Matrices, moments and quadrature

Gérard MEURANT

September 2009
\([a, b] = \text{finite or infinite interval of the real line}\)

We will use \textit{Riemann–Stieltjes} integrals of a real valued function \(f\) of a real variable with respect to a real function \(\alpha\) which are denoted by

\[
\int_a^b f(\lambda) \, d\alpha(\lambda)
\]  

(1)

In many cases \textit{Riemann–Stieltjes} integrals are directly written as

\[
\int_a^b f(\lambda) \, w(\lambda) \, d\lambda
\]

where \(w\) is called the weight function
Moments

Let $\alpha$ be a nondecreasing function on the interval $(a, b)$ having finite limits at $\pm \infty$ if $a = -\infty$ and/or $b = +\infty$

**Definition**
The numbers

$$\mu_i = \int_a^b \lambda^i \, d\alpha(\lambda), \quad i = 0, 1, \ldots$$

are called the *moments* related to the measure $\alpha$
A little bit of history

Famous names are associated with moment problems: Chebyshev, Markov, Stieltjes, Hamburger, Hausdorff.


He proposed and solved the following problem:
Find a bounded non-increasing function \( \alpha \) in the interval \([0, \infty)\) such that its moments have a prescribed set of values \( \mu_n \)

\[
\int_0^\infty \lambda^n \, d\alpha(\lambda) = \mu_n, \quad n = 0, 1, 2, \ldots
\]

The name “problem of moments” was chosen by Stieltjes in analogy with mechanical problems.
A little bit of history 2

see the books by
Shohat and Tamarkin
*The problem of moments*, American Mathematical Society, (1943)
and
Akhiezer
Russian edition 1961
A little bit of history 3

Moment problems in numerical linear algebra (related to the estimation of error norms):


The first paper considers a sequence of Krylov vectors
\[ r^{i+1} = Ar^i, \ i = 0, 1, \ldots k - 1 \] (this may not be such a good idea numerically) and looks at the moments
\[ (r^i, r^j) = (A^{i+j} r^0, r^0) = \mu_{i+j} \]

Assume \( A \) is SPD, given \( \mu_i, \ i = 0, \ldots, 2k \) how do we compute bounds for \( \mu_{-2} = (A^{-2} r^0, r^0) \)?

We will see in a moment why they were interested in that problem.

In these 2 papers, the authors used beautiful relationships between matrices, moments, orthogonal polynomials, quadrature, \ldots
Examples of applications 1

Solve

\[ Ax = b \]

Let \( x^k \) be an approximate solution and \( r^k = b - Ax^k \) be the residual vector.

Since \( \| r^k \| \) is often misleading for stopping iterative methods, it is of interest to obtain bounds or estimates of norms of the error.

\[ \epsilon^k = x - x^k \]

We have

\[ A\epsilon^k = r^k \]
Therefore,

\[
\| \epsilon_k \|_A^2 = (A \epsilon_k, \epsilon_k) = (A^{-1} r_k, r_k) = (r_k)^T A^{-1} r_k \quad (A \text{ SPD})
\]

\[
\| \epsilon_k \|_2^2 = (r_k)^T A^{-2} r_k
\]

We have to consider quadratic forms \( u^T f(A) u \) with \( u = r_k \) and \( f(x) = 1/x \) or \( 1/x^2 \)
Examples of applications 2

Ill-posed problems
We want to solve

$$Ax = y$$

where $A$ ($m \times n$ matrix) arises from the discretization of an inverse problem (Fredholm integral equation of the first kind)
Generally, the right hand side is corrupted with (an unknown) noise

$$y = \hat{y} + e$$

The matrix $A$ may have very small singular values

Tikhonov regularization

$$(A^TA + \nu I)x = A^Ty$$
How to choose the regularization parameter $\nu$?

