Numerical Analysis Project
Manuscript NA-87-02

February 1987

Estimates of Eigenvalues for Iterative Methods

Gene H. Golub
Mark D. Kent

Numerical Analysis Project
Computer Science Department
Stanford University
Stanford, California 94305

February 23, 1987

ESTIMATES OF EIGENVALUES FOR ITERATIVE METHODS*

GENE H. GOLUB AND MARK D. KENT†

Abstract. We describe procedures for determining estimates of the eigenvalues of operators used in various iterative methods for the solution of linear systems of equations. We also show how to determine upper and lower bounds for the error in the approximate solution of linear equations using essentially the same information as that needed for the eigenvalue calculations. The methods described depend strongly upon the theory of moments and Gauss quadrature.

Key words. iterative methods, modified Chebyshev, moments.

1. Introduction. We wish to solve the system of equations

$$Ax = b \tag{1}$$

where A is an n x n, symmetric, positive definite matrix. It is frequently desirable to rewrite (1) as

$$Mx = Nx + c \tag{2}$$

where M and N are symmetric and M is positive definite. We are interested in those situations where it is a much simpler computational task to solve the system $Mz = d$ than it is to solve (1).

We shall use an *iterative* procedure of the form

$$x_{k+1} = x_{k-1} + \omega_{k+1}(\gamma_k z_k + x_k - x_{k-1}), \tag{3}$$

where $Mz_k = c - Ax_k \equiv c - (M - N)x_k$. Depending on the choice of parameters, (3) describes the conjugate gradient **(CG)** method, **the Richardson second order (RSO)** method, or the Chebyshev semi-iterative **(CSI)** method. The success of the latter two methods depends on having good **estimates for the smallest** and largest eigenvalues of the iteration matrix M^{-1} N. In this paper we show how to obtain the *optimal* parameters for the **CSI** method using modified moments calculated from the successive iterates produced by the method. The same can be done for the **RSO** method (see the appendix), however, the results for the **CSI** method are superior.

The algorithm for estimating the optimal parameters is based on the *modified Chebyshev* algorithm given in [5]. See also [6], [14] and [15].

In §2 we review the connection between moments and eigenvalues and introduce modified moments. §3 shows how modified moments arise naturally in certain iterative methods. In §4 we show how to use modified moments to determine successive elements in a tridiagonal matrix whose eigenvalues approach the eigenvalues of the iteration matrix. Improvements are given in §5 which lead to a fairly stable algorithm. Determining error bounds is the topic of §6 and this is followed by computational results in §7.

The appendix describes variations of the primary material of this paper.

*This work was in part supported by National Science Foundation Grant Number DCR 8412314.
†Partially supported by the Natural Sciences and Engineering Research Council of Canada.

2. Moments and Eigenvalues. The connection between moments and eigenvalues is well known (e.g., [13], [12], [8]). For completeness, we include the following short derivation taken from [8].

The Cayley-Hamilton theorem states that every matrix A satisfies its characteristic *polynomial*

$$\chi(\lambda) \equiv \lambda^n - \xi_{n-1}\lambda^{n-1} - \cdots - \xi_0 = 0 \tag{4}$$

i.e.,

$$A^n = \xi_{n-1}A^{n-1} + \ldots + \xi_0 I. \tag{5}$$

For this discussion assume that A has distinct eigenvalues $\lambda_1 < \lambda_2 < \ldots < \lambda_n$. Let

$$v = \sum_{i=1}^{n} \alpha_i w_i, \tag{6}$$

where $\alpha_i \neq 0$ for all i, $Aw_i = \lambda_i w_i$ and $\|w_i\|_2 = 1$. Post-multiplying (5) by v we obtain

$$A^n v = \xi_{n-1}A^{n-1}v + \ldots + \xi_0 v$$

$$A^n v = [v, Av, \ldots, A^{n-1}v]\xi \tag{7}$$

where $\xi = [\xi_0, \xi_1, \ldots, \xi_{n-1}]^T$. The matrix on the right hand side of (7) is the *Krylov matrix* $K^n(v)$ associated with A and v. Let $u = A^n v$. The coefficients of the characteristic polynomial (4) are given by the solution of the following system of n equations in n unknowns

$$K^n(v)\xi = u. \tag{8}$$

Unfortunately, the system of equations (8) is usually ill-conditioned.

Let $H = (K^n(v))^T K^n(v)$ and $c = (K^n(v))^T u$. An equivalent system to (8) *is* the system of *normal equations* $H\xi = c$ (which is even *more* ill-conditioned than (8)). Note that

$$\begin{aligned}[H]_{i+1,j+1} &= (A^i v)^T A^j v \\ &= v^T A^{i+j} v\end{aligned} \tag{9}$$

and so the element $[H]_{ij}$ depends only on the sum of the indices. H is a *Hankel matrix* whose elements are the *moments* $\mu_{i+j} = [H]_{i+1,j+1}$:

$$H = \begin{bmatrix} \mu_0 & \mu_1 & \cdots & \cdots & \mu_{n-1} \\ \mu_1 & \mu_2 & \cdots & \mu_{n-1} & \mu_n \\ \vdots & & & & \vdots \\ \mu_{n-1} & \cdots\cdots & & \cdots & \mu_{2n-2} \end{bmatrix}.$$

To see why we call μ_{i+j} a moment we use (6) and (9) to obtain

$$\mu_{i+j} = v^T A^{i+j} v = v^T \sum_{p=1}^{n} \alpha_p A^{i+j} w_p = v^T \sum_{p=1}^{n} \alpha_p \lambda_p^{i+j} w_p$$

$$\mu_{i+j} = \sum_{p=1}^{n} \alpha_p^2 \lambda_p^{i+j} \tag{10}$$

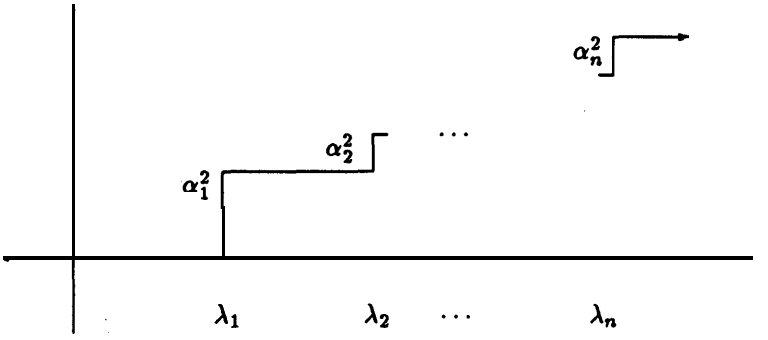

FIG. 2.1: *The measure da(X).*

Equation (10) is equivalent to

$$\mu_k = \int \lambda^k \, da(A), \qquad k = 0, 1, \ldots, 2n-2, \tag{11}$$

where $da(A)$ is a discrete non-negative measure determined as follows:

$$\alpha(\lambda) = \begin{cases} 0, & \text{for } \lambda \leq \lambda_1, \\ \alpha_1^2 + \alpha_2^2 + \cdots + \alpha_t^2, & \text{for } \lambda_t < \lambda \leq \lambda_{t+1}, \quad t = 1, 2, \ldots, \text{n-l}, \\ \alpha_1^2 + \alpha_2^2 + \cdots + \alpha_n^2, & \text{for } \lambda_n < \text{A}. \end{cases} \tag{12}$$

and is shown in Fig. 2.1.

