Bounds for bilinear forms $u^T f(A)v$

Gérard MEURANT

October, 2008

1. Introduction
2. The case $u = v$
3. The case $u \neq v$
4. The block case

Introduction

We are interested in computing bounds or approximations for bilinear forms

$$u^T f(A) v$$

where A is a symmetric square matrix of order n, u and v are given vectors and f is a smooth (possibly C^∞) function on a given interval of the real line

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

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

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

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

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

$$
I[f] = u^T f(A) v = \int_a^b f(\lambda)\ d\alpha(\lambda)
$$

where the measure α is piecewise constant and defined by

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

The case $u = v$

When $u = v$, we remark that α is an increasing positive function

How do we generate the Jacobi matrix corresponding to the measure α which is unknown?

The algorithm is the following:

- normalize u if necessary to obtain v^1
- run k iterations of the Lanczos algorithm with A starting from v^1, compute the Jacobi matrix J_k
- if we use the Gauss–Radau or Gauss–Lobatto rules, modify J_k to $\tilde{J}_k$ accordingly. For the Gauss rule $\tilde{J}_k = J_k$
- if this is feasible, compute $(e^1)^T f(\tilde{J}_k) e^1$. Otherwise, compute the eigenvalues and the first components of the eigenvectors using the Golub and Welsch algorithm to obtain the approximations from the Gauss, Gauss–Radau and Gauss–Lobatto quadrature rules

Let n be the order of the matrix A and V_k be the $n \times k$ matrix whose columns are the Lanczos vectors
If A has distinct eigenvalues, after n Lanczos iterations we have $AV_n = V_n J_n$
If Q (resp. Z) is the matrix of the eigenvectors of A (resp. J_n) we have the relation $V_n Z = Q$

$$u^T f(A) u = (e^1)^T V_n^T Q f(\Lambda) Q^T V_n e^1 = (e^1)^T Z^T f(\Lambda) Z e^1 = (e^1)^T f(J_n) e^1$$

$$R[f] = (e^1)^T f(J_n) e^1 - (e^1)^T f(J_k) e^1$$

The convergence of the Gauss quadrature approximation to the integral depends on the convergence of the Ritz values to the eigenvalues of A

Preconditioning

The convergence rate can be improved in some cases by preconditioning

If we are interested in $u^T A^{-1} u$ and if we have a preconditioner $M = LL^T$ for A

$$u^T A^{-1} u = u^T L^{-T} (L^{-1} A L^{-T})^{-1} L^{-1} u$$

$L^{-1} A L^{-T}$ is the preconditioned matrix to which we apply the Lanczos algorithm

The case $u \neq v$

A first possibility is to use the identity

$$u^T f(A)v = [(u+v)^T f(A)(u+v) - (u-v)^T f(A)(u-v)]/4$$

Another possibility is to apply the nonsymmetric Lanczos algorithm to the symmetric matrix A
The framework of the algorithm is the same as for the case $u = v$.
However, the algorithm may break down
A way to get around the breakdown problem is to introduce a parameter δ and use $v^1 = u/\delta$ and $\tilde{v}^1 = \delta u + v$. This will give an estimate of $u^T f(A)v/\delta + u^T f(A)u$

The block case

$$I_B[f] = W^T f(A) W = \int_a^b f(\lambda)\, d\alpha(\lambda)$$

For the generation of the matrix orthogonal polynomials we use the block Lanczos algorithm. However, we have seen that we have to start the algorithm from an $n \times 2$ matrix X_0 such that $X_0^T X_0 = I_2$

Considering the bilinear form $u^T f(A) v$ we would like to use $X_0 = [u\, v]$ but this does not fulfill the condition on the starting matrix

We have to orthogonalize the pair $[u\, v]$ before starting the algorithm. Let u and v be independent vectors and $n_u = \|u\|$

$$\tilde{u} = \frac{u}{n_u}, \quad \bar{v} = v - \frac{u^T v}{n_u^2} u, \quad n_v = \|\bar{v}\|, \quad \tilde{v} = \frac{\bar{v}}{n_v},$$

and we set $X_0 = [\tilde{u}\, \tilde{v}]$

Let J^1 be the leading 2×2 submatrix of the matrix $f(J_k)$

$$u^T f(A) v \approx (u^T v) J^1_{1,1} + n_u n_v J^1_{1,2}$$

Moreover

$$u^T f(A) u \approx n_u^2 J^1_{1,1}$$

$$v^T f(A) v \approx n_v^2 J^1_{2,2} + 2(u^T v) \frac{n_u}{n_v} J^1_{1,2} + \frac{(u^T v)^2}{n_u^2} J^1_{1,1}$$

Extensions to nonsymmetric matrices

- nonsymmetric Lanczos algorithm (Saylor and Smolarski)
- Arnoldi algorithm (Calvetti, Kim and Reichel)
- Generalized LSQR (Golub, Stoll and Wathen)
- Vorobyev moment problem (Strakoš and Tichý)

D. Calvetti, Sun–Mi Kim and L. Reichel, *Quadrature rules based on the Arnoldi process*, SIAM J. Matrix Anal. Appl., v 26 n 3, (2005), pp 765–781

G.H. Golub and G. Meurant, *Matrices, moments and quadrature*, in Numerical Analysis 1993, D.F. Griffiths and G.A. Watson eds., Pitman Research Notes in Mathematics, v 303, (1994), pp 105–156

G.H. Golub, M. Stoll and A. Wathen, *Approximation of the scattering amplitude*, submitted to Elec. Trans. Numer. Anal., (2007)

P.E. Saylor and D.C. Smolarski, *Why Gaussian quadrature in the complex plane?*, Numer. Algo., v 26, (2001), pp 251–280

P.E. Saylor and D.C. Smolarski, *Addendum to: Why Gaussian quadrature in the complex plane?*, Numer. Algo., v 27, (2001), pp 215–217

Z. Strakoš, *Model reduction using the Vorobyev moment problem*, to appear in Numer. Algo., (2008)

Z. Strakoš and P. Tichý, *Estimation of $c^* A^{-1} b$ via matching moments*, submitted to SIAM J. Sci. Comput., (2008)