SIAM J. MATRIX ANAL. APPL.
Vol. 16, No. 4, pp. 1154–1171, October 1995

009

A RESTARTED GMRES METHOD AUGMENTED WITH EIGENVECTORS *

RONALD B. MORGAN†

Abstract. The GMRES method for solving nonsymmetric linear equations is generally used with restarting to reduce storage and orthogonalization costs. Restarting slows down the convergence. However, it is possible to save some important information at the time of the restart. It is proposed that approximate eigenvectors corresponding to a few of the smallest eigenvalues be formed and added to the subspace for GMRES. The convergence can be much faster, and the minimum residual property is retained.

Key words. GMRES, conjugate gradient, Krylov subspaces, iterative methods, nonsymmetric systems

AMS subject classifications. 65F15, 15A18

1. Introduction.

The GMRES method [32] is popular for solving the large nonsymmetric system of linear equations

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

But GMRES is generally used with restarting, and this slows down the convergence. We examine a way to retain some of the information lost at the time of the restart. The convergence can be improved in many situations. This section gives background material on GMRES. Section 2 gives the new method and analyzes its effectiveness for certain cases. Section 3 discusses the implementation and the expenses. Examples and comparisons are given in §4, and §5 looks at the possibility of having a procedure that selects the number of approximate eigenvectors and decides how long they should be used.

For symmetric problems, the conjugate gradient method [13], [17] is often the best iterative method. It extracts an approximate solution from the Krylov subspace $\text{Span}\{b, Ab, A^2b, \ldots, A^{m-1}b\}$. There is an efficient recurrence formula for generating a sequence of orthogonal vectors that span the Krylov subspace. Also the convergence properties are fairly well understood for a Krylov subspace. They depend on the eigenvalue distribution. A simple bound for the minimum residual version [15], [17], [28] of the conjugate gradient method applied to a symmetric positive definite matrix is

$$\begin{aligned} \frac{||r||}{||b||} &\leq 2\Big(\Big(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\Big)^m + \Big(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\Big)^m\Big)^{-1} \\ &\leq 2\Big(1 - \frac{2}{\sqrt{\kappa}+1}\Big)^m, \end{aligned} \tag{2}$$

where r is the residual vector $b - A\hat{x}$, and $\hat{x}$ is the approximate solution. Also $\kappa \equiv \frac{\lambda_n}{\lambda_1}$ is the condition number, the ratio of largest to smallest eigenvalues. So convergence

*Received by the editors August 18, 1993; accepted for publication (in revised form) by A. Greenbaum December 3, 1994. This research was supported by National Science Foundation contract CCR-8910665.

†Department of Mathematics, Baylor University, P.O. Box 97328, Waco, Texas 76798-7328 (morgan@baylor.edu).

is generally slow if there is an extremely small eigenvalue. But the placement of the other eigenvalues also influences convergence. Clumping of eigenvalues is favorable. The actual convergence rate often improves as the method proceeds [4], [5], [7], [35]. This is because some of the outlying eigenvalues are effectively eliminated from the spectrum once the Krylov subspace contains a good approximation to the corresponding eigenvector. Another good thing about the conjugate gradient method is that the convergence can usually be improved by preconditioning (multiplying (1) by an approximate inverse to A) [3], [6], [13], [15], [23].

The conjugate gradient method can be generalized to nonsymmetric problems in several ways. The three main approaches are the nonsymmetric Lanczos algorithm [19], [20], [37], the conjugate gradient method applied to the normal equations (CGNE) [8], [16], and GMRES [32]. The nonsymmetric Lanczos method is similar to the conjugate gradient method in that it uses a Krylov subspace and has a recurrence formula. The algorithm is unstable, but improvements have been made [11], [12], [14], [18], [29], [36]. In particular, the QMR version [11], [12] has attracted attention. The CGNE method transforms to another problem (the normal equations), so the convergence properties are different. Often convergence is much slower. Nevertheless there are some problems, particularly indefinite and fairly nonsymmetric ones, for which CGNE is best [26]. GMRES is currently a popular method for large nonsymmetric problems (see, for example, [21], [27]). It uses the Arnoldi algorithm [1], [30], [31], [37] to build an orthonormal basis for the Krylov subspace, so full orthogonalization is needed. The best approximate solution is extracted from the subspace, in that the norm of the residual vector is minimized.

Because full orthogonalization is used, the method becomes more expensive as the subspace grows. Also importantly, the storage requirements increase. Restarting can be used when the subspace reaches a certain size.

RESTARTED GMRES

1. *Start:* Choose x_0 and compute $r_0 = b - Ax_0$ and $v_1 = r_0/||r_0||$.
2. *Iterate:* For $j = 1, 2, \ldots, m$ do:
   $h_{ij} = (Av_j, v_i), i = 1, 2, \ldots, j,$
   $\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} h_{ij} v_i,$
   $h_{j+1,j} = ||\hat{v}_{j+1}||$, and
   $v_{j+1} = \hat{v}_{j+1}/h_{j+1,j}$.
3. *Form the approximate solution:* $\hat{x} = x_0 + V\hat{d}$, where $\hat{d}$ minimizes $||\beta e_1 - \bar{H}d||$, for all $d \in R^m$. Here $\bar{H}$ is the $(m+1)$ by m matrix with elements h_{ij} defined in step 2, and $\beta = ||r_0||$.
4. *Restart:* Compute $r = b - A\hat{x}$; if satisfied then stop, else let $x_0 = \hat{x}$, $v_1 = r/||r||$, $r_0 = r$, and go to 2.

