

## ADJUSTING THE RAYLEIGH QUOTIENT IN SEMIORTHOGONAL LANCZOS METHODS\*

G. W. STEWART†

**Abstract.** In a semiorthogonal Lanczos algorithm, the orthogonality of the Lanczos vectors is allowed to deteriorate to roughly the square root of the rounding unit, after which the current vectors are reorthogonalized. A theorem of Simon [*Linear Algebra Appl.*, 61 (1984), pp. 101–132] shows that the Rayleigh quotient—i.e., the tridiagonal matrix produced by the Lanczos recursion—contains fully accurate approximations to the Ritz values in spite of the lack of orthogonality. Unfortunately, the same lack of orthogonality can cause the Ritz vectors to fail to converge. It also makes the classical estimate for the residual norm misleadingly small. In this paper we show how to adjust the Rayleigh quotient to overcome this problem.

**Key words.** large eigenproblem, Lanczos method, partial reorthogonalization, symmetric matrix, adjusted Rayleigh quotient

**AMS subject classifications.** 65F15, 65F25, 65F50

**PII.** S1064827501388984

**1. Introduction and background.** Let  $A$  be a symmetric matrix of order  $n$  and let  $u_1$  be an  $n$ -vector of 2-norm one. The Lanczos algorithm generates a sequence of orthonormal vectors by the recurrence

$$(1.1) \quad \begin{aligned} \beta_1 u_2 &= Au_1 - \alpha_1 u_1, \\ \beta_k u_{k+1} &= Au_k - \alpha_k u_k - \beta_{k-1} u_{k-1}, \quad k = 1, 2, \dots, \end{aligned}$$

where

$$\alpha_k = u_k^T A u_k \quad \text{and} \quad \beta_k = 1 / \|Au_k - \alpha_k u_k - \beta_{k-1} u_{k-1}\|.$$

If we introduce the tridiagonal matrix

$$T_k = \begin{pmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \beta_2 & \alpha_3 & \beta_3 & \\ & & \ddots & \ddots & \ddots \\ & & & \beta_{k-2} & \alpha_{k-1} & \beta_{k-1} \\ & & & & \beta_{k-1} & \alpha_k \end{pmatrix}$$

and set

$$U_k = (u_1 \ \cdots \ u_k),$$

then we can express the effect of the recurrence (1.1) by the *Lanczos decomposition*

$$(1.2) \quad AU_k = U_k T_k + \beta_k u_{k+1} \mathbf{e}_k^T,$$

\*Received by the editors May 4, 2001; accepted for publication (in revised form) November 5, 2001; published electronically May 20, 2002.

<http://www.siam.org/journals/sisc/24-1/38898.html>

†Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 (stewart@cs.umd.edu).

where  $\mathbf{e}_k$  is the  $k$ th unit vector (the  $k$ th column of the identity matrix of order  $k$ ).

It is well known (e.g., see [3]) that as  $k$  increases the space spanned by  $U_k$  contains increasingly accurate approximations to eigenvectors corresponding to the extreme eigenvalues of  $A$ . The approximations can be retrieved by a process known as the Rayleigh–Ritz method. It is based on the observation that if  $(\theta, U_k w)$  is an eigenpair of  $A$ , then  $(\theta, w)$  is an eigenpair of  $T_k$ . A continuity argument suggests that if  $U_k$  contains a good approximation to an eigenvector of  $A$ , there should be an eigenvector  $w$  of  $T_k$  such that  $U_k w$  approximates that eigenvector. For a general analysis of this procedure, see [2]. In what follows we will call  $\theta$  a Ritz value,  $w$  a primitive Ritz vector,  $U_k w$  a Ritz vector, and  $T_k$  the Rayleigh quotient.<sup>1</sup>

A difficulty with the Lanczos algorithm is that the Lanczos vectors tend to lose orthogonality. One cure is to reorthogonalize the vectors at each step—a process known as full reorthogonalization. Unfortunately, reorthogonalizing  $u_{k+1}$  requires the vectors  $u_1, \dots, u_k$ , and for  $k$  large enough the cost of moving them in and out of working storage becomes prohibitive. Consequently, it has been proposed to let orthogonality deteriorate up to a point, after which a reorthogonalization step is performed. There are several varieties of this procedure, all going under the common name of semiorthogonal methods. They each require that the elements of  $U_k^T U_k - I$  be kept less than some multiple of  $\sqrt{\epsilon_M}$ , where  $\epsilon_M$  is the rounding unit for the machine in question. In this paper we will be concerned with periodic and partial reorthogonalization [1, 5].