Associated with the measure $d\alpha(\lambda)$ is a set of discrete orthogonal polynomials $\{\varphi_k(\lambda)\}_{k=0}^n$ such that

$$\int \varphi_i(\lambda)\varphi_j(\lambda)\, d\alpha(\lambda) = 0 \quad \text{when } i \neq j$$

and

$$\varphi_n(\lambda) \propto \chi(\lambda).$$

Using the moments $\{\mu_i\}_{i=0}^{2n-2}$ it is possible to compute the coefficients $\{\pi_i\}_{i=1}^n$, $\{\eta_i\}_{i=1}^{n-1}$ which appear in the three term recurrence relation

$$\varphi_{k+1}(\lambda) = (\lambda - \pi_k)\varphi_k(\lambda) - \eta_k\varphi_{k-1}(\lambda) \tag{13}$$

via the Chebyshev algorithm [5]. This, however, is a very ill-conditioned process. The situation can often be improved by replacing the ordinary moments (11) by modified *moments*

$$\nu_k = \int p_k(\lambda)\, d\alpha(\lambda)$$

for a set of suitably chosen polynomials $p_k(\lambda)$. Note that if $p_k(\lambda) = \lambda^k$ then the modified moments reduce to the ordinary moments. The numerical condition of the map from modified moments to the coefficients $\{\pi_i\}_{i=0}^{n-1}$, $\{\eta_i\}_{i=1}^{n-1}$ has been studied by Gautschi [5]. In general, the condition is improved. However, it is clear that the choice of polynomials $p_k(\lambda)$ will have some effect on the stability of the transformation.

3. Modified Moments and Iterative Methods. We now demonstrate how modified moments arise in certain iterative methods for the solution of symmetric, positive definite linear systems $Ax = b$. In the following discussion let $\gamma = \gamma_k$ be fixed.[1]

For future use, we derive some relationships concerning the "generalized residual" vectors z_k:

$$x_{k+1} = x_{k-1} + \omega_{k+1}(\gamma z_k + x_k - x_{k-1})$$

$$c - Ax_{k+1} = c - Ax_{k-1} - \omega_{k+1}(\gamma Az_k + Ax_k - c + c - Ax_{k-1})$$

$$Mz_{k+1} = Mz_{k-1} - \omega_{k+1}(\gamma Az_k + Mz_{k-1} - Mz_k) \tag{14}$$

$$z_{k+1} = z_{k-1} - \omega_{k+1}(\gamma M^{-1}Az_k + z_{k-1} - z_k) \tag{15}$$

$$z_{k+1} = \omega_{k+1}(I - \gamma M^{-1}A)z_k + (1 - \omega_{k+1})z_{k-1}$$

$$z_{k+1} \equiv \omega_{k+1}Bz_k + (1 - \omega_{k+1})z_{k-1} \tag{16}$$

where $B = I - \gamma M^{-1}A \equiv (1 - \gamma)I + \gamma M^{-1}N$. Note that $z_1 = Bz_0$, $z_2 = \omega_2 B^2 z_0 + (1 - \omega_2)z_0$ and in general

$$z_k = \omega_k\omega_{k-1}\cdots\omega_2 B^k z_0 + \cdots$$

$$z_k = P_k(B)z_0 \tag{17}$$

where $P_k(B)$ is a matrix polynomial of degree k in B. From (16) we have the related set of scalar polynomials

$$p_{k+1}(\lambda) = \omega_{k+1}\lambda p_k(\lambda) + (1 - \omega_{k+1})p_{k-1}(\lambda) \tag{18}$$
$$p_{-1}(\lambda) = 0 \qquad p_0(\lambda) = 1.$$

We will need to work with the eigenvalues and eigenvectors of B and so we digress for a moment to investigate the well-known (e.g., Wilkinson [16]) properties of B.

First, consider the eigenproblem $M^{-1}Nx = \lambda x$ or, equivalently, the generalized eigenproblem $Nx = \lambda Mx$, sometimes called the *matrix pencil* (N, M). Note that M and N are symmetric and M is positive definite. We can write M as $M = U\Theta^2U^T$ where U is an orthonormal matrix and $\Theta = \text{diag}(\theta_1, \theta_2, \ldots, \theta_n)$. We use this decomposition of M to write

$$(N - \lambda M) = U\Theta(\Theta^{-1}U^TNU\Theta^{-1} - \lambda I)\Theta U^T$$

$$\det(N - \lambda M) = (\det U)^2(\det \Theta)^2 \det(S - \lambda I)$$

where $S = \Theta^{-1}U^TNU\Theta^{-1}$. Now, $(\det U)^2 = 1$ and $(\det \Theta)^2 = \prod_i \theta_i^2$ hence the zeros of $\det(N - \lambda M)$ are those of $\det(S - \lambda I)$. S is a real symmetric matrix and so it has

[1] This is not redly necessary. However, in many instances the value of γ can be deduced from the specific problem and the computational procedures discussed here may vary γ from a known optimal value.

a complete set of orthonormal eigenvectors $\mathbf{y}_i$. We have

$$\begin{aligned} S\mathbf{y}_i &= \lambda_i \mathbf{y}_i \\ \Theta^{-1}U^T N U \Theta^{-1}\mathbf{y}_i &= \lambda_i \mathbf{y}_i \\ N(U\Theta^{-1}\mathbf{y}_i) &= \lambda_i U\Theta\mathbf{y}_i \\ &= \lambda_i U\Theta(\Theta U^T U\Theta^{-1})\mathbf{y}_i \\ &= \lambda_i M(U\Theta^{-1}\mathbf{y}_i). \end{aligned}$$

Let $\mathbf{q}_i = U\Theta^{-1}\mathbf{y}_i$ then $N\mathbf{q}_i = \lambda_i M\mathbf{q}_i$. The vectors $\mathbf{y}_i$ are orthogonal and $U\Theta^{-1}$ is real and non-singular so the vectors $\mathbf{q}_i$ form a complete set of eigenvectors for the matrix pencil (N, *M*). For i $\neq j$ we have

$$0 = \mathbf{y}_i^T\mathbf{y}_j = (\Theta U^T\mathbf{q}_i)^T\Theta U^T\mathbf{q}_j = \mathbf{q}_i^T U\Theta^2 U^T\mathbf{q}_j = \mathbf{q}_i^T M\mathbf{q}_j$$

and so the vectors $\mathbf{q}_i$ are orthogonal with *respect* to *M*.

Finally

$$\begin{aligned} B\mathbf{q}_i &\equiv [(1-\gamma)I + \gamma M^{-1}N]\,\mathbf{q}_i \\ &\equiv [1 + \gamma(\lambda_i - 1)]\,\mathbf{q}_i \\ &\equiv \tilde{\lambda}_i\mathbf{q}_i. \end{aligned}$$

Express B as $B = Q\tilde{\Lambda}Q^{-1}$ where $Q = [\mathbf{q}_1, \ldots, \mathbf{q}_n]$ and $\tilde{\Lambda} = \text{diag}\,(\tilde{\lambda}_1, \tilde{\lambda}_2, \ldots, \tilde{\lambda}_n)$. **Clearly,** $P_k(B) = QP_k(\tilde{\Lambda})Q^{-1}$. We can write $\mathbf{z}_0$ as a linear combination of the eigenvectors $\mathbf{q}_i$ of B

$$\mathbf{z}_0 = \sum_{i=1}^{n}\alpha_i\mathbf{q}_i.$$

Let $P_k(B) = \sum_{j=0}^{k} c_j B^j$ then using (17) we have

$$\begin{aligned} \mathbf{z}_k &= QP_k(\tilde{\Lambda})\sum_{i=1}^{n}\alpha_i Q^{-1}\mathbf{q}_i \\ &= Q\sum_{j=0}^{k} c_j\tilde{\Lambda}^j\sum_{i=1}^{n}\alpha_i\mathbf{e}_i \\ &= \sum_{j=0}^{k} c_j\sum_{i=1}^{n}\alpha_i\tilde{\lambda}_i^j Q\mathbf{e}_i \\ &= \sum_{i=1}^{n}\alpha_i p_k(\tilde{\lambda}_i)\mathbf{q}_i. \end{aligned}$$

Now, forming the inner product

$$\begin{aligned} \langle \mathbf{z}_k, \mathbf{z}_l\rangle &= (\mathbf{z}_k, M\mathbf{z}_l) \\ &= \left(\sum_{i=1}^{n}\alpha_i p_k(\tilde{\lambda}_i)\mathbf{q}_i\right)^T \cdot \left(M\sum_{j=1}^{n}\alpha_j p_l(\tilde{\lambda}_j)\mathbf{q}_j\right) \\ &= \sum_{i=1}^{n}\alpha_i^2 p_k(\tilde{\lambda}_i)p_l(\tilde{\lambda}_i) \\ &= \int p_k(\tilde{\lambda})p_l(\tilde{\lambda}) \qquad d,(i). \end{aligned}$$

Hence $\langle z_k, z_0 \rangle = \nu_k$. With each iteration of (3) we can calculate a modified moment.