The convergence of GMRES is similar to that for the conjugate gradient method if the matrix is nearly normal. Again the presence of small eigenvalues slows convergence. Suppose A has spectral decomposition $A \equiv Z\Lambda Z^{-1}$, with all the eigenvalues being real and positive. Assuming that the initial guess x_0 is the zero vector, we have

$$\frac{||r||}{||b||} \leq 2||Z||||Z^{-1}||\Big(1 - \frac{2}{\sqrt{\kappa}+1}\Big)^m, \tag{3}$$

(see [32] for similar but more general results). Again $\kappa = \frac{\lambda_n}{\lambda_1}$, but here it is not necessarily the same as the standard condition number. For more highly nonnormal matrices, convergence properties are more complicated. Some analysis has been done,

especially if all of the eigenvalues are in an ellipse not containing the origin (see [9], [22], [30–32]).

The disadvantage with restarting is that some information is lost at the time of the restart. The subspace is discarded, and this slows down the convergence. Other methods such as nonsymmetric Lanczos and CGNE avoid restarting. But they have their own disadvantages as mentioned earlier. Another attempt at avoiding restarting is incomplete orthogonalization [30], [31], but convergence properties are not well understood.

2. Adding approximate eigenvectors to the subspace. We attempt to improve GMRES by reducing the ill effects of restarting. Some information can be retained at the time of the restart. This is done by saving vectors from the old subspace and adding them to the new subspace that is generated. For instance, one could save the last few Arnoldi vectors (the v_i's). However, there are other vectors that are more helpful to the convergence.

We note that information about the eigenvalues and eigenvectors of A is available during GMRES. They can be calculated with the Arnoldi method for eigenvalues [1], [30], [37]. Eigenvalue calculations have been used before in conjunction with GMRES to implement hybrid methods (see [10], for example).

We investigate saving approximate eigenvectors of A corresponding to the smallest eigenvalues in magnitude. These vectors are added to the new subspace. The motivation for this is that if a converged eigenvector is added to the subspace, the corresponding eigenvalue is effectively eliminated from the spectrum or deflated. Convergence proceeds according to the modified spectrum. This is demonstrated in the following theorem for the case of real and positive eigenvalues. We let $\kappa_e \equiv \frac{\lambda_n}{\lambda_{k+1}}$, the "effective condition number," and assume that the initial guess x_0 is zero.

THEOREM 1. *Suppose A has spectral decomposition $A \equiv Z\Lambda Z^{-1}$, with all the eigenvalues being real and positive. Assume that the minimum residual solution $\hat{x}$ is extracted from the subspace* $\mathrm{Span}\{b, Ab, \ldots, A^{m-1}b, z_1, z_2, \ldots, z_k\}$, *where the z_i's are columns of Z. Then*

$$\frac{||r||}{||b||} \leq 2||Z||||Z^{-1}||\Big(1 - \frac{2}{\sqrt{\kappa_e}+1}\Big)^m, \tag{4}$$

where $r \equiv b - A\hat{x}$ is the residual vector.

Proof. Any vector $\hat{x}$ from the subspace $\mathrm{Span}\{b, Ab, \ldots, A^{m-1}b, z_1, z_2, \ldots, z_k\}$ can be written in the form

$$\hat{x} = \sum_{i=1}^{k} \alpha_i z_i + p(A)b,$$

where p is a polynomial of degree $m-1$ or less. Expand b in terms of the eigenvectors:

$$b = \sum_{i=1}^{n} \beta_i z_i \tag{5}$$

and define the polynomial q as $q(x) = 1 - xp(x)$. Then we can calculate that

$$\begin{aligned} r = b - A\hat{x} &= -\sum_{i=1}^{k} \alpha_i \lambda_i z_i + q(A)b \\ &= \sum_{i=1}^{k} \gamma_i z_i + \sum_{i=k+1}^{n} \beta_i q(\lambda_i) z_i, \end{aligned} \tag{6}$$

where $\gamma_i = \beta_i q(\lambda_i) - \alpha_i \lambda_i$.

Since the solution minimizes the residual norm, it will be at least as good as any choice we make. Pick q to be the shifted-and-scaled Chebyshev polynomial that is small over the interval $[\lambda_{k+1}, \lambda_n]$. Then pick $\alpha_i = \frac{\beta_i q(\lambda_i)}{\lambda_i}$, so that each γ_i is zero. Now

$$r = \sum_{i=k+1}^{n} \beta_i q(\lambda_i) z_i,$$

and the desired result follows from the standard bound in (3).

Next, the effect of saving eigenvectors is examined for a couple of specific distributions of eigenvalues. First suppose the eigenvalues are distributed $1, 2, 3, \ldots, n$. Some linear equations problems do have a spectrum roughly similar to this (for example, see the model problem [15] from finite difference descretization of Poisson's equation). Suppose the eigenvectors corresponding to the k smallest eigenvalues are added to the subspace. Then the convergence bound improves from

$$\frac{||r||}{||b||} \leq 2||Z||||Z^{-1}||\Big(1 - \frac{2}{\sqrt{n}+1}\Big)^m$$

to

$$\frac{||r||}{||b||} \leq 2||Z||||Z^{-1}||\Big(1 - \frac{2\sqrt{k+1}}{\sqrt{n}+\sqrt{k+1}}\Big)^m.$$

We can roughly compare convergence by comparing $\sqrt{\kappa}$ to $\sqrt{\kappa_e}$. The ratio is

$$\frac{\sqrt{\kappa}}{\sqrt{\kappa_e}} = \frac{\sqrt{n}}{\sqrt{\frac{n}{k+1}}} = \sqrt{k+1}. \tag{7}$$

So convergence is roughly $\sqrt{k+1}$ times as fast with the eigenvectors added to the subspace. For example, with $k = 3$, the rate of convergence is about twice as fast. However, to get quadruple the convergence requires $k = 15$. The returns are diminishing as more eigenvectors are added.

Next, consider the eigenvalue distribution $\frac{-n}{2}, \frac{-n}{2}+1, \ldots, -2, -1, 1, 2, \ldots, \frac{n}{2}$. This is a much tougher problem than the previous one, because it is indefinite. The residual norm is reduced by roughly a factor of $1 - \frac{2}{n}$ in each iteration [2], [24]. If k is even and the k eigenvectors with smallest eigenvalues in magnitude are added to the subspace, then the factor improves to $1 - \frac{k+2}{n}$. This means convergence is approximately $\frac{k+2}{2}$ times better with the eigenvectors. Adding eigenvectors to the subspace is even more important in this indefinite case than it was in the previous positive definite example. With $k = 8$, convergence is about five times better. This compares to three times better in the previous example.