**Generalized Cross Validation (GCV):**
see G.H. Golub, M. Heath and G. Wahba (1979)
find the minimum of

$$G(\nu) = \frac{1}{m} \| (I - A(A^T A + \nu I)^{-1} A^T) y \|^2 \left( \frac{1}{m} \text{tr}(I - A(A^T A + \nu I)^{-1} A^T) \right)^2$$

**L-curve:** find the “corner” of $\log(\|x(\nu)\|)$ as a function of $\log(\|y - Ax(\nu)\|)$

Easy to solve if we know the SVD of $A$, not feasible if the matrix is large
In these methods and others we need to compute

\[ y^T A (A^T A + \nu I)^{-p} A^T y \]

and/or

\[ y^T (AA^T + \nu I)^{-p} y \]

\( p = 1, 2, 3, 4, \) for given \( \nu \) and \( y \)
Examples of applications

Total least squares (TLS)
see G.H. Golub and C. Van Loan (1980)

We look for the solution of

\[
\min_{E,r} \|(E, r)\|_F
\]

subject to

\[
(A + E)x = b + r, \quad A : m \times n
\]
The minimum is given by $\sigma_{n+1}$ (the smallest singular value of $(A\ b)$) which is the solution of the secular equation

$$\sigma_{n+1}^2 = b^T b - b^T A(A^T A - \sigma_{n+1}^2 I)^{-1} A^T b$$

This is the same type of function as in GCV.
Other examples

Rank one change for eigenvalues

\[ Ax = \lambda x, \quad (A + c^T c)y = \mu y \]

Secular equation:

\[ 1 + c^T (A - \mu I)^{-1} c = 0 \]

Quadratic constraint

\[ A = A^T, \quad \min_x x^T Ax - 2b^T x, \quad x^T x = \alpha^2 \]

Secular equation:

\[ b^T (A - \mu I)^{-2} b = \alpha^2 \]
Relation to quadrature

Assume $A$ is symmetric positive definite

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

$\Lambda$ diagonal

Consider

$$u^T f(A) u, \quad f(A) = Q f(\Lambda) Q^T$$

We write the quadratic form as a Riemann–Stieltjes integral

$$I[f] = u^T f(A) u = \int_a^b f(\lambda) \, d\alpha(\lambda)$$
the measure $\alpha$ is piecewise constant
if $y = Q^T u$

$$\alpha(\lambda) = \begin{cases} 
0 & \text{if } \lambda < a = \lambda_1, \\
\sum_{j=1}^{i} y_j^2 & \text{if } \lambda_i \leq \lambda < \lambda_{i+1}, \\
\sum_{j=1}^{n} y_j^2 & \text{if } b = \lambda_n \leq \lambda
\end{cases}$$

$\lambda_i$ are the eigenvalues of $A$ that we usually don't know
We would like to approximate or to bound the integral by using Gauss quadrature rules
Gauss quadrature

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

the weights \([w_j]_{j=1}^{N}, [v_k]_{k=1}^{M}\) and the nodes \([t_j]_{j=1}^{N}\) are unknowns
and the nodes \([z_k]_{k=1}^{M}\) are prescribed

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

\[ a < \eta < b \]
What is important is the sign of the remainder

Gauss rule: \( M = 0 \) no prescribed nodes

Suppose \( f^{(2n)}(\xi) > 0, \forall n, \forall \xi, a < \xi < b \), then

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

\( L_G[f] \leq I[f] \)
Gauss–Radau rule: \( M = 1 \) (one node is prescribed), \( z_1 = a \) or \( z_1 = b \). Suppose \( f^{(2n+1)}(\xi) < 0, \forall n, \forall \xi, a < \xi < b \), then

\[
U_{GR}[f] = \sum_{j=1}^{N} w^a_j f(t_j^a) + v_1^a f(a), \quad z_1 = a
\]

\[
L_{GR}[f] = \sum_{j=1}^{N} w^b_j f(t_j^b) + v_1^b f(b), \quad z_1 = b
\]

\[L_{GR}[f] \leq l[f] \leq U_{GR}[f],\]
Gauss–Lobatto rule: $M = 2$ (two nodes are prescribed), $z_1 = a, z_2 = b$

Suppose $f^{(2n)}(\xi) > 0$, $\forall n$, $\forall \xi$, $a < \xi < b$, then

$$U_{GL}[f] = \sum_{j=1}^{N} w_{j}^{GL} f(t_{j}^{GL}) + v_{1} f(a) + v_{2} f(b)$$

$$I[f] \leq U_{GL}[f],$$
Computation of nodes and weights

Relation to orthogonal polynomials, see G.H. Golub and J.H. Welsch (1969)

