algorithms, 8 (1994), pp. 241–268.
[33] G. H. GOLUB AND J. H. WELSCH, *Calculation of Gauss quadrature rules*, Math. Comp., 23 (1969), pp. 221–230. [Addendum: loose microfiche suppl. A1–A10.]
[34] R. G. GORDON, *Error bounds in equilibrium statistical mechanics*, J. Mathematical Phys., 9 (1968), pp. 655–663.
[35] W. B. GRAGG AND W. J. HARROD, *The numerically stable reconstruction of Jacobi matrices from spectral data*, Numer. Math., 44 (1994), pp. 317–335.
[36] M. R. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards, 49 (1952), pp. 409–436.
[37] J. KAUTSKY AND S. ELHAY, *Calculation of the weights of interpolatory quadratures*, Numer. Math., 40 (1982), pp. 407–422.
[38] J. KAUTSKY AND G. H. GOLUB, *On the calculation of Jacobi matrices*, Linear Algebra Appl., 52/53 (1983), pp. 439–455.
[39] A. S. KRONROD, *Nodes and Weights of Quadrature Formulas. Sixteen-Place Tables*, Consultants Bureau, New York, 1965. [Authorized translation from the Russian.]
[40] D. P. LAURIE, *Calculation of Gauss-Kronrod quadrature rules*, Math. Comp., 66 (1997), pp. 1133–1145.
[41] A. F. NIKIFOROV, S. K. SUSLOV, AND V. B. UVAROV, *Classical Orthogonal Polynomials of a Discrete Variable*, Springer Series in Computational Physics, Springer-Verlag, Berlin, 1991. [Translated from the Russian.]
[42] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Classics in Applied Mathematics, Vol. 20, SIAM, Philadelphia, PA, 1998. [Corrected reprint of the 1980 original.]
[43] R. PIESSENS, E. DE DONCKER-KAPENGA, C. W. ÜBERBUBER, AND D. K. KAHANER, *QUADPACK. A Subroutine Package for Automatic Integration*, Springer Series in Computational Mathematics, Vol. 1, Springer-Verlag, Berlin, 1983.
[44] E. L. STIEFEL, *Kernel polynomials in linear algebra and their numerical applications*, in Further Contributions to the Solution of Simultaneous Linear Equations and the Determination of Eigenvalues, Nat. Bur. Standards Appl. Math. Ser., Vol. 49, U.S. Government Printing Office, Washington, DC, 1958, pp. 1–22.
[45] H. S. WILF, *Mathematics for the Physical Sciences*, John Wiley, New York, 1962. [Reprinted in 1978 by Dover Publications, New York.]
[46] H. S. WILF, Personal communication, 1980.