The results in the preceding two paragraphs may not always apply. We will discuss three problems with them. First, the distribution of eigenvalues may not be so favorable. For example, there may not be small eigenvalues. Then convergence could still be slow for indefinite and highly nonsymmetric problems, yet saving approximate eigenvectors would not be beneficial. Also, if the eigenvalues are not so evenly spaced, the results may not be as good.

Second, we have only analyzed bounds on convergence and estimates of rates, not the actual rates of convergence. As mentioned earlier, the convergence of the conjugate gradient method does not depend strictly on the condition number. An outlying

eigenvalue does not have a perpetual effect on the covergence rate. It will at most add a number of iterations, then the outlying eigenvalue is taken care of and the convergence proceeds according to the other eigenvalues. This is because the underlying polynomial can have a zero at that eigenvalue. However, Cline [5] observed in some experiments that adding an extremely small eigenvalue to a particular distribution of eigenvalues requires from 5 to 19 extra iterations. With restarted GMRES, reducing the number of iterations by a few is worthwhile, because this reduction occurs during every restart cycle.

A third problem with the earlier analysis is that it may be awhile before the approximate eigenvectors become very accurate. However, an approximate eigenvector can have beneficial effects long before it has attained full accuracy. This is shown in the following theorem for the case of one approximate eigenvector.

THEOREM 2. *Suppose A has spectral decomposition $A \equiv Z\Lambda Z^{-1}$, with Λ diagonal. Suppose the* GMRES *with eigenvectors method is used with one approximate eigenvector y_1. Let $\psi \equiv \angle(y_1, z_1)$, and let β_1 be the coefficient of z_1 in the expansion of b; see* (5). *Then*

$$||r|| \leq ||Z||||Z^{-1}|| \max_{i \neq 1} |q(\lambda_i)| \, ||b|| + \frac{||A||}{\lambda_1} \tan\psi \, |q(\lambda_1)| \, |\beta_1|, \tag{8}$$

where q is a polynomial of degree m or less such that $q(0) = 1$.

Proof. Similar to (6), we can derive

$$r = q(A)b - \alpha_1 A y_1,$$

where q is a polynomial of degree m or less, such that $q(0) = 1$. Decompose y_1 as

$$y_1 = \cos\psi \, z_1 + \sin\psi \, u,$$

where y_1, z_1, and u are all unit vectors and $u \perp z_1$. Then

$$\begin{aligned} r &= q(A)b - \alpha_1 \lambda_1 \cos\psi z_1 - \alpha_1 \sin\psi Au \\ &= \sum_{i=2}^{n} \beta_i q(\lambda_i) z_i + \big(\beta_1 q(\lambda_1) - \alpha_1 \lambda_1 \cos\psi\big) z_1 - \alpha_1 \sin\psi Au. \end{aligned}$$

Pick $\alpha_1 = \frac{\beta_1 q(\lambda_1)}{\lambda_1 \cos\psi}$, and use the minimum residual property. Then

$$\begin{aligned} ||r|| &\leq \left\| \sum_{i=2}^{n} \beta_i q(\lambda_i) z_i - \frac{\beta_1 q(\lambda_1) \sin\psi Au}{\lambda_1 \cos\psi} \right\| \\ &\leq \left\| \sum_{i=2}^{n} \beta_i q(\lambda_i) z_i \right\| + \left\| \frac{\beta_1 q(\lambda_1) \tan\psi Au}{\lambda_1} \right\| \\ &\leq ||Z||||Z^{-1}|| \max_{i \neq 1} |q(\lambda_i)| \, ||b|| + \frac{||A||}{\lambda_1} \tan\psi \, |q(\lambda_1)| \, |\beta_1|. \end{aligned}$$

The second term in the right-hand side of (8) occurs because of the inaccuracy of the approximate eigenvector. Roughly, it appears that this term will not be significant as long as the accuracy of the approximate eigenvector is greater than the amount of

improvement brought by the polynomial q (as long as $\tan\psi$ is somewhat less than $\max_{i\neq 1}|q(\lambda_i)|$).

If the eigenvalues are all real and positive, (8) can be made more specific by choosing the polynomial q to be a shifted and scaled Chebyshev polynomial that is small over the interval $[\lambda_2, \lambda_n]$. Then

$$||r|| \leq ||Z||||Z^{-1}||\left(1 - \frac{2}{\sqrt{\kappa_e}+1}\right)^m ||b|| + \frac{||A||}{\lambda_1}\tan\psi\,|\beta_1|, \tag{9}$$

where $\kappa_e \equiv \frac{\lambda_n}{\lambda_2}$. And this can be put in a form more similar to (4):

$$\frac{||r||}{||b||} \leq ||Z||||Z^{-1}||\left(\left(1 - \frac{2}{\sqrt{\kappa_e}+1}\right)^m + \frac{||A||}{\lambda_1}\tan\psi\right). \tag{10}$$

3. Implementation. The implementation presented here first generates the Krylov subspace, then adds the approximate eigenvectors. There is still an upper-Hessenberg matrix for the linear equations problem, but the eigenvalue problem is more complicated.

Let m be the dimension of the Krylov subspace, and suppose k approximate eigenvectors are used. Let $l = m + k$. Let W be the n by l matrix whose first m columns are the orthonormalized Arnoldi vectors (the v_i vectors in step 2 of GMRES) and whose last k vectors are the approximate eigenvectors y_i, for $i = 1, \ldots, k$. Let Q be the n by $l+1$ matrix whose first $m+1$ columns are Arnoldi vectors and whose last k columns are formed by orthogonalizing the vectors Ay_i, for $i = 1, \ldots, k$, against the previous columns of Q. Then

$$AW = Q\bar{H}, \tag{11}$$

where $\bar{H}$ is an $(l+1)$ by l upper-Hessenberg matrix (this is similar to (3) in [32], for the standard Arnoldi iteration on which GMRES is based).

