Matrix Computation and the Theory of Moments*

Gene H. Golub

Abstract

We study methods to obtain bounds or approximations to $u^T f(A)v$ where A is a symmetric, positive definite matrix and f is a smooth function. These methods are based on the use of quadrature rules and the Lanczos algorithm. We give some theoretical results on the behavior of these methods based on results for orthogonal polynomials as well as analytical bounds and numerical experiments on a set of matrices for several functions f. We discuss the effect of rounding error in the quadrature calculation.

1 Introduction

The classical theory of moments plays a vital role in numerical linear algebra. It has long been recognized that there is a strong connection between the theory of moments, Gauss quadrature, orthogonal polynomials and the conjugate gradient method and Lanczos process. In this paper, we will be exploring these connections in order to obtain bounds for various matrix functions which arise in applications.

Let A be a real symmetric positive definite matrix of order n. We want to find upper and lower bounds (or approximations, if bounds are not available) for the entries of a function of a matrix. We shall examine analytical expressions as well as numerical iterative methods which produce good approximations in a few steps. This problem leads us to consider

$$u^T f(A)v, \tag{1.1}$$

where u and v are given vectors and f is some smooth (possibly C^∞) function on a given interval of the real line. As an example, if $f(x) = \frac{1}{x}$ and $u^T = e_i^T = (0, \ldots, 0, 1, 0, \ldots, 0)$, the non zero element being in the i-th position and $v = e_j$, we will obtain bounds on the elements of the inverse A^{-1}.

Some of the technique presented in this paper have been used (without any mathematical justification) to solve problems in solid state physics, particularly to compute elements of the resolvant of a Hamiltonian modeling the interaction of

*Reprinted from Longman Scient. Publ.

atoms in a solid, see [6], [8], [9]. In these studies the function f is the inverse of its argument.

The outline of the paper is as follows. Section 2 considers the problem of characterizing the elements of a function of a matrix. The theory is developed in Section 3 and Section 4 deals with the construction of the orthogonal polynomials that are needed to obtain a numerical method for computing bounds. The Lanczos method used for the computation of the polynomials is presented there. Applications are described in Section 5 where very simple iterative algorithms are give to compute bounds. In Section 6, we discuss some extensions and recent work.

2 Elements of a function of a matrix

Since $A = A^T$,we write A as

$$A = Q\Lambda Q^T,$$

where Q is the orthonormal matrix whose columns are the normalized eigenvectors of A and Λ is a diagonal matrix whose diagonal elements are the eigenvalues λ_i which we order as

$$\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n.$$

By definition, we have

$$f(A) = Qf(\Lambda)Q^T.$$

Therefore,

$$\begin{aligned} u^T f(A)v &= u^T Q f(\Lambda) Q^T v \\ &= \alpha^T f(\Lambda)\beta, \\ &= \sum_{i=1}^{n} f(\lambda_i)\,\alpha_i\beta_i. \end{aligned}$$

This last sum can be considered as a Riemann-Stieltjes integral

$$I[f] = u^T f(A)v = \int_a^b f(\lambda)d\alpha(\lambda), \tag{2.1}$$

where the measure α is piecewise constant and defined by

$$\alpha(\lambda) = \begin{cases} 0 & \text{if} \quad \lambda < a = \lambda_1 \\ \sum_{j=1}^{i} \alpha_j\beta_j & \text{if} \quad \lambda_i \leq \lambda < \lambda_{i+1} \\ \sum_{j=1}^{n} \alpha_j\beta_j & \text{if} \quad b = \lambda_n \leq \lambda. \end{cases}$$

In this paper, we are looking for methods to obtain upper and lower bounds L and U for $I[f]$,

$$L \leq I[f] \leq U.$$

In the next section, we review and describe some basic results from Gauss quadrature theory as this plays a fundamental role in estimating the integrals and computing bounds.