4. Using Modified Moments. The results of the previous section indicate that we can calculate the modified moments[2]

$$\nu_k = \int p_k(\lambda)\, d\alpha(\lambda). \tag{19}$$

We will use these moments to obtain another set of polynomials $\psi_k(\lambda)$ with $\psi_{-1}(\lambda) = 0$, $\psi_0(\lambda) = 1$,

$$\int \psi_k(\lambda)\psi_l(\lambda)\, d\alpha(\lambda) = 0, \quad \text{for } k \neq l \tag{20}$$

and $\psi_n(\lambda_i) = 0$, (i.e., $\psi_n(\lambda) \propto \chi(\lambda)$). Additionally,-we require the polynomials $\psi_k(\lambda)$ to have the form

$$\psi_{k+1}(\lambda) = (\lambda\omega_{k+1} - a_k)\psi_k(\lambda) - b_k\psi_{k-1}(\lambda) \tag{21}$$

rather than the more conventional form (13). This choice will produce a more **computationally** attractive algorithm. The results for $\psi_k(\lambda)$ of the form (13) are contained in the appendix. **Our** aim is to compute the coefficients a_k, b_k in (21). The technique used here is similar to that in [5]. It is repeated here due to the variations imposed by the choice (21).

Define

$$\sigma_{kl} = \int \psi_k(\lambda)p_l(\lambda)\, d\alpha(\lambda). \tag{22}$$

Then, since we can write

$$p_l(\lambda) = \psi_l(\lambda) + \sum_{j=0}^{l-1} \eta_j \psi_j(\lambda)$$

we have $\sigma_{kl} = 0$ for $k > 1$. Now consider

$$\begin{aligned} \int \psi_k^2(\lambda)\, d\alpha(\lambda) &= \int \psi_k(\lambda) \left(\sum_{j=0}^{k} \theta_j p_j(\lambda) \right) d\alpha(\lambda) \\ &= \theta_k \int \psi_k(\lambda)p_k(\lambda)\, d\alpha(\lambda) \qquad (23) \\ &= \sigma_{kk} \quad \textit{since } \theta_k = 1. \end{aligned}$$

Then, replacing $p_k(\lambda)$ in (23) with the recurrence relation (18) we obtain

$$\begin{aligned} \sigma_{kk} &= \int \psi_k(\lambda)\left[\omega_k\lambda p_{k-1}(\lambda) + (1 - \omega_k)p_{k-2}(\lambda)\right] d\alpha(\lambda) \\ &= \omega_k \int \lambda\psi_k(\lambda)p_{k-1}(\lambda)\, d\alpha(\lambda). \qquad (24) \end{aligned}$$

Rearranging (21) gives

$$\lambda\psi_k(\lambda) = \left(\psi_{k+1}(\lambda) + a_k\psi_k(\lambda) + b_k\psi_{k-1}(\lambda)\right)/\omega_{k+1}, \tag{25}$$

[2] **From** now on, we will not **distinguish** between λ and $\tilde{\lambda}$.

and substituting (25) in (24)

$$\sigma_{kk} = \frac{\omega_k}{\omega_{k+1}} \int [\psi_{k+1}(\lambda) + a_k\psi_k(\lambda) + b_k\psi_{k-1}(\lambda)]p_{k-1}(\lambda)\, d\alpha(\lambda),$$

$$= \frac{\omega_k}{\omega_{k+1}} b_k \sigma_{k-1,k-1}. \tag{26}$$

From (26) we get an expression for b_k :

$$b_k = \frac{\omega_{k+1}}{\omega_k} \cdot \frac{\sigma_{kk}}{\sigma_{k-1,k-1}}. \tag{27}$$

To find an expression for a_k we note that

$$\sigma_{k+1,k} = \int \psi_{k+1}(\lambda)p_k(\lambda)\, d\alpha(\lambda) = 0, \tag{28}$$

and replacing $\psi_{k+1}(\lambda)$ in (28) by the recurrence relation (21)

$$0 = \int [(\lambda\omega_{k+1} - a_k)\psi_k(\lambda) - b_k\psi_{k-1}(\lambda)]\, p_k(\lambda)\, d\alpha(\lambda),$$

$$= \int \omega_{k+1}\lambda p_k(\lambda)\psi_k(\lambda)\, d\alpha(\lambda) - a_k\sigma_{kk} - b_k\sigma_{k-1,k}. \tag{29}$$

The recurrence relation for $p_k(\lambda)$ in (18) gives

$$\omega_{k+1}\lambda p_k(\lambda) = p_{k+1}(\lambda) - (1 - \omega_{k+1})p_{k-1}(\lambda), \tag{30}$$

and substituting (30) into (29) results in

$$0 = \int \psi_k(\lambda)\big(p_{k+1}(\lambda) - (1 - \omega_{k+1})p_{k-1}(\lambda)\big)\, d\alpha(\lambda) - a_k\sigma_{kk} - b_k\sigma_{k-1,k},$$

$$0 = \sigma_{k,k+1} - a_k\sigma_{kk} - b_k\sigma_{k-1,k}. \tag{31}$$

Rearranging (31) gives an expression for a_k :

$$a_k = \frac{\sigma_{k,k+1}}{\sigma_{kk}} - b_k\frac{\sigma_{k-1,k}}{\sigma_{kk}},$$

$$= \frac{\sigma_{k,k+1}}{\sigma_{kk}} - \frac{\omega_{k+1}}{\omega_k}\frac{\sigma_{k-1,k}}{\sigma_{k-1,k-1}}. \tag{32}$$

Now we seek a recurrence for σ_{kl} for $l \geq k$.

$$\sigma_{kl} = \int \psi_k(\lambda)p_l(\lambda)\, d\alpha(\lambda),$$

$$= \int [(\lambda\omega_k - a_{k-1})\psi_{k-1}(\lambda) - b_{k-1}\psi_{k-2}(\lambda)]p_l(\lambda)\ da(X),$$

$$= \omega_k \int \lambda\psi_{k-1}(\lambda)p_l(\lambda)\ d\alpha(\lambda) - a_{k-1}\sigma_{k-1,l} - b_{k-1}\sigma_{k-2,l}. \tag{33}$$

Once again we use (18)

$$\lambda p_l(\lambda) = [p_{l+1}(\lambda) - (1 - \omega_{l+1})p_{l-1}(\lambda)]/\omega_{l+1}, \tag{34}$$

and substituting (34) in (33) gives an expression for σ_{kl}:

$$\sigma_{kl} = \frac{\omega_k}{\omega_{l+1}}[\sigma_{k-1,l+1} - (1 - \omega_{l+1})\sigma_{k-1,l-1}] - a_{k-1}\sigma_{k-1,l} - b_{k-1}\sigma_{k-2,l}. \tag{35}$$

Equations (35), (32) and (27) furnish the algorithm for determining the coefficients of (21). The initialization required is

$$\begin{aligned} \sigma_{-1,l} &= \int \psi_{-1}(\lambda)p_l(\lambda)\, d\alpha(\lambda) = 0, \\ \sigma_{0,l} &= \int p_l(\lambda)\, d\alpha(\lambda) = \nu_l, \quad \text{for} \quad l = 0, 1, \ldots, 2m - 1, \\ a_0 &= \frac{\nu_1}{\nu_0}, \quad b_0 = 0, \end{aligned} \tag{36}$$

and the algorithm continues with

$$\begin{aligned} \sigma_{kl} &= \frac{\omega_k}{\omega_{l+1}}[\sigma_{k-1,l+1} - (1 - \omega_{l+1})\sigma_{k-1,l-1}] - a_{k-1}\sigma_{k-1,l} - b_{k-1}\sigma_{k-2,l}, \\ &\qquad \text{for } l = k,\ k + 1, \ldots, 2m - k - 1, \\ a_k &= \frac{\sigma_{k,k+1}}{\sigma_{kk}} - \frac{\omega_{k+1}}{\omega_k} \cdot \frac{\sigma_{k-1,k}}{\sigma_{k-1,k-1}}, \quad b_k = \frac{\omega_{k+1}}{\omega_k} \cdot \frac{\sigma_{kk}}{\sigma_{k-1,k-1}}, \end{aligned} \tag{37}$$

for $k = 1, 2, \ldots, m - 1$.