The restarted linear equations problem is

$$A(x - x_0) = r_0.$$

The approximate solution $\hat{x} - x_0$ is a combination of the columns of W, so

$$\hat{x} - x_0 = W\hat{d}.$$

The minimum residual solution can be calculated in the same way as for standard GMRES. Let

$$P\bar{H} = R, \tag{12}$$

where P is orthogonal and R is upper triangular. Then

$$\begin{aligned} ||r|| &= ||b - A\hat{x}|| \\ &= ||r_0 - A(\hat{x} - x_0)|| \\ &= ||r_0 - AW\hat{d}|| \\ &= ||r_0 - Q\bar{H}\hat{d}|| \\ &= ||Q^*r_0 - \bar{H}\hat{d}|| \\ &= ||PQ^*r_0 - R\hat{d}||. \end{aligned} \tag{13}$$

The minimal solution is then found by solving for $\hat{d}$ that makes the first l entries be zero. Note Q^*r_0 is a multiple of the first coordinate vector. As in standard GMRES, the residual norm is a byproduct. It is the magnitude of the last entry of PQ^*r_0.

We wish to find approximate eigenvectors from the subspace spanned by the columns of W. Since W is not orthonormal, the generalized Rayleigh–Ritz procedure with reduced eigenvalue problem

$$W^*AWg_i = \theta_i W^*Wg_i$$

could be used. However, we choose a version of Rayleigh-Ritz that finds good approximations to the eigenvalues nearest to zero [24], [25]. This version uses the reduced problem

$$W^*A^*Wg_i = \frac{1}{\theta_i} W^*A^*AWg_i. \tag{14}$$

Let $F \equiv W^*A^*W$ and $G \equiv W^*A^*AW$. Then the reduced eigenvalue problem is the l by l generalized eigenvalue problem

$$Fg_i = \frac{1}{\theta_i} Gg_i. \tag{15}$$

The g_i's associated with the k largest $\frac{1}{\theta_i}$'s (or the k smallest θ_i's) are needed. An approximate eigenvector is $y_i = Wg_i$. And $Ay_i = AWg_i = Q\bar{H}g_i$. If y_i is complex, the real and imaginary parts are used separately.

Little calculation is required for G, because

$$\begin{aligned} G &= W^*A^*AW \\ &= \bar{H}^*Q^*Q\bar{H} \\ &= \bar{H}^*\bar{H} \\ &= R^*R. \end{aligned} \tag{16}$$

The first m columns of F are the same as the first m columns of $\bar{H}^*$. Entries in the intersection of the last k rows and the last k columns can be cheaply computed using the previous F, since $f_{ij} = y_i^*A^*y_j = g_i^*W_{\text{old}}^*A^*W_{\text{old}}g_j = g_i^*F_{\text{old}}g_j$. The remaining entries are calculated as $f_{ij} = y_i^*A^*y_j = (Ay_i)^*y_j$, so they are more expensive.

The small generalized eigenvalue problem (15) is solved with EISPACK [33] in the examples in the next section. However an iterative method, such as subspace iteration, could also be used. Only the eigenvectors associated with the largest values of $\frac{1}{\theta_i}$ are needed, G is already in a factored form, good starting vectors are the last k coordinate vectors, and full convergence is not necessary.

The implementation is a little different for the first run, before any restart. Standard GMRES is used, except eigenvector calculations are added on at the end. F is the same as $\bar{H}^*$ except that the last column is removed, and G can be found with (16). For simplicity, the listing of the algorithm is given just for the second and subsequent runs.

One restarted run of GMRES with eigenvectors

1. *Initial definitions and calculations:* The Krylov subspace has dimension m, k is the number of approximate eigenvectors, and $l = m + k$. Let $q_1 = r_0/||r_0||$ and

$w_1 = q_1$. Let $y_1, y_2, \ldots, y_k$ be the approximate eigenvectors. Let $w_{m+i} = y_i$, for $i = 1, \ldots, k$. For $j = m+1, \ldots, l$ do: $f_{ij} = g_i^* F_{\text{old}} g_j, i = m+1, \ldots, l$.

2. *Generation of Arnoldi vectors:* For $j = 1, 2, \ldots, m$ do:
   $h_{ij} = (Aq_j, q_i), i = 1, 2, \ldots, j$,
   $f_{ji} = h_{ij}, i = 1, 2, \ldots, j$,
   $f_{j,m+i} = (Aq_j, y_i), i = 1, 2, \ldots, k$,
   $\hat{q}_{j+1} = Aq_j - \sum_{i=1}^{j} h_{ij} q_i$,
   $h_{j+1,j} = ||\hat{q}_{j+1}||$, and
   $q_{j+1} = \hat{q}_{j+1}/h_{j+1,j}$.
   If $j < m$, let $w_{j+1} = q_{j+1}$ and $f_{j,j+1} = h_{j+1,j}$.
3. *Addition of approximate eigenvectors:* For $j = m+1, \ldots, l$ do:
   $h_{ij} = (Aw_j, q_i), i = 1, 2, \ldots, j$,
   $f_{ji} = h_{ij}, i = 1, 2, \ldots, m$,
   $\hat{q}_{j+1} = Aw_j - \sum_{i=1}^{j} h_{ij} q_i$,
   $h_{j+1,j} = ||\hat{q}_{j+1}||$, and
   $q_{j+1} = \hat{q}_{j+1}/h_{j+1,j}$.
4. *Form the approximate solution:* Let $\beta = ||r_0||$. Find $\hat{d}$ that minimizes $||\beta e_1 - \bar{H} d||$ for all $d \in R^l$. The orthogonal factorization $P\bar{H} = R$, for R upper triangular, is used. Then $\hat{x} = x_0 + W\hat{d}$.
5. *Form the new approximate eigenvectors:* Calculate $G = R^* R$. Solve $Fg_i = \frac{1}{\theta_i} Gg_i$, for the appropriate g_i (separate g_i into real and complex parts if it is complex and treat as two distinct vectors). Form $y_i = Qg_i$ and $Ay_i = Q\bar{H}g_i$. Let $F_{\text{old}} = F$.
6. *Restart:* Compute $r = b - A\hat{x}$; if satisfied with the residual norm then stop, else let $x_0 = \hat{x}$ and go to 2.