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

They satisfy a three–term recurrence

\[ \gamma_j p_j(\lambda) = (\lambda - \omega_j)p_{j-1}(\lambda) - \gamma_{j-1}p_{j-2}(\lambda), \quad j = 1, 2, \ldots, N \]
In matrix form

$$\lambda P(\lambda) = J_N P(\lambda) + \gamma_N p_N(\lambda)e_N$$

$$P(\lambda)^T = [p_0(\lambda) \ p_1(\lambda) \cdots \ p_{N-1}(\lambda)]$$

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

The nodes of the Gauss rule are the eigenvalues of $J_N$ and the weights are the squares of the first elements of the normalized eigenvectors.
To obtain the Gauss–Radau rule ($M = 1$), we extend the matrix $J_N$ such that it has one prescribed eigenvalue ($a$ or $b$). This is an inverse eigenvalue problem, see G.H. Golub (1973)

\[
\hat{J}_{N+1} = \begin{pmatrix} J_N & \gamma_N e_N \\ \gamma_N (e_N)^T & \omega_{N+1} \end{pmatrix}
\]

We compute $\omega_{N+1}$ by

\[
\omega_{N+1} = a - \gamma_N \frac{p_{N-1}(a)}{p_N(a)}
\]

\[
(J_N - al) \delta(a) = \gamma_N^2 e_N, \quad \omega_{N+1} = a + \delta_N(a)
\]
We do something similar for Gauss–Lobatto

\[(J_N - al)\delta = e_N, \quad (J_N - bl)\mu = e_N\]

\[
\begin{pmatrix}
1 & -\delta_N \\
1 & -\mu_N
\end{pmatrix}
\begin{pmatrix}
\omega_{N+1} \\
\gamma^2_N
\end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}
\]
We can compute the nodes and weights by using Golub and Welsch, but this is not always necessary.

We have

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

where \( e_1 \) is the first column of the identity matrix.

Sometimes, we can compute the (1,1) element of \( f(J_N) \) efficiently (example: \( f(x) = 1/x \)).
Computation of the orthogonal polynomials

Suppose $\|u\| = 1$ then the matrix $J_N$ is computed by $N$ iterations of the Lanczos algorithm starting from $v^1 = u$

$$\gamma_k = \|\tilde{v}^k\|,$$

$$v^k = \frac{\tilde{v}^k}{\eta_k},$$

$$\omega_k = (v^k, Av^k) = (v^k)^T Av^k,$$

$$\tilde{v}^{k+1} = Av^k - \omega_k v^k - \gamma_k v^{k-1}.$$
The algorithm to compute bounds of $u^T f(A) u$

Suppose the derivatives of $f$ have constant signs, then

- do Lanczos iterations from $u/\|u\|$ to compute $J_i$
- compute $e_1^T f(J_i)e_1$ or $e_1^T f(\hat{J}_i)e_1$ to obtain bounds

or compute the eigenvalues and (first components of) eigenvectors of $J_i$
Bilinear forms

To estimate $u^T f(A)v$ when $u \neq v$ we can use

- $u^T f(A)v = [(u + v)^T f(A)(u + v) - (u - v)^T f(A)(u - v)]/4$
- the non-symmetric Lanczos algorithm
- the block Lanczos algorithm

see G.H. Golub and G. Meurant (1994)
Reprinted in
Milestones in Matrix Computations, the selected works of Gene H. Golub with commentaries, R.H. Chan, C. Greif and D.P. O’Leary Eds, Oxford University Press, (2007)
The conjugate gradient algorithm

What to do for CG?

It does not make sense to do Lanczos iterations starting from \( r^k / \| r^k \| \)

However, since \( A \epsilon^k = r^k = r^0 - AV_k y^k \) and \( J_k y^k = \| r^0 \| ^2 e^1 \)

\[
\| \epsilon^k \|^2_A = \| r^0 \|^2 \left[ (J_n^{-1} e^1, e^1) - (J_k^{-1} e^1, e^1) \right]
\]

Note that \( \| r^0 \|^2 (J_n^{-1} e^1, e^1) = (A^{-1} r^0, r^0) \)

Hence, \( \| \epsilon^k \|^2_A \) is the remainder of Gauss quadrature for the Riemann–Stieltjes integral \( (A^{-1} r^0, r^0) \)
\[ \| \epsilon^k \|_A^2 = \frac{1}{\xi_{2k+1}} \sum_{i=1}^{n} \left[ \prod_{j=1}^{k} (\lambda_i - \theta_j^{(k)})^2 \right] (r^0, q^i)^2, \]

where \( q^i \) is the \( i \)th eigenvector of \( A \) corresponding to \( \lambda_i, \theta_j^{(k)} \) Ritz values (eigenvalues of \( J_k \)), \( a \leq \xi \leq b \).