The choice of polynomials of the form (21) leads to a generalized eigenvalue problem. Writing (21) in matrix form we obtain

$$\begin{bmatrix} a_0 & 1 & & & \\ b_1 & a_1 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{k-1} & a_{k-1} & 1 \\ & & & b_k & a_k \end{bmatrix} \begin{bmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \vdots \\ \psi_k \end{bmatrix} = \lambda \operatorname{diag}(\omega_1, \ldots, \omega_{k+1}) \begin{bmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \vdots \\ \psi_k \end{bmatrix} - \mathbf{e}_{k+1}\psi_{k+1}. \tag{38}$$

To find the zeros of $\psi_{k+1}(\lambda)$ we solve (38) by transforming it to a standard eigenvalue problem $J_{k+1}\mathbf{x} = \lambda\mathbf{x}$ where J_{k+1} is a symmetric tridiagonal matrix

$$J_{k+1} = \begin{bmatrix} \alpha_0 & \beta_1 & & & \\ \beta_1 & \alpha_1 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{k-1} & \alpha_{k-1} & \beta_k \\ & & & \beta_k & \alpha_k \end{bmatrix}, \quad \text{with} \quad \alpha_i = \frac{a_i}{\omega_{i+1}} \quad \text{and} \quad \beta_i = \sqrt{\frac{b_i}{\omega_i\omega_{i+1}}}.$$

The computational procedure described is, in some sense, equivalent to the Lanczos algorithm [13], [9]. Thus, we would expect that the extreme eigenvalues of J_k will provide good approximations to the extreme eigenvalues of B as k increases. Once we have sufficiently accurate estimates for the largest and smallest eigenvalue we can restart the CSI method with (near) optimal parameters.

5. Improving the Calculation of the Modified Moments. We will now derive improved expressions for the modified moments ν_k in terms of the vectors z_k for the CSI method. After $2m$ iterations of (3) we can calculate $\{a_k\}_{k=0}^{m-1}$ and $\{b_k\}_{k=0}^{m-2}$ and so have, in principle, the roots of $\psi_m(\lambda)$.

Obviously, it would be advantageous to have the roots of $\psi_m(\lambda)$ after only m iterations of (3). It is possible to attain this goal, if the iteration (3) is the Chebyshev semi-iterative method. For the CSI method we have estimates, a and b, of the smallest and largest eigenvalues of B and the parameters, in terms of $\mu = (b - a)/(b + a)$, are given by

$$\gamma = \frac{2}{b+a}, \qquad \omega_{k+1} = \frac{1}{1 - \frac{\mu^2}{4}\omega_k} \quad \text{with } \omega_1 = 1, \qquad \omega_2 = \frac{1}{1 - \frac{\mu^2}{2}}.$$

It is well known [10] that

$$p_k(\lambda) = \frac{C_k(\lambda/\mu)}{C_k(1/\mu)} \tag{39}$$

where

$$C_k(x) = \begin{cases} \cos(k \cos^{-1} x), & |x| \le 1,\ k \ge 0; \\ \cosh(k \cosh^{-1} x), & |x| \ge 1,\ k \ge 0. \end{cases}$$

$C_k(x)$ satisfies the three term recurrence relation $C_{k+1}(x) = 2xC_k(x) - C_{k-1}(x)$ with initial conditions $C_0(x) = 1$ and $C_1(x) = x$. From the classical identity $\cos(k + l)\theta = 2 \cos k\theta \cos l\theta - \cos(k - l)\theta$, we have $C_{k+l} = 2C_kC_l - C_{|k-l|}$. Hence for $l = k$,

$$C_{2k} = 2C_k^2 - C_0, \quad C_{2k+1} = 2C_kC_{k+1} - C_1. \tag{40}$$

Working from (39) and making liberal use of these identities we obtain

$$p_{2k}(\lambda) = \frac{C_{2k}(\lambda/\mu)}{C_{2k}(1/\mu)} = p_k^2(\lambda) + \frac{1}{C_{2k}(1/\mu)}[p_k^2(\lambda) - 1]$$

which leads directly to the modified moment

$$\nu_{2k} = \langle z_k, z_k \rangle + \frac{1}{C_{2k}(1/\mu)}(\langle z_k, z_k \rangle - \nu_0). \tag{41}$$

Similarly, for $p_{2k+1}(\lambda)$:

$$p_{2k+1}(\lambda) = p_k(\lambda)p_{k+1}(\lambda) + \frac{1}{\mu C_{2k+1}(1/\mu)}[p_k(\lambda)p_{k+1}(\lambda) - p_1(\lambda)]$$

leads to the modified moment

$$\nu_{2k+1} = \langle z_k, z_{k+1} \rangle + \frac{1}{\mu C_{2k+1}(1/\mu)}(\langle z_k, z_{k+1} \rangle - \nu_1). \tag{42}$$

The first two modified moments are given by $\nu_0 = \langle z_0, z_0 \rangle$ and $\nu_1 = \langle z_0, z_1 \rangle$.

6. Error Bounds. In this section we indicate how to obtain error bounds for an approximate solution to the system $Ax = b$. We now summarize some of the results given in [3] and [7].

Let the eigenvalues of A satisfy $a < \lambda_1 \le \lambda_2 \le \ldots \le \lambda_n \le b$ corresponding to eigenvectors $\{w_i\}_{i=1}^n$. Suppose we have an approximation to x which we denote as

$\tilde{\mathbf{x}}$. Now we seek to find upper and lower bounds on $\|\mathbf{x} - \tilde{\mathbf{x}}\|$. Let the residual vector be $\mathbf{r}_0 = \mathbf{b} - A\tilde{\mathbf{x}}$ so that $\mathbf{r}_0 = A(\mathbf{x} - \tilde{\mathbf{x}})$. Hence

$$\frac{\|\mathbf{r}_0\|}{\|A\|} \leq \|\mathbf{x} - \tilde{\mathbf{x}}\| \leq \|A^{-1}\| \cdot \|\mathbf{r}_0\|.$$

Writing $\mathbf{r}_0 = \sum_{i=1}^{n} \alpha_i \mathbf{w}_i$, we have (cf. 32)

$$\mu_p = (A^p \mathbf{r}_0, \mathbf{r}_0) = \sum_{i=1}^{n} \alpha_i^2 \lambda_i^p, \quad (p = 0, 1, \ldots, 2k). \tag{43}$$

Now $\mathbf{x} - \tilde{\mathbf{x}} = A^{-1}\mathbf{r}_0$ so

$$\|\mathbf{x} - \tilde{\mathbf{x}}\|_2^2 = (A^{-2}\mathbf{r}_0, \mathbf{r}_0) = \sum_{i=1}^{n} \alpha_i^2 \lambda_i^{-2} = \mu_{-2}. \tag{44}$$

Equations (43) and (44) are equivalent to (see (11) and (12))

$$\mu_p = \int_a^b \lambda^p \, d\alpha(\lambda), \quad (p = -2, 0, 1, \ldots, 2k). \tag{45}$$