3 Bounds on matrix functions as integrals

One way to obtain the bounds on the integral $I[f]$ is to match the moments associated with the distribution $\alpha(\lambda)$. Thus, we seek to compute quadrature rules so that

$$I\left[\lambda^r\right] = \int_a^b \lambda^r d\alpha(\lambda) = \sum_{j=1}^{N} w_j t_j^r + \sum_{k=1}^{M} v_k z_k^r$$

for $r = 0, 1, \ldots, 2N + M - 1$.
The quantity $I\left[\lambda^r\right]$ is the r^{th} moment associated with the distribution $\alpha(\lambda)$. Note this can be easily calculated since

$$\mu_r \equiv I\left[\lambda^r\right] = u^T A^r v \quad (r = 0, 1, \ldots, 2N + M - 1).$$

The general form of the Gauss, Gauss-Radau and Gauss-Lobatto quadrature formulas, are given by

$$\int_a^b f(\lambda) d\alpha(\lambda) = \sum_{j=1}^{N} w_j f\left(t_j\right) + \sum_{k=1}^{M} v_k f\left(z_k\right) + R[f], \tag{3.1}$$

where the weights $[w_j]_{j=1}^{N}, [v_k]_{k=1}^{N}$ and the nodes $[t_j]_{j=1}^{N}$ are unknowns and the nodes $[z_k]_{k=1}^{M}$ are prescribed, see [1], [2], [3], [4].

When $u = v$, the measure is a positive increasing function and it is known (see for instance [10]) that

$$R[f] = \frac{f^{(2N+M)}(\eta)}{(2N+M)!} \int_a^b \prod_{k=1}^{M} (\lambda - z_k) \left[\prod_{j=1}^{N} (\lambda - t_j)\right]^2 d\alpha(\lambda), \tag{3.2}$$

$$a < \eta < b.$$

If $M = 0$, this leads to the Gauss rule with no prescribed nodes. If $M = 1$ and $z_1 = a$, or $z_1 = b$ we have the Gauss-Radau formula. If $M = 2$ and $z_1 = a, z_2 = b$, this is the Gauss-Lobatto formula.

Let us recall briefly how the nodes and weights are obtained in the Gauss, Gauss-Radau and Gauss-Lobatto rules. For the measure α, it is possible to define a sequence of polynomials $p_0(\lambda), p_1(\lambda), \ldots$ that are orthonormal with respect to α :

$$\int_a^b p_i(\lambda) p_j(\lambda) d\alpha(\lambda) = \begin{cases} 1 & \text{if} \quad i = j \\ 0 & \text{otherwise} \end{cases}$$

and p_k is of exact degree k. Moreover, the roots of p_k are distinct, real and lie in the interval $[a, b]$. We will see how to compute these polynomials in the next section. This set of orthonormal polynomials satisfies a three term recurrence relationship (see [12]):

$$\gamma_j p_j(\lambda) = (\lambda - \omega_j) p_{j-1}(\lambda) - \gamma_{j-1} p_{j-2}(\lambda), \quad j = 1, 2, \ldots, N \tag{3.3}$$

$$p_1(\lambda) \equiv 0, \quad p_0(\lambda) \equiv 1, \text{if} \quad \int d\alpha = 1.$$

In matrix form, this can be written as

$$\lambda p(\lambda) = J_N p(\lambda) + \gamma_N p_N(\lambda) e_N,$$

where

$$p(\lambda)^T = [p_0(\lambda) p_1(\lambda) \dots p_{N-1}(\lambda)], \quad e_N^T = (0, 0, \dots 0, 1),$$

$$J_N = \begin{pmatrix} \omega_1 & \gamma_1 & & & \\ \gamma_1 & \omega_2 & \gamma_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \gamma_{N-2} & \omega_{N-1} & \gamma_{N-1} \\ & & & \gamma_{N-1} & \omega_N \end{pmatrix}. \tag{3.4}$$

The eigenvalues of J_N (which are the zeroes of p_N) are the nodes of the Gauss quadrature rule (i.e. $M = 0$). The weights are the squares of the first elements of the normalized eigenvectors of J_N, cf. [4]. We note that all the eigenvalues of J_N are real and simple.

For the Gauss quadrature rule (renaming the weights and nodes w_j^G and t_j^G), we have

$$\int_a^b f(\lambda) d\alpha(\lambda) = \sum_{j=1}^{N} w_j^G f\left(t_j^G\right) + R_G[f],$$

with

$$R_G[f] = \frac{f^(2N)(\eta)}{(2N)!} \int_a^b \left[\prod_{j=1}^{N} \left(\lambda - t_j^G\right)\right]^2 d\alpha(\lambda),$$

and the next theorem follows.

Theorem 1 *Suppose $u = v$ in (2.1) and f is such that $f^{(2n)}(\xi) > 0$, $\forall n$, $\forall \xi$, $a < \xi < b$, and let*

$$L_G[f] = \sum_{j=1}^{N} w_j^G f\left(t_j^G\right).$$

Then, $\forall N, \exists \eta \in [a, b]$ such that

$$L_G[f] \leq I[f],$$

$$I[f] - L_G[f] = \frac{f^{(2N)}(\eta)}{(2N)!}.$$

A proof of this is given in [10]. To obtain the Gauss-Radau rule ($M = 1$ in 3.1 - 3.2), we extend the matrix J_N in 3.4 in such a way that it has one prescribed eigenvalue, see [5].