The formula for \( \| \epsilon^k \|_A^2 \) is equivalent to a formula proved in Hestenes and Stiefel (1952):

\[ \| \epsilon^k \|_A^2 = \sum_{j=k}^{n-1} \gamma_j \| r^k \|_2^2 \]

\( \gamma_j \) is one of the two CG parameters.
Approximation of the norm of the error

Of course, we do not know \((J_n^{-1}e_1, e_1)\)

Let \(d\) be a positive integer, at iteration \(k\) we use

\[
\|\epsilon^{k-d}\|_A^2 \simeq \|r_0\|^2 \left[ (J_k^{-1}e_1, e_1) - (J_{k-d}^{-1}e_1, e_1) \right]
\]

or

\[
\|\epsilon^{k-d}\|_A^2 \simeq \sum_{j=k-d+1}^{k} \gamma_j \|r^j\|^2
\]

The first formula can also be used with Gauss–Radau or Gauss–Lobatto rules to obtain upper bounds.
If we want a lower bound for $\|e^{k-d}\|_A^2$ we use the H-S formula.

If we have an estimate of the smallest eigenvalue, we compute $(\hat{J}_k^{-1})_{1,1}$ incrementally by using the Sherman–Morrison formula to obtain an upper bound, see Meurant (1997, 1999).

Strakoš and Tichý (2002) have proved that these formulas work also in finite precision arithmetic.

Arioli (2004) and Arioli, Loghin and Wathen (2005) have used these techniques to provide reliable stopping criteria for finite element problems.
Elements of $f(A)$

Finite difference approximation of the Poisson equation on a $16 \times 16$ mesh ($n = 256$)
We look for $(A^{-1})_{125,125}$ whose value is 0.5604

<table>
<thead>
<tr>
<th>rule</th>
<th>Nit=2</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>10</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>0.3333</td>
<td>0.4337</td>
<td>0.4920</td>
<td>0.5201</td>
<td>0.5378</td>
<td>0.5600</td>
</tr>
<tr>
<td>G–R $b_L$</td>
<td>0.3639</td>
<td>0.4514</td>
<td>0.5006</td>
<td>0.5255</td>
<td>0.5414</td>
<td>0.5601</td>
</tr>
<tr>
<td>G–R $b_U$</td>
<td>1.5208</td>
<td>0.8154</td>
<td>0.6518</td>
<td>0.5925</td>
<td>0.5730</td>
<td>0.5604</td>
</tr>
<tr>
<td>G–L</td>
<td>2.1011</td>
<td>0.8983</td>
<td>0.6803</td>
<td>0.6012</td>
<td>0.5760</td>
<td>0.5604</td>
</tr>
</tbody>
</table>
Block Lanczos, $m = 6$, $n = 36$

$(A^{-1})_{2,1} = 0.1040$

<table>
<thead>
<tr>
<th>rule</th>
<th>Nit=2</th>
<th>4</th>
<th>6</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>0.0894</td>
<td>0.1008</td>
<td>0.1033</td>
<td>0.1040</td>
</tr>
<tr>
<td>G–R $b_L$</td>
<td>0.0931</td>
<td>0.1017</td>
<td>0.1035</td>
<td>0.1040</td>
</tr>
<tr>
<td>G–R $b_U$</td>
<td>0.1257</td>
<td>0.1059</td>
<td>0.1042</td>
<td>0.1040</td>
</tr>
<tr>
<td>G–L</td>
<td>0.1600</td>
<td>0.1079</td>
<td>0.1041</td>
<td>0.1041</td>
</tr>
</tbody>
</table>
Larger example

Block Lanczos, $m = 30, n = 900$

$(A^{-1})_{1,1} = 0.302346, (A^{-1})_{2,2} = 0.344408, (A^{-1})_{2,1} = 0.104693$

Results after 10 block iterations for Gauss:

$$
\begin{pmatrix}
0.3021799137963044 & 0.1043616568803480 \\
0.1043616568803480 & 0.3437475221129595
\end{pmatrix}
$$

Results after 10 block iterations for Gauss–Radau with exact eigenvalues:

$$
\begin{pmatrix}
0.3022010722636479 & 0.1044036770842950 \\
0.1044036770842950 & 0.3438314340061286
\end{pmatrix}
$$

$$
\begin{pmatrix}
0.3039414302035057 & 0.1078375193911064 \\
0.1078375193911064 & 0.3506698361080970
\end{pmatrix}
$$
Exponential of $A$