Full orthogonalization has the advantage that the Lanczos relation (1.2) continues to be satisfied to working accuracy. As we shall see, however, in semiorthogonal methods the Lanczos relation must be replaced by the relation

$$(1.3) \quad AU_k = U_k H_k + \beta_k u_{k+1} \mathbf{e}_k^T,$$

where  $H_k$  is an upper Hessenberg matrix with elements of up to order  $\sqrt{\epsilon_M}$  above the first superdiagonal. We will call (1.3) a Krylov decomposition and  $H_k$  the adjusted Rayleigh quotient. It is shown in [6] that the motivating argument given above for the Rayleigh–Ritz procedure applies equally to this Krylov decomposition. Consequently, if the column space of  $U_k$  contains a good approximation to an eigenvector of  $A$ , we can (under mild restrictions) obtain an approximating eigenpair in the form  $(\theta, U_k w)$ , where  $(\theta, w)$  is a suitable eigenpair of  $H_k$ .

It seems obvious that the use of  $T_k$  in place of  $H_k$  will introduce errors into the Ritz pairs. Surprisingly, this is not true of the Ritz values. According to a remarkable theorem of Simon [4], if  $U_k = QR$  is the QR factorization of  $U_k$ , then  $T_k = Q^T A Q + O(\epsilon_M)$ . Consequently, from the standard perturbation theory for eigenvalues of symmetric matrices, the eigenvalues of  $T_k$  are Ritz values to working accuracy.

On the other hand, if a primitive Ritz vector  $w$  is computed from  $T_k$ , the corresponding Ritz vector is  $Qw$ . However,  $Q$  is unavailable to us, and the attempt to use the approximation  $U_k w$  will introduce errors. These errors cannot be larger than  $O(\sqrt{\epsilon_M})$ , but, as we shall see, they can cause the convergence of a Ritz vector to stagnate before full working accuracy has been attained. For this reason primitive Ritz vectors should be computed from the adjusted Rayleigh quotient  $H_k$ .

A second use for  $H_k$  is to compute accurate residual norms. In the classic Lanczos algorithm if  $(\theta, z = U_k w)$  is a Ritz pair computed from  $T_k$  in (1.2), it is easy to see

<sup>1</sup>Since  $T = U^H A U$  and  $U^T U = I$ ,  $T$  is a natural generalization of the scalar Rayleigh quotient  $u^T A u / u^T u$ .

that in the 2-norm

$$\|Az - \theta z\| = |\beta_k w_k|,$$

where  $w_k$  is the last component of  $w$ , and it is this quantity that is used to decide when a Ritz vector has converged. In a semiorthogonal method, the formula can be used only if  $w$  is computed from the adjusted Rayleigh quotient.

More generally, suppose  $(\theta, U_k w)$  is an approximate eigenpair of  $A$ , with no assumptions made about the origin of  $w$ . Then from (1.3) it follows that

$$AU_k w - \theta U_k w = U_k r + w_k \beta_k u_{k+1},$$

where

$$r = H_k w - \theta w.$$

By semiorthogonality  $U_k r$  and  $w_k \beta_k u_{k+1}$  are almost orthogonal, so that

$$(1.4) \quad \|AU_k w - \theta U_k w\|^2 \cong \|r\|^2 + |w_k \beta_k|^2.$$

Thus we can approximate the residual norm of  $U_k w$  from the norm of the residual  $r$ , the last component of  $w$ , and  $\beta_k$ . In general, it is not possible to replace  $H_k$  by  $T_k$  in computing  $r$ .