For Gauss-Radau, the remainder R_{GR} is

$$R_{GR}[f] = \frac{f^{(2N+1)}(\eta)}{(2N+1)!} \int_a^b (\lambda - z_1) \left[\prod_{j=1}^{N} (\lambda - t_j)\right]^2 d\alpha(\lambda).$$

Therefore, if we know the sign of the derivatives of f , we can bound the remainder. This is stated in the following theorem.

Theorem 2 *Suppose $u = v$ and f is such that $f^{(2n+1)}(\xi) < 0, \forall n, \forall \xi, a < \xi < b$. Let U_{GR} be defined as*

$$U_{GR}[f] = \sum_{j=1}^{N} w_j^a f\left(t_j^a\right) + v_1^a f(a),$$

w_j^a, v_1^a, t_j^a being the weights and nodes computed with $z_1 = a$ and let L_{GR} be defined as

$$L_{GR}[f] = \sum_{j=1}^{N} w_j^b f\left(t_j^b\right) + v_1^b f(b),$$

w_j^b, v_1^b, t_j^b being the weights and nodes computed with $z_1 = b$. Then, $\forall N$ we have

$$L_{GR}[f] \leq I[f] \leq U_{GR}[f],$$

and

$$I[f] - U_{GR}[f] = \frac{f^{(2N+1)}(\eta)}{(2N+1)!} \int_a^b (\lambda - a) \left[\prod_{j=1}^{N} \left(\lambda - t_j^a\right)\right]^2 d\alpha(\lambda),$$

$$I[f] - L_{GR}[f] = \frac{f^{(2N+1)}(\eta)}{(2N+1)!} \int_a^b (\lambda - b) \left[\prod_{j=1}^{N} \left(\lambda - t_j^b\right)\right]^2 d\alpha(\lambda).$$

We remark that we need not always compute the eigenvalues and eigenvectors of the tridiagonal matrix. Let Y_N be the matrix of the eigenvectors of J_N (or $\hat{J}_N$) whose columns we denote by y_i and T_N be the diagonal matrix of the eigenvalues t_i which give the nodes of the Gauss quadrature rule. It is well known that the weights w_i are given by (cf. [13])

$$\frac{1}{w_i} = \sum_{l=0}^{N-1} p_l^2(t_i).$$

It can be easily shown that

$$w_i = \left(\frac{y_i^1}{p_0(t_i)}\right)^2,$$

where y_i^1 is the first component of y_i. But, since $p_0(\lambda) \equiv 1$, we have,

$$w_i = \left(y_i^1\right)^2 = \left(e_1^T y_i\right)^2.$$

Theorem 3

$$\sum_{l=1}^{N} w_l f(t_l) = e_1^T f(J_N) e_1.$$

Proof:

$$\begin{aligned}
\sum_{l=1}^{N} w_l f(t_l) &= \sum_{l=1}^{N} e_1^T y_l f(t_l) y_l^T e_1 \\
&= e_1^T \left(\sum_{l=1}^{N} y_l f(t_l) y_l^T\right) e_1 \\
&= e_1^T Y_N f(T_N) Y_N^T e_1 \\
&= e_1^T f(J_N) e_1.
\end{aligned}$$

The same statement is true for the Gauss-Radau and Gauss-Lobatto rules. Therefore, in some cases where $f(J_N)$ (or the equivalent) is easily computable (for instance, if $f(\lambda) = 1/\lambda$, see Section 5), we do not need to compute the eigenvalues and eigenvectors of J_N.

4 Construction of the orthogonal polynomials

In this section we consider the problem of computing the orthonormal polynomials or equivalently the tridiagonal matrices that we need. A very natural and elegant way to do this is to use Lanczos algorithms. When $u = v$, we use the classical Lanczos algorithm.