Finite difference approximation of the Poisson equation on a $30 \times 30$ mesh

We look for $(\exp(A))_{18,18}$ whose value is 197.9724768113708 using Gauss quadrature

After 5 iterations: 197.9599617609761

After 10 iterations: 197.9724768113530
CG error norm

Matrix Bcsstk01, \( n = 48 \)

\[
A \text{ norm (blue), Gauss quad } d = 1 \text{ (red)}
\]
Matrix Bcsstk01, $n = 48$

A norm (blue), Gauss quad $d = 10$ (red)
PCG error norm

Elliptic problem, diff. coeff. = \( \frac{1}{(2+p \sin \frac{x}{\eta})(2+p \sin \frac{y}{\eta})} \)

\( p = 1.99 \) and \( \eta = 0.01 \), mesh \( 100 \times 100 \), IC(0)

A norm (blue), Gauss quad \( d = 1 \) (red)
Ill–posed problems (Tikhonov using GCV)

The matrix to consider is \( B = A^T A \) or \( B = AA^T \)

We use the Golub–Kahan bidiagonalization algorithm (1965) which produces a lower bidiagonal matrix \( C_k \)

We have to compute quantities like

\[
I[C_k] = (e_1^T) (C_k^T C_k + \nu I)^{-p} e_1
\]

This can be done by solving least squares problems or by using the SVD of \( C_k \)

For computing the trace we use a result of Hutchinson (1989)

\[
tr[(AA^T + \nu I)^{-1}] \approx \frac{1}{q} \sum_{i=1}^{q} (u_i^T)(AA^T + \nu I)^{-1} u_i
\]

where \( u_i \) are random vectors. In practice, \( q = 1 \)
Problem: Baart, ReguTools (Hansen), $n = 100$

functions $G$ and $\tilde{G}$, Baart, $m = n = 100$, $\|e\| = 10^{-3}$
functions $G$, $\tilde{G}$ and upper bound, Baart, $m = n = 100$, $\|e\| = 10^{-3}$
Notice that

- We do not want to compute the bounds for too many points $\nu_i$
- It would be nice to know that the upper bound has “converged” before looking for the minimum
- The upper bound does not have the right asymptotic behavior when $\nu \to 0$

We modify the function for the upper bound, instead of $p(\nu)/q(\nu)^2$, we consider

$$\frac{p(\nu)}{q(\nu)^2 + \|y\|^2}$$
functions $G$, $\tilde{G}$ and modified upper bound, Baart,

\[ m = n = 100, \|e\| = 10^{-3} \]
• We test the convergence of the upper bounds for a small value of $\nu$
• We compute the minimum
• We test its convergence
• Functions values are computed using SVDs of $C_k$
### Baart

<table>
<thead>
<tr>
<th>$|e|$</th>
<th>nb it</th>
<th>$\nu/m$</th>
<th>$f_{\text{min}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-7}$</td>
<td>12</td>
<td>$1.0706 \times 10^{-14}$</td>
<td>436</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>12</td>
<td>$1.0581 \times 10^{-11}$</td>
<td>437</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>8</td>
<td>$1.3077 \times 10^{-8}$</td>
<td>293</td>
</tr>
<tr>
<td>$10^{-1}$</td>
<td>7</td>
<td>$1.1104 \times 10^{-7}$</td>
<td>294</td>
</tr>
<tr>
<td>$10$</td>
<td>7</td>
<td>$9.1683 \times 10^{-8}$</td>
<td>294</td>
</tr>
<tr>
<td>$| e |$</td>
<td>nb it</td>
<td>$\nu/m$</td>
<td>$f$ min</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>-----------------</td>
<td>--------</td>
</tr>
<tr>
<td>$10^{-7}$</td>
<td>58</td>
<td>$4.2396 \times 10^{-14}$</td>
<td>5239</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>28</td>
<td>$5.4552 \times 10^{-11}$</td>
<td>1453</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>17</td>
<td>$2.3046 \times 10^{-8}$</td>
<td>440</td>
</tr>
<tr>
<td>$10^{-1}$</td>
<td>15</td>
<td>$2.0896 \times 10^{-5}$</td>
<td>293</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>$6.8436 \times 10^{-3}$</td>
<td>296</td>
</tr>
</tbody>
</table>
x₀ is the noise free exact solution
Another application: computing the trace of the inverse