**2. Adjusting the Rayleigh quotient.** The foregoing discussion suggests that in a semiorthogonal method the adjusted Rayleigh quotient should be computed. For definiteness we will show how this is done in the context of periodic reorthogonalization, in which an orthogonalization is performed against all preceding vectors whenever the semiorthogonality condition is violated. For convenience we have partitioned this algorithm into the main Lanczos loop (Figure 2.1) and an inner reorthogonalization step (Figure 2.2).

The reorthogonalization corresponds to statements 12–14 in the main loop.

```

1.  for  $k = 1, 2, \dots$ 
2.       $v = A * u_k$ 
3.      if ( $k \neq 1$ )
4.           $v = v - \beta_{k-1} * u_{k-1}$ 
5.           $H[k-1, k] = H[k, k-1] = \beta_{k-1}$ 
6.      end if
7.       $\alpha_k = u_k^T * v$ 
8.       $H[k, k] = \alpha_k$ 
9.       $v = v - \alpha_k * u_k$ 
10.     if ( $k \neq 1$ )  $v = v - (u_{k-1}^T v) * u_{k-1}$  fi
11.      $v = v - (u_k^T v) * u_k$ 
12.     if (semiorthogonality is violated)
13.         Reorthogonalize  $u_k$  and  $v$ . Update  $H$ .
14.     end if
15.      $\beta_k = \|v\|_2$ 
16.      $u_{k+1} = v / \beta_k$ 
17. end for  $i$ 
```

FIG. 2.1. *Lanczos with periodic reorthogonalization: Main loop.*

```

1.  if (semiorthogonality is violated)
2.       $\rho = \|v\|_2$ 
      Reorthogonalize  $u_k$  and  $v$ 
3.  for  $j = 1$  to  $k-1$ 
4.       $w[j] = u_j^T * u_k$ 
5.       $u_k = u_k - w[j] * u_j$ 
6.       $x[j] = u_j^T * v$ 
7.       $v = v - x[j] * u_j$ 
8.  end for  $j$ 
9.   $\eta = u_k^T * v$ 
10.  $v = v - \eta * u_k$ 
      If necessary orthogonalize  $v$  again
11. if  $\|x\|_2 \geq \sqrt{n\epsilon_M} * \rho$ 
12.     for  $j = 1$  to  $k-1$ 
13.          $t = u_j^T * v$ 
14.          $v = v - t * u_j$ 
15.          $x[j] = x[j] + t$ 
16.     end for  $j$ 
17.      $t = u_k^T * v$ 
18.      $v = v - t * u_k$ 
19.      $\eta = \eta + t$ 
20. end if
      Adjust  $H$ 
21.      $H[1:k-1, k-1] = H[1:k-1, k-1] + H[k-1, k] * w$ 
22.      $H[1:k-1, k] = H[1:k-1, k] - H[1:k-1, 1:k-1] * w$ 
         $-(\beta[k-1] * w[k-1] - H[k, k] * w + x$ 
23.      $H[k, k] = H[k, k] - \beta_{k-1} * w[k-1] + \eta$ 
24. end if

```

FIG. 2.2. *Lanczos with periodic reorthogonalization: Reorthogonalization.*

The main loop is typical of most implementations of the Lanczos method. The vector  $v = Au$  is computed via the Lanczos recurrence. Note that the algorithm fills in the tridiagonal part of  $H_k$ , so that if there were no reorthogonalizations  $H_k$  would be identical to  $T_k$ . Statements 10–11 are a local reorthogonalization step that insures that  $v$  is orthogonal to  $u_k$  and  $u_{k-1}$  to working accuracy.

The reorthogonalization section (Figure 2.2) is entered only if the off-diagonal elements of  $U_k^T U_k$  are too large. These elements cannot be computed directly, except by bringing  $U_k$  into working storage at each step. Fortunately, there are recurrences that can be used to estimate the components. For details see [5].

It is necessary to reorthogonalize both  $u_k$  and  $v$ . The reason is that  $u_{k+2}$  will depend on both these vectors, and if one fails to be fully orthogonal, its lack of orthogonality will be propagated into  $u_{k+2}$  and its successors. The reorthogonalization is done by the modified Gram–Schmidt in statements 3–10. The reorthogonalization coefficients for  $u_k$  are stored in  $w$ ; those for  $v$  are stored in  $x$  and  $\eta$ .