Let $x_{-1} = 0$ and x_0 be given such that $||x_0|| = 1$. The Lanczos algorithm is defined by the following relations,

$$\gamma_j x_j = r_j = (A - \omega_j I)\, x_{j-1} - \gamma_{j-1} x_{j-2}, \quad j = 1, \ldots$$

$$\omega_j = x_{j-1}^T A x_{j-1},$$

$$\gamma_j = ||r_j||.$$

The sequence $\{x_j\}_{j=0}^{l}$ is an orthonormal basis of the Krylov space

$$\text{span}\{x_0, Ax_0, \ldots, A^l x_0\}.$$

Proposition 1 *The vector x_j is given by*

$$x_j = p_j(A)x_0,$$

where p_j is a polynomial of degree j defined by the three term recurrence (identical to 3.3)

$$\gamma_j p_j(\lambda) = (\lambda - \omega_j)\, p_{j-1} j(\lambda) - \gamma_{j-1} p_{j-2}(\lambda), \quad p_{-1}(\lambda) \equiv 0, \quad p_0(\lambda) \equiv 1.$$

Theorem 4 *If $x_0 = u$, we have*

$$x_k^T x_l = \int_a^b p_k(\lambda) p_l(\lambda) d\alpha(\lambda).$$

Proof: As the x_j's are orthonormal, we have

$$\begin{aligned} x_k^T x_l &= x_0^T P_k(A)^T P_l(A) x_0 \\ &= x_0^T Q P_k(\Lambda) Q^T Q P_l(\Lambda) Q^T x_0 \\ &= x_0^T Q P_k(\Lambda) P_l(\Lambda) Q^T x_0 \\ &= \sum_{j=1}^{n} p_k(\lambda_j) p_l(\lambda_j) \hat{x}_j^2, \end{aligned}$$

where $\hat{x} = Q^T x_0$. Therefore, the p_j's are the orthonormal polynomials related to α that we have referred to in 3.3.

5 Application

The applications are explained at length in [14], [15], [16].

5.1 Error bounds for linear systems

Suppose we solve a system of equations Ax=b and obtain an approximation ξ to the solution. We desire to estimate the vector e where $x = \xi + e$. Note that $r = b - A\xi = A(x - \xi) = Ae$. Hence, $||e||^2 = r^T A^{-2} r$. Thus, $u = r$, and $f(\lambda) = \lambda^{-2}$.

5.2 Minimizing a quadratic form with a quadratic constraint

Consider the problem of determining x such $x^T Ax - 2b^T x = min$ and $||x||^2 = \alpha^2$. Consider the Lagrangian: $\varphi(x;\mu) = x^T Ax - 2b^T x + \mu\left(x^T x - \alpha^2\right)$. Then grad $\varphi(x;\mu) = 0$ when $(A + \mu I)x = b$. This implies $b^T (A + \mu I)^{-2} b = \alpha^2$. We can approximate the quadratic form $b^T (A + \mu I)^{-2} b$ by using the Lanczos algorithm with the initial vector $b/||b||_2$. This procedure has been extensively studied in [16].

5.3 Inverse elements of a matrix

The elements of the inverse of a matrix are given by $e_j^T A^{-1} e_j$ where e_j is the $j - th$ unit vector. Hence, $f(\lambda) = \lambda^{-1}$. Thus, using the Lanczos process with the initial vector e_j will produce upper and lower bounds on a^{jj} providing a lower bound is known for the smallest eigenvalue and an upper bound for the largest eigenvalue of A. It is desirable to compute the diagonal of the inverse for the Vicsek Fractal Hamiltonian matrix. The matrices are defined as follows.

$$H_1 = \begin{bmatrix} -4 & 1 & 1 & 1 & 1 \\ 1 & -2 & 0 & 0 & 0 \\ 1 & 0 & -2 & 0 & 0 \\ 1 & 0 & 0 & -2 & 0 \\ 1 & 0 & 0 & 0 & -2 \end{bmatrix},$$

$$H_n = \begin{bmatrix} H_{n-1} & V_1^T & V_2^T & V_3^T & V_4^T \\ V_1 & H_{n-1} & 0 & 0 & 0 \\ V_2 & 0 & H_{n-1} & 0 & 0 \\ V_3 & 0 & 0 & H_{n-1} & 0 \\ V_4 & 0 & 0 & 0 & H_{n-1} \end{bmatrix},$$

$$\text{where} \quad H_n \in IR^{N_n \times N_n}$$

$$\text{and} \quad N_{n+1} = 5N_n.$$

The tables show the "exact" values of a^{ii} for some chosen i, and estimated bounds of a^{ii} by using Gauss quadrature rule and Gauss-Radau rule. The "exact" values are computed using the Cholesky decomposition and then triangular inversion. It is a dense matrix method, with storage, $O(N^2)$ and flops $O(N^3)$. The Gauss and the Gauss-Radau rule are sparse matrix methods, and both storage and flop only