Let $A$ be symmetric (positive definite for the sake of simplicity)

There are applications in physics where it is desired to compute bounds or estimates of the trace of the inverse $tr(A^{-1})$ and/or the determinant $\det(A)$ of large sparse matrices

Let $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ be the eigenvalues of $A$

We have

$$tr(A^r) = \sum_{i=1}^{n} \lambda_i^r$$

We are interested in $r = -1$
As we have seen the sum $\sum_{i=1}^{n} \lambda_i^r$ can be written as a Riemann–Stieltjes integral

$$tr(A^r) = \mu_r = \int_{a}^{b} \lambda^r \, d\alpha,$$

where $a \leq \lambda_1, \lambda_n \leq b$

where the (unknown) measure $\alpha$ is given as

$$\alpha(\lambda) = \sum_{j=1}^{n} H(\lambda - \lambda_j)$$

$H$ is the unit step function, $H(\lambda) = 0$, $\lambda < 0$, $H(\lambda) = 1$, $\lambda \geq 0$

The values $\mu_r$ are the moments related to $\alpha$

We wish to compute

$$\mu_{-1} = \int_{a}^{b} \frac{1}{\lambda} \, d\alpha$$
Bai and Golub results

As before use Gauss quadrature to estimate or bound the integral

Bai and Golub (1997) use three moments $r = 0, 1, 2$

$$\mu_0 = n, \quad \mu_1 = \text{tr}(A) = \sum_{i=1}^{n} a_{i,i}, \quad \mu_2 = \text{tr}(A^2) = \sum_{i,j=1}^{n} a_{i,j}^2 = \|A\|_F^2$$

to analytically compute the nodes and weights of a Gauss–Radau rule and then bounds for the integral of $1/\lambda$
Bai and Golub results 2

Their result is

\[
(\mu_1 \ n) \begin{pmatrix} \mu_2 & \mu_1 \\ b^2 & b \end{pmatrix}^{-1} \begin{pmatrix} n \\ 1 \end{pmatrix} \leq \text{tr}(A^{-1}) \leq (\mu_1 \ n) \begin{pmatrix} \mu_2 & \mu_1 \\ a^2 & a \end{pmatrix}^{-1} \begin{pmatrix} n \\ 1 \end{pmatrix}
\]

This result is nice since the moments \( \mu_0, \mu_1, \mu_2 \) are easy to compute, but in many cases, the bounds are far from being sharp.
Associated with the measure $\alpha$ there exist orthonormal polynomials $p_k$

They satisfy a three-term recurrence

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

The Jacobi matrix is

$$J_k = \begin{pmatrix}
\omega_1 & \gamma_1 & & \\
\gamma_1 & \omega_2 & \gamma_2 & \\
& \ddots & \ddots & \ddots \\
& & \gamma_{k-2} & \omega_{k-1} & \gamma_{k-1} \\
& & & \gamma_{n-1} & \omega_k
\end{pmatrix}$$
The main question is:

Can we compute the \textit{Jacobi} matrix from the moments?

If we can do this, we have the algorithm:

\texttt{moments (r \geq 0) \Rightarrow Jacobi matrix \Rightarrow eigensystem \Rightarrow nodes and weights \Rightarrow estimate for r = -1}
The Chebyshev algorithm

An answer to our question has been given by Chebyshev (1859) who described an algorithm to obtain the coefficients of the orthogonal polynomials from the moments.

One needs $2k$ moments to compute $J_k$.

The algorithm (as it is described today) uses the Cholesky factorization of the (Hankel) moment matrix $m_{i,j} = \mu_{i+j-2}$.

However, the map from the moments to the coefficients is ill-conditioned (see Gautschi) and the Chebyshev algorithm is often unstable.
The modified Chebyshev algorithm

This algorithm was developed by J. Wheeler in 1974, see also Sack and Donovan (1972)

Let $\pi_k$ be a family of known orthogonal polynomials satisfying

$$b_{k+1}\pi_{k+1}(\lambda) = (\lambda - a_{k+1})\pi_k(\lambda) - c_k\pi_{k-1}(\lambda)$$

The modified moments are

$$m_l = \int_a^b \pi_l(\lambda) \, d\alpha$$

which have to be known

The algorithm uses mixed moments which are

$$\sigma_{k,l} = \int_a^b p_k(\lambda)\pi_l(\lambda) \, d\alpha(\lambda)$$
Implementation