Then determining upper and lower bounds on $\|\mathbf{x} - \tilde{\mathbf{x}}\|_2$ is equivalent to the following. **Problem:** Given $2k+1$ moments $\{\mu_i\}_{i=0}^{2k}$, determine upper and lower bounds on μ_{-2}. The solution to this problem is related to the classical theory of moments. In order to give a numerical algorithm, we review some facts from the theory of Gaussian quadrature.,

Suppose we are given $\{\mu_i\}_{i=0}^{2k}$ and a function $\psi(\lambda)$ defined in the interval $(a \leq \lambda \leq b)$, and we wish to determine (L, U) so that

$$L \leq \int_a^b \psi(\lambda) \, d\alpha(\lambda) \leq U.$$

We can determine a quadrature rule such that

$$\mu_r = \int_a^b \lambda^r \, d\alpha(\lambda) = \sum_{i=1}^{k} A_i t_i^r + \sum_{j=1}^{m} B_j z_j^r, \tag{46}$$

for $r = 0, 1, \ldots, 2k + m - 1$. Here $\{A_i, t_i\}_{i=1}^{k}$ and $\{B_j\}_{j=1}^{m}$ are unknown and the $\{z_j\}_{j=1}^{k}$ is specified. Such a quadrature rule is called **a generalized Gauss-Radau rule.**[3]

Then

$$\int_a^b \psi(\lambda) \, d\alpha(\lambda) = \sum_{i=1}^{k} A_i \psi(t_i) + \sum_{j=1}^{m} B_j \psi(z_j) + R[\psi]$$

where the remainder term, $R[\psi]$, is given by

$$R[\psi] = \frac{\psi^{(2k+m)}(\eta)}{(2k+m)!} \int_a^b \prod_{j=1}^{m} (\lambda - z_j) \left[\prod_{i=1}^{k} (\lambda - t_i) \right]^2 d\alpha(\lambda), \quad a < \eta < b. \tag{47}$$

[3] To be distinguished from a normal Gauss-Radau rule where we have $m = 1$ and $z_1 = a$ or $z_1 = b$ *in (46).*

Thus if $\psi(\lambda) = \lambda^{-2}$ and $m = 1$,

$$R[\lambda^{-2}] = -2(k+1)\eta^{-(2k+3)} \int_a^b (\lambda - z_1) \left[\prod_{i=1}^{k} (\lambda - t_i)\right]^2 d\alpha(\lambda).$$

Hence for $z_1 = a > 0$, the Gauss-Radau rule yields an ***upper*** bound for μ_{-2} and if $z_1 = b$, we have a ***lower*** bound from the Gauss-Radau rule. It can be shown [1, pg. 80] that these bounds are attainable. Later in our discussion, we shall need upper and lower bounds for μ_{-1}. Note that since $R[\lambda^{-1}] < 0$ when $z_1 = a$, the Gauss-Radau rule will yield an upper bound on μ_{-1} and that when $z_1 = b$, we have a lower bound.

It is well known that finding the Gauss quadrature rule is equivalent to finding the zeroes of the set of orthogonal polynomials, $\{\varphi_j(\lambda)\}_{j=0}^{k}$, associated with the measure $d\alpha(\lambda)$. Thus if $\varphi_k(t_i) = 0$ for $i = 1, 2, \ldots, k$, then

$$\mu_r = \int_a^b \lambda^r d\alpha(\lambda) = \sum_{i=1}^{k} A_i t_i^r, \quad (r = 0, 1, \ldots, 2k-1). \tag{48}$$

As previously mentioned (cf. §2), the coefficients $\{\pi_j\}_{j=1}^{k}$, $\{\eta_j^2\}_{j=1}^{k-1}$ can be computed directly from the moments but this process is generally numerically unstable.

Given $\{\pi_j\}_{j=1}^{k}$, $\{\eta_j^2\}_{j=1}^{k-1}$, we can compute the Gauss quadratures as follows [7]: Let $J_k = \text{tridiag}\{\eta_{j-1}, \pi_j, \eta_j\}$. Then the eigenvalues of J_k are the nodes of the quadrature rule. Furthermore, if $\mu_0 = 1$, then the square of the first component of each normalized eigenvector is the weight associated with each t_i. The eigenvalues and first component can be efficiently computed by the QR method.

The computation of the Gauss-Radau rule ($m = 1$) is only slightly more complicated. The idea is the following. Let

$$\bar{J}_{k+1} = \begin{bmatrix} \pi_1 & \eta_1 & & & \\ \eta_1 & \pi_2 & \eta_2 & & \\ & \eta_2 & \ddots & \ddots & \\ & & \ddots & \pi_k & \eta_k \\ & & & \eta_k & \bar{\pi}_{k+1} \end{bmatrix}.$$

For notational convenience, we designate t_0 as the prescribed node.[4] We wish to calculate $\bar{\pi}_{k+1}$ so that $\varphi_{k+1}(t_0) = 0$, and hence the eigenvalues and eigenvectors of $\bar{J}_{k+1}$ yield the Gauss-Radau rule. Now

$$0 = \varphi_{k+1}(t_0) = (t_0 - \bar{\pi}_{k+1})\varphi_k(t_0) - \eta_k^2 \varphi_{k-1}(t_0)$$

and hence $\bar{\pi}_{k+1} = t_0 - \eta_k^2 \varphi_{k-1}(t_0)/\varphi_k(t_0)$. The Computation of $\bar{\pi}_{k+1}$ is simplified as follows: Solve

$$(J_k - t_0 I)\delta = \eta_k^2 \mathbf{e}_k$$

where $\mathbf{e}_k = (0, 0, \ldots, 0, 1)^T$. Then $\bar{\pi}_{k+1} = t_0 + \delta_k$.

We need **not** compute the eigenvalue-eigenvector decomposition to determine upper and lower bounds on μ_{-2} and μ_{-1}. Let $\bar{J}_{k+1} = QTQ^T$ where $Q^TQ = I_{k+1}$

[4] The due of t_0 is either a or b.

and T is the diagonal matrix of eigenvalues of $\bar{J}_{k+1}$. The vector $Q^T\mathbf{e}_1$ consists of the first component of each eigenvector of $\bar{J}_{k+1}$. Hence (cf. (48)),

$$\mu_s \cong \sum_{i=0}^{k} A_i t_i^s = \mathbf{e}_1^T \bar{J}_{k+1}^s \mathbf{e}_1.$$

Thus if $s = -2$

$$\mu_{-2} \cong \sum_{i=0}^{k} A_i t_i^{-2} = (\mathbf{e}_1^T \bar{J}_{k+1}^{-1})(\bar{J}_{k+1}^{-1}\mathbf{e}_1),$$

and for $s = -1$

$$\mu_{-1} \cong \sum_{i=0}^{k} A_i t_i^{-1} = \mathbf{e}_1^T \bar{J}_{k+1}^{-1}\mathbf{e}_1.$$

6.1. The Conjugate Gradient Method. We now describe how the ideas given above can be used in combination with the conjugate gradient method. Our idea is to be able to construct bounds on the error after k steps of the conjugate gradient method.

The conjugate gradient algorithm can also be written in the form (3) [2]. Recall that (x, y) = (x, My). The parameters are chosen to ensure that the generalized *residual* vectors satisfy $\langle \mathbf{z}_k, \mathbf{z}_{k+1}\rangle = 0$ and $\langle \mathbf{z}_{k-1}, \mathbf{z}_{k+1}\rangle = 0$. In exact arithmetic, all the generalized residuals would be mutually orthogonal and in the following derivation it is assumed that $\langle \mathbf{z}_l, \mathbf{z}_m\rangle \equiv (\mathbf{z}_l, M\mathbf{z}_m) = 0$ for $l \neq m$.