Table 1: $N = 125$

i	"exact"	Gauss		Gauss-Radau		
		iter	lower bound	iter	lower bound	upper bound
1	$9.480088e-01$	15	$9.480088e-01$	12	$9.479939e-01$	$9.480112e-01$
10	$6.669905e-01$	13	$6.669846e-01$	13	$6.669864e-01$	$6.669969e-01$
20	$1.156877e+00$	14	$1.156848e+00$	14	$1.156868e+00$	$1.156879e+00$

Table 2: $N = 625$

i	"exact"	Gauss		Gauss-Radau		
		iter	lower bound	iter	lower bound	upper bound
1	$9.480142e-0$	15	$9.480123e-01$	13	$9.480026e-01$	$9.480197e-01$
100	$1.100525e+0$	14	$1.100512e+00$	15	$1.100520e+00$	$1.100527e+00$
301	$9.243102e-0$	14	$9.243074e-01$	12	$9.242992e-01$	$9.243184e-01$
625	$6.440025e-0$	12	$6.439994e-01$	13	$6.440017e-01$	$6.440054e-01$

$O(N)$ because of the structure of the matrix H_n. From these two tables, we see that the error between the "exact" and estimated value is at $O(10^{-5})$, which is generally satisfactorily and also is the stopping criterion used in the inner loop of the Gauss rule and the Gauss-Radau rule.

6 Extensions

These methods, though simple, can be used in many situations involving large scale computations. We have extended these results to bilinear forms and to the situation where one wishes to estimate $W^T f(A)W$ where W is an $N \times p$ matrix ([14]).

It is well known that the *numerical* Lanczos process will produce sequences different than that defined by the mathematical sequence. Nevertheless, it has been shown in [15] that robust estimates of the quadratic form are obtained even in the presence of roundoff.

7 Acknowledgements

I want to thank G.Meurant and Z. Strakoš with whom I have had an active and stimulating collaboration. I, also, want to thank Z. Bai who provided the example given in section 5.

References

[1] P. Davis, P. Rabinowitz, "Methods of numerical integration", Second Edition (1984) Academic Press.

[2] W. Gautschi, "Construction of Gauss–Christoffel quadrature formulas", Math. Comp. 22 (1968) pp 251–270.

[3] W. Gautschi, "Orthogonal polynomials– constructive theory and applications", J. of Comp. and Appl. Math. 12 & 13 (1985) pp 61–76.

[4] G.H. Golub, J.H. Welsch, "Calculation of Gauss quadrature rule" Math. Comp. 23 (1969) pp 221–230.

[5] G.H. Golub, "Some modified matrix eigenvalue problems", SIAM Review v15 n2 (1973) pp 318–334.

[6] R. Haydock, "Accuracy of the recursion method and basis non–orthogonality", Computer Physics Communications 53 (1989) pp 133–139.

[7] G. Meurant, "A review of the inverse of tridiagonal and block tridiagonal matrices", SIAM J. Matrix Anal. Appl. v13 n3 (1992) pp 707–728.

[8] C.M. Nex, "Estimation of integrals with respect to a density of states", J. Phys. A, v11 n4 (1978) pp 653–663.

[9] C.M. Nex, "The block Lanczos algorithm and the calculation of matrix resolvents", Computer Physics Communications 53 (1989) pp 141–146.

[10] J. Stoer, R. Bulirsch, "Introduction to numerical analysis", Second Edition (1983) Springer Verlag.

[11] G.W. Struble, "Orthogonal polynomials: variable–signed weight functions", Numer. Math v5 (1963) pp 88–94.

[12] G. Szegö, "Orthogonal polynomials", Third Edition (1974) American Mathematical Society.

[13] H.S. Wilf, "Mathematics for the physical sciences", (1962) Wiley.

[14] G.H. Golub and G. Meurant, "Matrices, Moments, and Quadrature in Proceedings of the 15-th Dundee Conference, June-July, 1993, D.F. Sciffeths & G.A. Watson, Ests., Longman Scientific & Technical, 1994.

[15] G.H. Golub and Z. Strakoš "Estimates in quadratic formulas", accepted for publication in Numerical Algorithms.

[16] G.H. Golub and U.von Matt "Quadratically constrained least squares and quadratic problems, Num. Math(59), 561-580, 1991.

Computer Science Department
Stanford University
Stanford, CA. 94305 USA
golub@sccm.stanford.edu