As auxiliary polynomials, we use the shifted Chebyshev polynomials:

\[ C_0(\lambda) \equiv 1, \quad \left( \frac{\lambda_n - \lambda_1}{2} \right) C_1(\lambda) = \lambda - \left( \frac{\lambda_n + \lambda_1}{2} \right) \]

\[ \left( \frac{\lambda_n - \lambda_1}{4} \right) C_{k+1}(\lambda) = \left( \lambda - \frac{\lambda_n + \lambda_1}{2} \right) C_k(\lambda) - \left( \frac{\lambda_n - \lambda_1}{4} \right) C_{k-1}(\lambda) \]

Computing the modified moment \( m_l \) is computing the trace of the matrix \( C_l(A) \)

Pb: we have to compute the product of “sparse” matrices, but we have to store only the last 2 of them
Numerical experiments

Example: Poisson equation

\( n = 36, \text{tr}(A^{-1}) = 13.7571 \), Bai and Golub bounds

\[ 10.2830 \leq \text{tr}(A^{-1}) \leq 24.3776 \]

Moments

<table>
<thead>
<tr>
<th>( k )</th>
<th>bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9.0000</td>
</tr>
<tr>
<td>2</td>
<td>11.3684</td>
</tr>
<tr>
<td>3</td>
<td>12.5714</td>
</tr>
<tr>
<td>4</td>
<td>13.1581</td>
</tr>
<tr>
<td>5</td>
<td>13.4773</td>
</tr>
<tr>
<td>6</td>
<td>13.6363</td>
</tr>
<tr>
<td>7</td>
<td>13.7139</td>
</tr>
<tr>
<td>8</td>
<td>13.7452</td>
</tr>
<tr>
<td>9</td>
<td>13.7550</td>
</tr>
<tr>
<td>10</td>
<td>13.7568</td>
</tr>
</tbody>
</table>
After $k = 10$ the moment matrices are no longer positive definite

**Modified Moments**

<table>
<thead>
<tr>
<th>$k$</th>
<th>bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9.0000</td>
</tr>
<tr>
<td>2</td>
<td>11.3684</td>
</tr>
<tr>
<td>3</td>
<td>12.5714</td>
</tr>
<tr>
<td>4</td>
<td>13.1581</td>
</tr>
<tr>
<td>5</td>
<td>13.4773</td>
</tr>
<tr>
<td>6</td>
<td>13.6363</td>
</tr>
<tr>
<td>7</td>
<td>13.7139</td>
</tr>
<tr>
<td>8</td>
<td>13.7452</td>
</tr>
<tr>
<td>9</td>
<td>13.7550</td>
</tr>
<tr>
<td>10</td>
<td>13.7568</td>
</tr>
<tr>
<td>11</td>
<td>13.7571</td>
</tr>
</tbody>
</table>
\( n = 900, \quad tr(A^{-1}) = 512.6442, \quad \text{Bai and Golub bounds} \)

\[ 261.003 \leq tr(A^{-1}) \leq 8751.76 \]

The \text{Chebyshev} algorithm breaks down after \( k = 10 \)

### Modified Moments

<table>
<thead>
<tr>
<th>( k )</th>
<th>bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>400.0648</td>
</tr>
<tr>
<td>10</td>
<td>463.2560</td>
</tr>
<tr>
<td>15</td>
<td>489.5383</td>
</tr>
<tr>
<td>20</td>
<td>502.0008</td>
</tr>
<tr>
<td>25</td>
<td>508.0799</td>
</tr>
<tr>
<td>30</td>
<td>510.9301</td>
</tr>
<tr>
<td>35</td>
<td>512.1385</td>
</tr>
<tr>
<td>40</td>
<td>512.5469</td>
</tr>
</tbody>
</table>
We can avoid computing the matrices $C_i(A)$ and then their traces by using a Monte Carlo technique:
Use random vectors $z$ whose components are 1 and $-1$ with probability $1/2$, then $z^T B z$ is an estimator of $\text{tr}(B)$.
One just needs to compute $C_i(A)z$ and an inner product.

The same techniques can be used to estimate $\det(A)$ by remarking that $\det(A) = \exp[\text{tr}(\ln(A))]$. 
There is a forthcoming book which should appear soon:

G.H. Golub and G. Meurant
Matrices, moments and quadrature with applications
Princeton University Press

for details see http://pagesperso-orange.fr/gerard.meurant
Gene H. Golub (1932–2007)