Starting from the three-term recurrence for the residual vectors (14) we form $\langle \mathbf{z}_k, \mathbf{z}_{k+1}\rangle$ and $\langle \mathbf{z}_{k-1}, \mathbf{z}_{k+1}\rangle$:

$$\langle \mathbf{z}_k, \mathbf{z}_{k+1}\rangle = -\omega_{k+1}[\gamma_k(\mathbf{z}_k, (M-N)\mathbf{z}_k) - \langle \mathbf{z}_k, \mathbf{z}_k\rangle], \tag{49}$$

$$\langle \mathbf{z}_{k-1}, \mathbf{z}_{k+1}\rangle = \langle \mathbf{z}_{k-1}, \mathbf{z}_{k-1}\rangle - \omega_{k+1}[\gamma_k(\mathbf{z}_{k-1}, (M-N)\mathbf{z}_k) + \langle \mathbf{z}_{k-1}, \mathbf{z}_{k-1}\rangle]. \tag{50}$$

From (49) we see that if $\langle \mathbf{z}_k, \mathbf{z}_{k+1}\rangle = 0$ then either $\omega_{k+1} = 0$ (in which case iteration (3) has terminated), or

$$\gamma_k = \frac{\langle \mathbf{z}_k, \mathbf{z}_k\rangle}{(\mathbf{z}_k, (M-N)\mathbf{z}_k)}. \tag{51}$$

Now, setting (50) equal to zero we obtain

$$\omega_{k+1} = \left(1 - \gamma_k \frac{(\mathbf{z}_{k-1}, N\mathbf{z}_k)}{\langle \mathbf{z}_{k-1}, \mathbf{z}_{k-1}\rangle}\right)^{-1}, \tag{52}$$

or, using only M-inner products

$$\omega_{k+1} = \left(1 - \frac{\gamma_k}{\omega_k \gamma_{k-1}} \frac{\langle \mathbf{z}_k, \mathbf{z}_k\rangle}{\langle \mathbf{z}_{k-1}, \mathbf{z}_{k-1}\rangle}\right)^{-1}. \tag{53}$$

Not surprisingly, it is possible to get eigenvalue estimates from the conjugate gradient algorithm. From (15) we have

$$\mathbf{z}_{k+1} = \mathbf{z}_{k-1} - \omega_{k+1}\big(\gamma(I - M^{-1}N)\mathbf{z}_k + \mathbf{z}_{k-1} - \mathbf{z}_k\big). \tag{54}$$

Rearranging (54) gives us

$$(I - M^{-1}N)\mathbf{z}_k = c_{k-1}\mathbf{z}_{k-1} + a_k\mathbf{z}_k + b_{k+1}\mathbf{z}_{k+1}, \tag{55}$$

defining a_k, b_k and c_k. We can Write (55) in matrix notation as $(I - M^{-1}N)Z_k = Z_k J_k$ where $Z_k = [z_0, z_1, \ldots, z_k]$ and

$$J_k = \begin{bmatrix} a_0 & c_0 & & & \\ b_1 & a_1 & c_1 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{k-2} & a_{k-2} & c_{k-2} \\ & & & b_{k-1} & a_{k-1} \end{bmatrix}.$$

Note that J_k can be symmetrized. This is essentially the Lanczos process and we can expect that the extreme eigenvalues of J_k will approximate the extreme eigenvalues of $(I - M^{-1}$ N) even for relatively small k.

In the special case where $M = I$, an algorithm has been given in [4] for computing upper and lower bounds for

$$\varepsilon_k^T A \varepsilon_k \tag{56}$$

where ε_k 3 x $- x_k$. The bounds are based on the fact that the conjugate gradient method yields a minimum for (56). We are able to show that for the CG method

$$\varepsilon_k^T A \varepsilon_k = \mu_{-1} - \mu^T M_1^{-1} \mu \tag{57}$$

where $\mu = (\mu_0, \mu_1, \ldots, \mu_k)^T$ and

$$M_1 = \begin{bmatrix} \mu_1 & \mu_2 & \cdots & \cdots & \mu_{k+1} \\ \mu_2 & \mu_3 & \cdots & \mu_{k+1} & \mu_{k+2} \\ | & & & & | \\ \mu_{k+1} & \cdots & \cdots & \cdot & \mu_{2k+1} \end{bmatrix}.$$

Thus obtaining an upper and lower bound in (56) is equivalent to determining upper and lower bounds for μ_{-1} and we have already shown how to make this computation. The remaining term in (57) is computed as follows:

$$\mu^T M_1^{-1} \mu = \mu_0 \{J_{k+1}^{-1}\}_{11} = \|r_0\|^2 \{J_{k+1}^{-1}\}_{11}.$$

Thus, we obtain the following algorithm for bounding $\|\varepsilon_{k+1}\|$:

1. Construct $\overline{J}_{k+1}$.
2. Solve $J_{k+1}\mathbf{h} = \mathbf{e}_1$.
3. Solve $\overline{J}_{k+1}\mathbf{g} = \mathbf{e}_1$.
4. $\|\varepsilon_{k+1}\|_A^2 \leq \mu_0(\mathbf{e}_1^T \overline{J}_{k+1}^{-1} \mathbf{e}_1 - \mathbf{e}_1^T J_{k+1}^{-1} \mathbf{e}_1) = \mu_0(g_1 - h_1)$.

6.2. The Chebyshev Semi-Iterative Method. We return to the CSI method for $Ax = b$. We have

$$(x - x_k) = B(x - x_{k-1}) = B^k(x - x_0),$$

where $B = I - \gamma A$ (*i.e.*, $M = I$ in (14-17)).

Now since $r_k = A\varepsilon_k$, and $A = \$(I - B)$, we see that $A\varepsilon_k = P_k(B)r_0$, and hence

$$\varepsilon_k^T A \varepsilon_k = \gamma r_0^T P_k(B)(I - B)^{-1} P_k(B) r_0 = 7 \int_{\theta_1}^{\theta_n} \frac{p_k^2(\theta)}{1 - \theta} d\alpha(\theta). \tag{58}$$

Now

$$\frac{p_k^2(\theta)}{1-\theta} = \frac{p_k^2(\theta)-1}{1-\theta} + \frac{1}{1-\theta}.$$

Note that $\frac{p_k^2(\theta)-1}{1-\theta}$ is a polynomial of degree $2k-1$ and

$$\frac{d^r}{d\theta^r}(1-\theta)^{-1} = r!(1-\theta)^{-(r+1)} \geq 0, \quad \text{for } |\theta| < 1.$$

Thus, if we apply the Gauss-Radau rule to

$$q_k(\theta) \equiv \frac{p_k^2(\theta)}{1-\theta}$$

with $m = 1$,

$$R[q_k(\theta)] = (1-\eta)^{-(2k+1)} \int_{\theta_1}^{\theta_n} (\theta - z_1) \left[\prod_{j=1}^{k} (\theta - t_i)\right]^2 d\alpha(\theta), \quad \text{with} \quad |\eta| < 1.$$

Hence if $z_1 = \theta_1$ the Gauss-Radau rule yields a lower bound for (58), and if $z_1 = \theta_n$, the rule yields an upper bound.

7. Computational Results. The eigenvalue calculations perform well in practice. Tables 7.1 and 7.2 show the results of two sets of experiments. The Chebyshev semi-iterative method was used with the same unit-length random initial vector and a right hand side of zero. Thus, the problems solved had a zero solution vector.

The matrix resulting from the discretization of the two-dimensional Laplace equation was used for the results in Table 7.1. The values of μ and y are the parameters used to start the iterations (calculated from estimates a and b). The column labeled 'dynamic' shows the number of iterations required to reduce the norm of the error to less than 0.5 x 10^{-4} when the parameters are changed at some point during the execution. The column labeled 'switch' indicates the iteration that the method stopped performing the eigenvalue calculations and switched to the standard CSI method. This point is either when the eigenvalue calculations have "converged" or when the eigenvalue calculations break down (see below). The eigenvalue calculations have converged when μ, calculated from the estimates of the extreme eigenvalues, changes by less than 1.0 x 10^{-6} from the μ calculated with the previous eigenvalue estimates. The column labeled 'fixed' gives the number of iterations that would have been required if the switch were not made, i.e., if the calculations were to continue with the initial choice of parameters. Finally, the columns $\hat{\mu}$ and $\hat{\gamma}$ indicate the parameters used to restart the process at the 'switch' point.