We now examine the expenses and storage requirements for the GMRES with eigenvectors method as compared to standard GMRES. We consider only the major expenses. Suppose the subspace is currently a Krylov subspace of dimension j. If we choose the next vector for the subspace to be one more Arnoldi vector, then there is one matrix-vector product needed. The orthogonalization requires about $2jn$ multiplications. If instead we let the next vector be an approximate eigenvector, no matrix-vector product is required. The other costs are approximately $5jn$ multiplications. This includes $2jn$ for the orthogonalization of Ay_i, jn for computing a portion of F, and $2jn$ for forming y_i and Ay_i. This can be reduced to $4jn$ if a matrix-vector product is used for Ay_i instead of forming it from the columns of Q. It is also possible to reduce costs by another jn if Ay_i is not explicitly orthogonalized (the entries of $\bar{H}$ can still be calculated). This last option has not been tested.

We compare the storage for a Krylov subspace of dimension $m + k$ in standard GMRES to storage for the GMRES with eigenvectors method using a Krylov subspace of dimension m and k approximate eigenvectors. The major storage requirement for GMRES($m + k$) is $m + k + 2$ vectors of length n. For GMRES with eigenvectors, the major storage requirement is for $m+2k+2$ vectors of length n. So using an approximate eigenvector requires about twice the storage of using an additional Arnoldi vector. This is because both y_i and Ay_i are stored.

The relative efficiency of the two methods depends on how expensive the matrix-vector product is compared to the orthogonalization costs. We consider two extreme cases, although many problems will fall somewhere in between. The first case is where the matrix-vector product is the main expense, and the second is where the matrix-vector product is fairly inexpensive and orthogonalization costs dominate. Saving approximate eigenvectors is particularly worthwhile for the first case, since no matrix-vector product is required for the approximate eigenvectors. The benefits of the

approximate eigenvectors are essentially free from expense. However, since storage is often limited, using one approximate eigenvector means two less Arnoldi vectors can be used.

For the second case of expensive orthogonalization, a matrix-vector product would be used to get Ay_i. So the expense is about $4jn$ for an approximate eigenvector. Therefore using an approximate eigenvector instead of an Arnoldi vector costs about twice as much. To be useful, an approximate eigenvector must be as effective as two Arnoldi vectors.

4. Examples. In the following examples, the right-hand sides have all entries 1.0. The first four examples are bidiagonal matrices with 0.1 in each superdiagonal position. The initial guesses x_0 are zero vectors. The calculations are done in double precision on either an IBM 3090-170J or a Vax 6510. We call the iteration between restarts a "run."

Example 1. We let the matrix have $1, 2, 3, \ldots, 999, 1000$ on the main diagonal and as mentioned above, the super diagonal elements have 0.1's. For this matrix, the quantity $||Z||||Z^{-1}||$ in (3) and (4) is small (about 1.2). The new GMRES with eigenvectors method using $m = 21$ and $k = 4$ (21 Krylov vectors and four approximate eigenvectors) is compared to GMRES(25). Thus the same size subspaces are used. After 12 runs, the eigenvector method has a residual norm of 0.42e-9 compared to 0.15e-4 for standard GMRES. See Fig. 1 for a graph of the convergence. After iteration 100, the eigenvector method converges more than twice as fast. This is roughly as predicted by (7), even though the Krylov portion of the subspace is smaller for the eigenvector method than for the regular GMRES. At iteration 100, after four runs, the eigenvector approximations are not very accurate. The approximate eigenvalues are 1.01, 2.20, 3.86, and 6.10, and the corresponding residual norms range from 0.13 to 1.9. But already the eigenvectors are accurate enough to assist convergence. After eight runs, the approximate eigenvalues are more accurate with from 8 to 2 significant digits and residual norms from 0.13e-3 to 0.17.

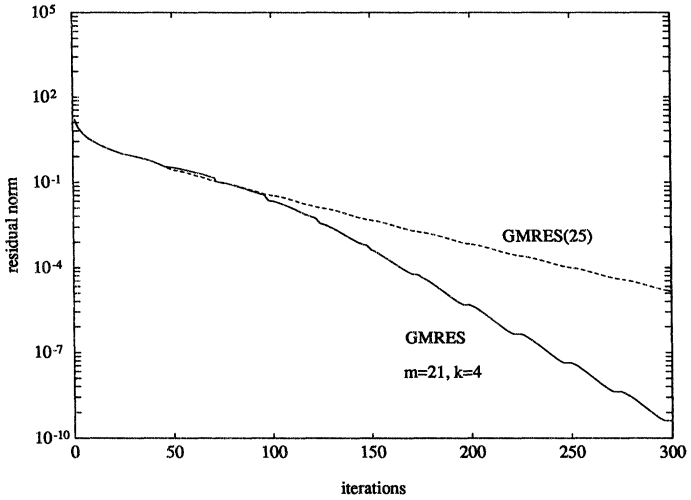

FIG. 1. GMRES *vs.* GMRES *with eigenvectors.*

Next, methods requiring about the same storage are compared. The eigenvector method with $m = 17$ and $k = 4$ reaches residual norm of 0.22e-6 after 12 runs (see Table 1). This is still better than GMRES(25), even though smaller subspaces are used and far less matrix-vector products are required. If an equal number of matrix-vector

TABLE 1.
Eigenvalues $1, 2, 3, \ldots, 1000$.