It may be necessary to re-reorthogonalize  $v$  against the columns of  $U_k$ . The reason is not the usual one: namely, that cancellation in the reorthogonalization can magnify nonorthogonal components. Rather the fact that  $U_k$  is only semiorthogonal

compromises its ability to purge components in the orthogonal complement of its column space. This re-orthogonalization is done in statements 11–19.

The Rayleigh quotient is adjusted in statements 21–23. To derive the formulas, note that before the reorthogonalization we have the relation

$$(2.1) \quad AU_k = U_k H_k + v \mathbf{e}_k^T,$$

which holds to working accuracy. The results of the reorthogonalization replace  $u_k$  with

$$(2.2) \quad \tilde{u}_k = u_k - U_{k-1} w$$

and  $v$  with

$$(2.3) \quad \tilde{v} = v - U_{k-1} x - \eta \tilde{u}_k.$$

Solving (2.2) and (2.3) for  $u_k$  and  $v$  and substituting the results into (2.1), we get

$$(2.4) \quad A(U_{k-1} \tilde{u}_k) + A(0 \ U_{k-1} w) = (U_{k-1} \tilde{u}_k) H_k + (0 \ U_{k-1} w) H_k + \tilde{v} \mathbf{e}_k^T + (U_{k-1} x + \eta \tilde{u}_k) \mathbf{e}_k^T.$$

Now since  $AU_{k-1} = UH_{k-1} + \beta_{k-1}u_k\mathbf{e}_k^T$ , we have from (2.2)

$$(2.5) \quad A(0 \ U_{k-1} w) = U_{k-1}(H_{k-1} + \beta_{k-1}w\mathbf{e}_{k-1}^T)w + \beta_{k-1}\tilde{u}_k\mathbf{e}_{k-1}^T w.$$

Also

$$(2.6) \quad (0 \ U_{k-1} w) H_k = U_{k-1} w \mathbf{e}_{k-1}^T H_k = h_{k-1,k} U_{k-1} w \mathbf{e}_{k-1}^T + h_{kk} U_{k-1} w \mathbf{e}_k^T.$$

If we substitute (2.5) and (2.6) into (2.4) and define  $\tilde{H}_k$  by

$$(2.7) \quad \begin{aligned} 1. \quad & \tilde{H}[1:k-1, k-1] = H[1:k-1, k-1] + h_{k-1,k} w, \\ 2. \quad & \tilde{H}[1:k-1, k] = H[1:k-1, k] - H_{k-1} w - \beta_{k-1} w_{k-1} w + h_{kk} w + x, \\ 3. \quad & \tilde{H}[k, k] = H[k, k] - \beta_{k-1} w[k-1] + \eta, \end{aligned}$$

then it is straightforward (but tedious) to verify that

$$A(U_{k-1} \tilde{u}_k) = (U_{k-1} \tilde{u}_k) \tilde{H}_k + \tilde{v} \mathbf{e}_k^T,$$

which is a Krylov decomposition involving the new vectors  $\tilde{u}_k$  and  $\tilde{v}$ . These are the corrections in statements 21–23.

It is worth noting that the complexity of these corrections is due to the fact that we must orthogonalize both  $u_k$  and  $v$ . If we omit the orthogonalization of  $u_k$ , then  $w = 0$ , and the corrections reduce to adding the orthogonalization coefficients for  $v$  to the  $k$ th column of  $H_k$ .

**3. An example.** The following example illustrates how periodic reorthogonalization behaves. Our matrix  $A$  is diagonal of order  $n = 500$  whose first eigenvalue is  $\lambda_1 = 1$  and whose other eigenvalues are given by the recurrence

$$\lambda_i = \frac{\lambda_{i-1}}{1 + 1/i^2}, \quad i = 2, \dots, n.$$

The periodically reorthogonalized Lanczos algorithm was run on this matrix with a random starting vector and a cutoff for reorthogonalization of  $\sqrt{10^{-16}/n}$ . Orthogonalizations took place for  $k = 11, 17, 22, 28, 33, 38$ . Various statistics are summarized in the following table, in which we are concerned with the approximate eigenpairs  $(\mu_5^{(k)}, U_k w_5^{(k)})$ , where  $(\mu_5^{(k)}, w_5^{(k)})$  is the fifth eigenpair of the Lanczos matrix  $T_k$ .