For this problem, the optimal Chebyschev semi-iterative method converges in 213 iterations. The same (optimal) parameters are obtained from the eigenvalue calculations at iteration 135. The desired solution vector is reached at iteration 221. The extra 8 iterations can be accounted for in two ways: the iteration method is restarted at iteration 135, and the calculated parameters are not exact.

Next (row 2) the initial parameters are chosen so that $\mu = 0.975$. Here the eigenvalue calculations break down at iteration 103. The best value of μ at this point is $\hat{\mu}$. Restarting with these parameters $(\hat{\mu}, \gamma_{opt})$ the method converges in

	a	b	μ	γ	dynamic	switch	fixed	$\hat{\mu}$	4
1	a_{opt}	b_{opt}	μ_{opt}	γ_{opt}	221	135	213	μ_{opt}	Yopt
2	0.1	7.9	0.975	γ_{opt}	252	103	1170	0.998775	γ_{opt}
3	0.01	7.99	0.9975	γ_{opt}	226	135	326	μ_{opt}	Yopt
4	0.0001	8.0	0.999975	0.249997	333	135	1448	μ_{opt}	Yopt

Table 7.1: *2D Laplace,* 4096 x 4096 in 64 x 64 *blocks.* $\mu_{opt} \approx 0.996632$, $\gamma_{opt} = 0.25$, $a_{opt} \approx 0.00467$, $b_{opt} \approx 7.99533$.

252 iterations. Note that, even though the eigenvalue calculations broke down, the estimates obtained were still good enough to provide immense savings in the number of iterations performed (252 instead of 1170).

Usually breakdown in the eigenvalue calculations occurs because the value of b_k in (37) becomes negative and we cannot calculate **a value** of β_k for the tridiagonal matrix J_{k+1}. This may be a stability related breakdown. Even though the use of modified moments has *improved* the stability of the determination of the coefficients, it may not be as stable as we would wish (see last paragraph of \$2).

An interesting point is that if the whole process is restarted with the approximate solution at the breakdown point as initial vector, then the eigenvalue calculations typically break down again in short order (usually less than 10 iterations). In most experiments it has proven fruitless to try to restart the eigenvalue calculations after an initial breakdown.

Regardless of whether the eigenvalue calculations complete successfully or not, the process of restarting the method slows down convergence for a few iterations. The basic reason for this is that the building of the Chebyshev polynomials has been broken and the process starts again from a linear Chebyshev polynomial.

In most cases the effect of the new parameters more than makes up for any hesitation caused by restarting.

The eigenvalue calculations complete successfully in the next entry (row 3) at iteration 135. The method converges in 226 iterations instead of the 326 iterations that would have been required if the method were not restarted with the improved parameters.

Finally, row 4 gives an example where the smallest eigenvalue is *under-esfimoted* and the largest eigenvalue is *over-estimated.* The eigenvalue calculations again converge in 135 iterations and both parameters have been correctly calculated. At this point, however, the error vector still has a rather large norm and 333 iterations are required to complete the process. However, if the method had not been restarted with the optimal parameters it would have required 1448 iterations to complete.

The next series of experiments *use* the *Krawtchouk matrix* which is a tridiagonal matrix associated with a discrete set of orthogonal polynomials called Krawtchouk polynomials. The $(n + 1) \times (n + 1)$ Krawtchouk matrix has the following form:

$$K = \begin{vmatrix} \alpha_1 & p & & & & \\ q/n & \alpha_2 & (n-1)p/n & & & \\ & 2q/n & \alpha_3 & (n-2)p/n & & \\ & & \ddots & \ddots & \ddots & \\ & & & \alpha_n & p/n & \\ & & & q & \alpha_{n+1} & \end{vmatrix}$$

where α_k is chosen to make the row sums equal one. This matrix has the property

	a	b	μ	γ	dynamic	switch	fixed	$\hat{\mu}$	$\hat{\gamma}$
1	a_{opt}	b_{opt}	μ_{opt}	γ_{opt}	42	—	42	—	—
2	0.01	1.1	0.981982	1.80180	67	41	102	0.898495	1.80005
3	0.06	1.0	0.886792	1.88679	82	40	> 256	μ_{opt}	γ_{opt}

Table 7.2: ***Krawtchouk matrix,*** *256 x 256,* $\mu_{opt} = 0.90$, $\gamma_{opt} = 1.8$, $a_{opt} = 1/18$, $b_{opt} = 1 + a_{opt}$.

that for $p > 0$, $q = 1 - p$, the $(n + 1)$ eigenvalues of K **are** $\lambda_j = j/n$, for $j = 0, \ldots, n$.

The tests used the 256 x 256 shifted matrix $\hat{K} = K + \frac{1}{18}I$ which has $\mu_{opt} = 0.90$, $\gamma_{opt} = 1.8$. Iterations were continued until the norm of the error was less than 0.5×10^{-8}. The extreme eigenvalues are a = $\frac{1}{18}$ ~0.0555556 and $b = 1 + a$.

The **CSI** method with optimal parameters required 42 iterations to meet the requested tolerance. Using the optimal parameters, the eigenvalue calculations did not converge before the solution was obtained (in 42 iterations).

Row 2 of Table 7.2 shows the results when estimates of $a = 0.01$ and $b = 1.1$ were used. The eigenvalue calculations breakdown at iteration 41 but the method converges at iteration 67 using the best available parameters. If the switch had not been made an additional 35 iterations would have been required to meet the error tolerance.

Finally we chose a = 0.06 and $b = 1.0$ giving the results in row 3 of Table 7.2. Here the eigenvalue calculations converge at iteration 48 with near optimal parameters and only 82 iterations are required to reach the solution. The error at iteration 256 in the fixed case is as large as $\mathcal{O}(10^{-4})$.

Appendix: Variations. Here we derive results for ψ_k chosen **to have the form**

$$\psi_{k+1}(\lambda) = (\lambda - a_k)\psi_k(\lambda) - b_k\psi_k(\lambda).$$

We also present results for Richardson's second order method. Note that Richardson's method can also be used with the techniques in §2 and §3 however §4 is not applicable. After $2m$ iterations of Richardson's method we will only have the roots of $\psi_m(\lambda)$ and so cannot expect to have as good eigenvalue estimates as those obtained with the Chebyshev semi-iterative method.

We start from (18) and define

$$q_k(\lambda) = \frac{p_k(\lambda)}{\omega_k \cdots \omega_2\omega_1} \tag{59}$$

then

$$\omega_{k+1} \bullet \quad **u2wlqk+l(A) = \omega_{k+1}\cdots\omega_2\omega_1\lambda q_k + (1 - \omega_{k+1})\omega_{k-1}\cdots\omega_2\omega_1 q_{k-1}$$

$$q_{k+1}(\lambda) = \lambda q_k + \frac{1 - \omega_{k+1}}{\omega_k\omega_{k+1}} q_{k-1} \tag{60}$$

and so $q_k(\lambda)$ is a **monic** polynomial.