			Residual norms		
m	k	l	Initial	After 12 runs	After 300 matrix-vector products
25	0	25	0.31e+2	0.15e-4	0.15e-4
21	4	25	0.31e+2	0.42e-9	0.35e-11
17	4	21	0.31e+2	0.22e-6	0.18e-10
21	2	23	0.31e+2	0.67e-7	0.20e-8
19	3	22	0.31e+2	0.76e-7	0.14e-9
13	6	19	0.31e+2	0.19e-4	0.23e-11
9	8	17	0.31e+2	0.25e-2	0.40e-11

products are taken, the eigenvector method with $m = 17$ and $k = 4$ is much further ahead. After 300 matrix-vector products, it attains 0.18e-10 versus 0.15e-4. Table 1 also gives results with different choices of k but with the same storage (the same $m+2k$). Using just two approximate eigenvectors gives the lowest residual norm after 12 runs. However, if one is most interested in the number of matrix-vector products, using six eigenvectors is better, even though the Krylov subspace has dimension of only $m = 13$.

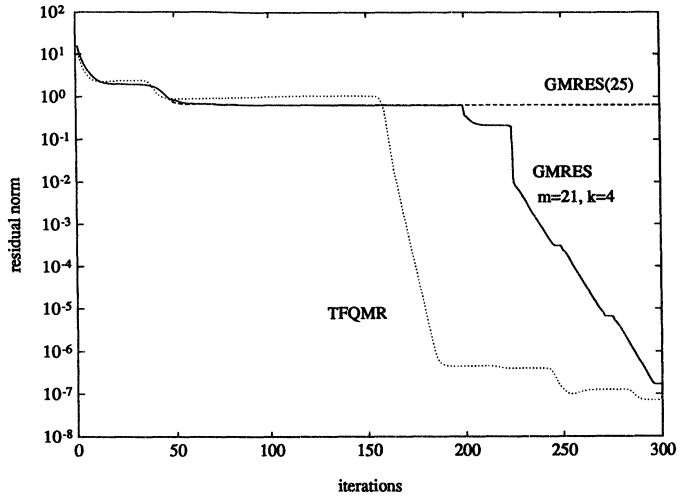

FIG. 2. *Comparison for Example* 2 (TFQMR *uses two* mvp *per iteration*).

Example 2. The next matrix has some very small eigenvalues that make the problem difficult. The entries on the main diagonal are $0.01, 0.02, 0.03, 0.04, 10, 11, 12, 13, \ldots, 1005$ (taking on all integer values from 10–1005). See Table 2 and Fig. 2 for the computational results. Figure 2 also includes the TFQMR method; this is discussed in Example 7. It appears that the regular GMRES method will not converge. Going past 12 runs, there is no further improvement in the residual norm. The four small eigenvalues make this problem too difficult for Krylov subspaces of dimension 25. The GMRES with eigenvectors method also stalls out for a while. Considering the case $m = 21$ and $k = 4$, the residual norm only improves from 0.65–0.63 during runs four through seven. But finally after run seven, there are rough approximations to *all four* of the small eigenvalues. With the corresponding approximate eigenvectors in the subspace, the convergence rate is soon fairly rapid.

Example 3. Here an indefinite matrix is used. The diagonal entries are $-2, -1, 1, 2, 3, 4, \ldots, 997, 998$. In this situation, the eigenvector method is much better than regular

TABLE 2.
Eigenvalues $0.01, 0.02, 0.03, 0.04, 10, 11, 12, \ldots, 1005$.

Residual norms

m	k	l	Initial	After 12 runs	After 300 matrix-vector products
25	0	25	0.32e+2	0.64	0.64
21	4	25	0.32e+2	0.17e-6	0.91e-10
17	4	21	0.32e+2	0.18e-2	0.47e-9

TABLE 3.
Eigenvalues $-2, -1, 1, 2, \ldots, 998$.

Residual norms

m	k	l	Initial	After 5 runs	After 10 runs	After 15 runs	After 20runs	After 500 mvp's
25	0	25	0.32e+2	0.90	0.58	0.38	0.24	0.24
21	4	25	0.32e+2	0.22	0.83e-4	0.21e-7	0.54e-11	0.14e-13
17	4	21	0.32e+2	0.53	0.24e-2	0.50e-5	0.11e-7	0.64e-13

GMRES; see Table 3. With $k = 4$, approximations are developed to the eigenvalues -2, -1, 1, and 2. This eliminates the two negative eigenvalues and effectively turns the indefinite problem into a definite one.

Example 4. This example is designed to be difficult for the GMRES with eigenvectors method. Let the matrix have diagonal elements $1, 1.01, 1.02, 1.03, 1.04, 2, 3, 4, 5, \ldots, 995, 996$ (see Table 4). Removing the four smallest eigenvalues does not have much effect on the spread of eigenvalues. The two methods are roughly equivalent when using the same size subspace. With equal storage, standard GMRES is one order of magnitude better after 15 runs and the eigenvector method is a little better after each has taken 375 matrix-vector products. Even in this difficult situation, using eigenvectors does not substantially decrease efficiency. In another test with five approximate eigenvectors ($m = 20$, $k = 5$), the method does not improve. The reason is that approximations to the five smallest eigenvalues do not develop in time to substantially help. The eigenvalues are so close together that they are difficult to compute (after 15 runs, the five approximate eigenpairs have residual norms no smaller than 0.05).

TABLE 4.
Eigenvalues $1, 1.01, 1.02, 1.03, 1.04, 2, 3, \ldots, 996$.