$k$	c1	c2	c3	c4	c5
10	3.9e-15	3.9e-15	1.9e-02	1.9e-02	1.9e-02
20	5.3e-15	3.9e-15	2.5e-07	2.5e-07	2.5e-07
30	6.5e-15	3.9e-15	2.8e-16	1.4e-12	1.4e-12
40	7.0e-15	3.9e-15	8.9e-19	1.4e-12	1.4e-12

The columns of the table contain the following quantities.

- c1.  $\|T_k - Q^T A Q\|_2$ , where  $Q$  is the  $Q$  factor of  $U_k$ .
- c2. The global residual norm

$$\|AU_k - U_k H_k - \beta_k u_{k+1} \mathbf{e}_k^T\|_k.$$

- c3. The classical estimate  $|\beta_k \mathbf{e}_k^T w_5^{(k)}|$  for the residual norm.
- c4. The true residual norm.
- c5. The residual norm estimated by (1.4).

The numbers in the first column illustrate the result of Simon mentioned above. The tridiagonal Lanczos matrix  $T_k$  is to working accuracy the Rayleigh quotient formed from the  $Q$  factor of  $U_k$ . This means that in spite of the contamination of  $T_k$  by the reorthogonalization process, its eigenvalues converge to eigenvalues of  $A$ .

The second column shows that the corrections in the algorithm depicted in Figure 2.2 produce a Krylov decomposition whose residual is of the order of the rounding unit. Although to the two figures shown this residual is not changing, actually it is increasing very slightly with  $k$ .

The third and fourth columns show that the classic estimate for the residual norm deviates from the true residual norm. In particular the latter stagnates at about  $10^{-12}$ , whereas the former continues to decrease. But the two values diverge only after the true residual norm falls below  $\sqrt{\epsilon_M}$ . The fifth column shows that the approximate residual norm (1.4) is a reliable estimate of the true residual norm.

**4. Discussion.** The algorithm in Figure 2.2 shows that the running computation of the adjusted Rayleigh quotient is computationally inexpensive. One can also economize on storage, since except for spikes at the reorthogonalization points the elements above the first superdiagonal are zero. These spikes can be squirreled away in a special data structure, so that one does not have to store a full matrix of order  $k$ . However, such economies may not be worth the trouble, since an array of order  $k$  is needed to store primitive Ritz vectors.

Although  $H_k$  must be updated at each orthogonalization step, it does not have to be used until the (suitably scaled) residual norms for the approximate eigenvectors approach  $\sqrt{\epsilon_M}$ . Thereafter it should be used in residual computations. If a residual stagnates at an unacceptably high value, then one must use primitive Ritz vectors computed from  $H_k$ . Since the eigenpairs of  $T_k$  are approximations to the eigenpairs of  $H_k$ , one can compute the former and use the inverse power method to adjust them—an inexpensive process given the structure of  $H_k$ .

**Acknowledgment.** I am grateful for the use of the facilities of the Mathematical and Computational Sciences Division of NIST, where part of this work was done.

## REFERENCES

- [1] J. GRGAR, *Analyses of the Lanczos Algorithm and of the Approximation Problem in Richardson's Method*, Ph.D. thesis, University of Illinois at Urbana-Champaign, 1981.
- [2] Z. JIA AND G. W. STEWART, *An analysis of the Rayleigh-Ritz method for approximating eigenspaces*, Math. Comp., 70 (2001), pp. 637–647.
- [3] Y. SAAD, *Numerical Methods for Large Eigenvalue Problems: Theory and Algorithms*, John Wiley, New York, 1992.
- [4] H. D. SIMON, *Analysis of the symmetric Lanczos algorithm with reorthogonalization methods*, Linear Algebra Appl., 61 (1984), pp. 101–132.
- [5] H. D. SIMON, *The Lanczos algorithm with partial reorthogonalization*, Math. Comp., 42 (1984), pp. 115–142.
- [6] G. W. STEWART, *A Krylov-Schur algorithm for large eigenproblems*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 601–614.