For both the Richardson and Chebyshev methods we have

$$\frac{1 - \omega_{k+1}}{\omega_k\omega_{k+1}} = \frac{\mu^2}{4} \tag{61}$$

where μ is defined in §4. Therefore for both methods we obtain:

$$q_{k+1} = \lambda q_k - \frac{\mu^2}{4} q_{k-1} \tag{62}$$

Thus in terms of a general orthogonal polynomial recurrence, $q_{k+1} = (\lambda - \zeta_k)q_k - \delta_k q_{k-1}$, we have $\zeta_k = 0$ for all k and $\delta_k = \frac{\mu^2}{4}$ for all $k > 1$. δ_1 is a special case since equation (61) does not hold for k = 1. The value of δ_1 for Richardson's method is $\delta_1 = -\frac{1}{2}(\sqrt{1-\mu^2} - 1)$ and for the Chebyshev method $\delta_1 = \mu^2/2$. Now we define new vectors $\tilde{z}_k$ as follows

$$\tilde{z}_0 = \frac{1}{\|z_0\|_2} \times z_0, \quad \tilde{z}_k = \frac{1}{\omega_1 \cdots \omega_k} \times z_k.$$

Similar to the derivation in §2, we now have $\tilde{z}_k = Q_k(B)\tilde{z}_0$ where $\tilde{z}_0 = \sum_{i=1}^{n} \tilde{\alpha}_i q_i$; hence

$$\langle \tilde{z}_l, \tilde{z}_m \rangle = \int q_l(\lambda) q_m(\lambda)\, d\tilde{\alpha}(\lambda).$$

The polynomials $\{q_i(\lambda)\}$ are orthogonal with respect to some measure but not necessarily &(A). The modified moments are given by

$$\nu_k = \int q_k(\lambda) q_0(\lambda)\, d\tilde{\alpha}(\lambda) = \langle \tilde{z}_k, \tilde{z}_0 \rangle.$$

Note that after $2m$ iterations, we have $\{\nu_0, \ldots, \nu_{2m-1}\}$.

To compute the coefficients of $\psi_k(\lambda)$ we use the *modified Chebyshev algorithm* [5]. Define

$$s = \langle z_0, z_0 \rangle, \qquad \nu_k = \langle z_0, \tilde{z}_k \rangle / s, \qquad \text{for } k = 0, 1, \ldots, 2m-1. \tag{63}$$

Note that $\tilde{z}_0 = z_0$ and $\tilde{z}_1 = z_1$. The algorithm is similar to (36) – (37).

Initialization:

$$\begin{aligned} \sigma_{-1,l} &= 0, \qquad \text{for } l = 1, 2, \ldots, 2m-2, \\ \sigma_{0,l} &= \nu_l, \qquad \text{for } l = 0, 1, \ldots, 2m-1, \\ a_0 &= \nu_1, \quad b_0 = 0. \end{aligned} \tag{64}$$

Continuation: for k = $1, 2, \ldots, m-1$

$$\begin{aligned} \sigma_{kl} &= \sigma_{k-1,l+1} - a_{k-1}\sigma_{k-1,l} - b_{k-1}\sigma_{k-2,l} + \delta_l \sigma_{k-1,l-1} \\ &\qquad \text{for } l = k, k+1, \ldots, 2m-k-1, \\ a_k &= \frac{\sigma_{k,k+1}}{\sigma_{kk}} - \frac{\sigma_{k-1,k}}{\sigma_{k-1,k-1}}, \quad b_k = \frac{\sigma_{kk}}{\sigma_{k-1,k-1}}. \end{aligned} \tag{65}$$

Improvement for the Chebyshev Algorithm. Once again we can use the techniques of §5 for the Chebyshev algorithm. From [10] we have

$$\omega_{k+1} = \frac{2C_k(1/\mu)}{\mu C_{k+1}(1/\mu)} \tag{66}$$

and

$$\omega_k \cdots \omega_1 = \left(\frac{2^{k-1}}{\mu^k}\right) \frac{1}{C_k(1/\mu)}. \tag{67}$$

Using (39) and (67) together with (59) we obtain

$$q_k(\lambda) = \frac{\mu^k}{2^{k-1}} C_k(\lambda/\mu), \tag{68}$$

which together with (40) give

$$q_{2k} = q_k^2 - 2q_0\left(\frac{\mu}{2}\right)^{2k}, \quad q_{2k+1} = q_k q_{k+1} - q_1\left(\frac{\mu}{2}\right)^{2k}, \tag{69}$$

which leads to the modified moments

$$\nu_{2k} = \langle \tilde{z}_k, \tilde{z}_k\rangle/s - 2\left(\frac{\mu}{2}\right)^{2k}, \quad \nu_{2k+1} = \langle \tilde{z}_k, \tilde{z}_{k+1}\rangle/s - \nu_1\left(\frac{\mu}{2}\right)^{2k}. \tag{70}$$

Here $\tilde{z}_k$ and s are given by (63), $\nu_0 = 1$ and $\nu_1 = \langle \tilde{z}_0, \tilde{z}_1\rangle/s$. Thus it is possible to obtain two moments with each iteration of (3).

Problems with this approach. It is clear from (67) that the product of all the ω_k's will increase exponentially with k. In (63) we divide the vector z_k by this product thus the moments are rapidly approaching zero as k increases. On a Vax 11/780 using double precision arithmetic we were never able to go beyond about 60 steps of (65) because the diagonal elements of the σ array, σ_{kk}, rapidly converged to zero. With the choice of polynomials (21) this problem is avoided, but it is still possible to obtain very small σ_{kk}.

Acknowledgements The first author wishes to acknowledge that he was prodded into this research after comments by Prof. R. Sacks. Dr. Roland Freund made helpful suggestions concerning §6.2.

REFERENCES

[1] N.I. AHEIZER, M. KREIN, *Some* questions in *the Theory of Moments,* Amer. Math. Soc., Providence, Rhode Island, (1962).

[2] P. CONCUS, G.H. GOLUB, D.P. O'LEARY, "A *Generalized Conjugate* Gradient *Method for the Numerical Solution of Elliptic Partial Differential Equations,*" in *Studies in Numerical Analysis,* (G.H. Golub, ed.), MAA Studies in Mathematics, Volume 24, pp. 178-198, (1985).

[3] G.G. DAHLQUIST, S.C. EISENSTAT, G.H. GOLUB, *"Bounds for the Error of Linear Systems of* Equations *Using the Theory of Moments,"* J. of Math. Anal. Appl., 37(1), pp. 151-166, (1972).

[4] G.G. DAHLQUIST, G.H. GOLUB, S.G. NASH, *"Bounds for the Error in Linear Systems,"* in *Proc.* Of *the Workshop on Semi-Infinite Programming,* ed. R. Hettich, Springer Verlag, pp. 154172, (1978).

[5] W. GAUTSCHI, *"On Generating Orthogonal Polynomials,"* Siam J. Sci. Statist. Comput., 3(3), pp. 289-317, (1982).

[6] ———, *"Questions of* Numerical *Condition Related to Polynomials,"* in *Studies in Numerical Analysis,* (G.H. Golub ed.), MAA Studies in Mathematics, Volume 24, pp. 140-177, (1985).

[7] G.H. GOLUB, "Some *Modified* Matrix *Eigenvalue* Problems," SIAM Rev., 15, 2, (1973), pp. 318-334.

[8] ———, *"Some Uses of the Lanczos Algorithm in Numerical* Linear Algebra," *Topics in Numerical Analysis,* ed. J. Miller, Academic Press, (1974).

[9] G.H. GOLUB, C.F. VAN LOAN, Matrix *Computations,* Johns Hopkins University Press, Baltimore, (1983).

[10] G.H. GOLUB, R.S. VARGA, *"Chebyschev semi-iterative methods, successive overrelaxation methods, and second order Richardson iterative methods,"* Numer. Math., 3, pp. 147-,168, (1961).

[11] G.H.. GOLUB, J.H. WELSCH, *"Calculation of Gauss quadrature rules,"* Math. Comp., 23, pp. 221-30, (1969).

[12] A.S. HOUSEHOLDER, *The 'Theory of Matrices in Numerical Analysis,* Blaisdell, New York, (1964).

[13] C. LANCZOS, *"An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators,"* J. Res. Nat. Bur. Standards, **45(4),** pp. 255-282, (1950).

[14] R.A. SACK, A.F. DONOVAN, *"An algorithm for* Gaussian *Quadrature given modified moments,"* Numer. **Math., 18,** pp. 465-478, (1971-72).

[15] J.C. WHEELER, *"Modified* **Moments** *and Gaussian* **Quadrature,"** **Rocky Mountain J. Math., 4(2), pp. 287-296, (1974).**

[16] J.H. WILKINSON, *The Algebraic Eigenvalue Problem,* Clarendon Press, Oxford, (1965).