Residual norms

m	k	l	Initial	After 5 runs	After 10 runs	After 15 runs	After 375 mvp's
25	0	25	0.32e+2	0.10e-1	0.58e-4	0.48e-6	0.48e-6
21	4	25	0.32e+2	0.67e-2	0.17e-4	0.12e-6	0.16e-7
17	4	21	0.32e+2	0.22e-1	0.24e-3	0.50e-5	0.77e-7

Example 5. Let the matrix be block diagonal with eigenvalues equally spaced around a circle in the complex plane with center at (1,0) and radius 0.99, starting at 0.01. The matrix is normal with blocks of size 2 or 1. With $n = 100$, the approximate eigenvectors are very helpful. Theoretical convergence results for GMRES often consider an ellipse or circle [32] containing the eigenvalues. The smallest circle containing all of the eigenvalues does not change when a few of the eigenvalues nearest the origin are eliminated. Nevertheless, the convergence is improved. With a change to $n = 200$,

TABLE 5.
Eigenvalues in a circle.

Residual norms for $n = 100$

m	k	l	Initial	After 5 runs	After 10 runs	After 15 runs	After 20 runs	After 500 mvp's
25	0	25	0.10e+2	0.47	0.13	0.38e-1	0.11e-1	0.11e-1
21	4	25	0.10e+2	0.52	0.45e-2	0.85e-5	0.16e-7	0.26e-9
17	4	21	0.10e+2	0.65	0.12	0.98e-3	0.77e-5	0.13e-8

Residual norms for $n = 200$

m	k	l	Initial	After 10 runs	After 20 runs	After 30 runs	After 40 runs	After 1000 mvp's
25	0	25	0.14e+2	0.19	0.15e-1	0.12e-2	0.10e-3	0.10e-3
21	4	25	0.14e+2	0.25	0.30e-1	0.36e-2	0.44e-3	0.92e-4
17	4	21	0.14e+2	0.38	0.68e-1	0.12e-1	0.22e-2	0.94e-4

the problem of finding the eigenvalues is tougher, because they are closer together. Here the eigenvalue problem is apparently more difficult than the linear equations problem and the eigenvector approximations never become accurate enough to really help (see Table 5). After run 10, the approximation to the smallest eigenvalue has residual norm 0.26e-1 and this does not improve during the next 30 runs. For comparison, during the test with $n = 100$, the approximation to the smallest eigenvalue has residual norm 0.13e-3 after 10 runs, and it slowly improves.

Example 6. This example has a standard test matrix (see Table 6). The problem is from the finite difference discretization of the partial differential equation $u_{xx}+u_{yy}+Du_x = -(41)^2$ on the unit square with $u = 0$ on the boundary. Central differences are used. The mesh spacing is $h = \frac{1}{41}$, so $n = 1600$. Tests are done with increasing degrees of nonsymmetry: $D = 1$, $D = 41$ and $D = (41)^2$. For the first two, the results are similar to Example 1. Using eigenvectors is definitely worthwhile. For the last test, the approximate eigenvectors are not particularly useful. There are no eigenvalues near to the origin, and the algorithm has trouble computing the ones that are closest to the origin. Perhaps this is because there are several about the same distance away.

The rest of this section has comparisons with the quasiminimal residual, or QMR, method of Freund and Nachtigal [12] and the QMR transpose-free variant, TFQMR [11]. Because these methods do not restart, they have an advantage for difficult problems that require large subspaces. The quasiminimization in QMR controls much of the instability, but it does not insure that the subspace is used as effectively as in GMRES.

In the tests here, TFQMR uses standard weights [11] and QMR has unit weights. The right-hand side of all 1.0's is used for both of the initial vectors. The look-ahead feature is not used [12], [22]. Convergence of QMR and TFQMR is monitored with the approximate residual norms given in [11], [12]. The matrices from Example 5 are left out because of their small size: while QMR reaches convergence faster than the GMRES methods, it is only after the dimension of the subspace is larger than the size of the problem (actually QMR is unstable, but it does converge when an initial vector is changed).

Example 7. Table 7 has comparisons between GMRES(25), the modified version of GMRES with 21 Krylov vectors and four approximate eigenvectors, and the two versions of QMR. Both the number of iterations and the number of matrix-vector products are given. The QMR methods require two matrix-vector products per iteration (and one for the residual norm at the end), while GMRES requires one

TABLE 6.
Finite difference matrix.

Residual norms for D=1

m	k	l	Initial	After 8 runs	After 200 matrix-vector products
25	0	25	0.40e+2	0.13e-3	0.13e-3
21	4	25	0.40e+2	0.52e-10	0.28e-12
17	4	21	0.40e+2	0.33e-7	0.12e-11

Residual norms for D=41

m	k	l	Initial	After 8 runs	After 200 matrix-vector products
25	0	25	0.40e+2	0.70e-4	0.70e-4
21	4	25	0.40e+2	0.33e-9	0.64e-11
17	4	21	0.40e+2	0.95e-7	0.90e-11

Residual norms for D=$(41)^2$

m	k	l	Initial	After 20 runs	After 500 matrix-vector products
25	0	25	0.40e+2	0.98e-7	0.98e-7
21	4	25	0.40e+2	0.71e-8	0.16e-9
17	4	21	0.40e+2	0.57e-6	0.32e-9

matrix-vector product per iteration, and the GMRES with eigenvectors method requires one for each Krylov vector but no matrix-vector products for the approximate eigenvectors. For these examples, TFQMR always converges in the least number of iterations, while GMRES with eigenvectors always uses the fewest matrix-vector products. Figure 2 also shows TFQMR converging in less iterations for the matrix from Example 2, but GMRES with eigenvectors uses less than half the number of matrix-vector products (see Table 7). For one more comparison not included in the table, the matrix in Example 2 is modified to have ten small eigenvalues from 0.01–0.10 and the rest from 10–999. For this problem, GMRES with eigenvectors using $m = 15$ and $k = 10$ was compared to TFQMR. The modified GMRES approach required fewer iterations, 707 compared to 1109, and it used less than one-fifth the number of matrix-vector products, 437–2219. These tests do not indicate that one method is better than the other, but they do show that GMRES with eigenvectors is worth considering, especially in situations where the matrix-vector product is expensive.

Example 8. The GMRES with eigenvectors method may also be particularly useful when there are several similar systems of linear equations or several right-hand sides. One such case occurs in solving time-dependent differential equations. In the following tests, a simple time-dependent problem is considered. Let the differential equation be $u_t = u_{xx} + u_{yy} + u_x$, on the unit square with t going from 0.0–1.0. The initial condition is u(x,y,0) = 1.0, the boundary condition is $u = 0$ on the boundary. The backward difference method is used with time steps of 0.1, and discretization of the spacial variables is as in Example 6. The termination criterion while solving the systems of linear equations is $||r|| < 10^{-4}$. Table 8 gives the number of iterations at each time step and the total number of iterations and matrix-vector products for the QMR methods, GMRES(20), and GMRES with eigenvectors with m=17 and k=3. The QMR methods have a tendency to start slowly, then converge rapidly. This can be a disadvantage when several systems are solved to low accuracy. Meanwhile the GMRES with eigenvectors method has an advantage, because it can use the approximate eigenvectors from the previous time step to help at the current one. GMRES with eigenvectors performs better than the QMR methods for this problem.

TABLE 7.
Comparison to QMR.

Iterations and matrix-vector products to reach $||r|| < 10^{-6}$

			GMRES m=25, k=0	GMRES w/e.vectors m=21, k=4	QMR	TFQMR
Ex. 1		it's	370	214	180	119
		mvp's	370	186	361	239
Ex. 2		it's	-	286	339	250
		mvp's	-	246	679	501
Ex. 3		it's	-	339	215	160
		mvp's	-	291	431	321
Ex. 4		it's	355	325	252	162
		mvp's	355	277	505	325
Ex. 6	D=1	it's	278	132	125	95
		mvp's	278	116	251	191
	D=41	it's	300	149	100	(83)*
		mvp's	300	134	201	(167)
	$D=41^2$	it's	441	382	366	218
		mvp's	441	326	733	437

* Because of instability, a different left initial vector was used.

TABLE 8.
Time-dependent problem.

Iterations for each time step

	GMRES m=20, k=0	GMRES w/e.vectors m=17, k=3	QMR	TFQMR
t = 0.1	130	91	92	62
t = 0.2	113	34	71	70
t = 0.3	48	27	60	67
t = 0.4	28	20	38	49
t = 0.5	14	14	29	44
t = 0.6	15	12	22	27
t = 0.7	13	14	11	24
t = 0.8	11	11	13	16
t = 0.9	6	6	11	18
t = 1.0	1	2	2	1
Total iter.'s	379	231	349	378
Total mvp's	379	213	708	766

5. Attempt at an automatic procedure. Here we deal with two questions. How many approximate eigenvectors should be used, and should the approximate

eigenvectors be discarded at some point? However, it is difficult to give answers that apply to all matrices.

For determining the proper number of approximate eigenvectors to use, we consider the model eigenvalue distribution $1, 2, \ldots, n$. If we assume that the storage is fixed, then methods with the same value of $m + 2k$ should be compared. We compute k that gives the lowest value of

$$\left(\left(\frac{\sqrt{\kappa_e}+1}{\sqrt{\kappa_e}-1}\right)^m + \left(\frac{\sqrt{\kappa_e}-1}{\sqrt{\kappa_e}+1}\right)^m\right)^{-1},$$

where $\kappa_e \equiv \frac{\lambda_n}{\lambda_{k+1}}$. This formula comes from Theorem 1, but with the more accurate bound given in the first part of (2). After doing some comparisons, we find that if $m + 2k$ is given, the best value is approximately

$$k = \frac{m+2k}{7}.$$

For values of $m + 2k$ greater than 50, slightly more should be used, and for values less than 20, the number should be rounded down.

It would be desirable to have a code that adaptively increases or decreases the number of approximate eigenvectors being used. However, it is difficult to determine if adding another eigenvector will help when no accurate approximation is available for the next eigenvalue. For now we just consider the possibility of releasing the approximate eigenvectors and going back to standard GMRES. This switch should be done if the eigenvectors are not helping. Even beneficial eigenvectors may lose their effectiveness once components of the residual vector in the directions of those eigenvectors have been purged.

One possibility is to check how effective the addition of the approximate eigenvectors is in lowering the residual norm. This information is readily available. The amount the eigenvectors lower the residual norm can be compared with the amount the residual norm decreases in the previous k Krylov iterations. This suggests the test: switch when

$$||r_m|| - ||r_{m+k}|| < ||r_{m-k}|| - ||r_m||.$$

However, it turns out that the beneficial effect of the eigenvectors is not fully reflected in how they lower the residual norm. They also enable the Krylov vectors to be more effective. This makes it difficult to determine whether the eigenvectors are useful or not. A factor can be added in

$$||r_m|| - ||r_{m+k}|| < 0.2\big(||r_{m-k}|| - ||r_m||\big). \tag{17}$$

This is effective for Examples 1 and 2, but it releases much too early for Example 3.

We consider the addition of some more complicated tests that involve the accuracy of the eigenvectors. For the approximate eigenvector y_i, denote the eigenvector residual norm by rne_i. Then

$$rne_i = \frac{||Ay_i - \rho_i y_i||}{||y_i||},$$

where

$$\rho_i = \frac{y_i^* A y_i}{y_i^* y_i} = \frac{g_i^* F^* g_i}{y_i^* y_i}.$$

The eigenvector residual norm can be computed explicitly since Ay_i and y_i have been formed, and there is also a formula involving F^*, G, ρ_i, and g_i. To determine if the

eigenvectors are not helpful, we check that the improvement of the best approximate eigenvector is at least one-tenth as much as the improvement of the linear equations approximate solution during a restarted run of the method. So the criterion is

$$(18)\quad -\log_{10}(rne_1)_{\text{new}} + \log_{10}(rne_1)_{\text{old}} < 0.1\big(-\log_{10}||r_{m+k}||_{\text{new}} + \log_{10}||r_{m+k}||_{\text{old}}\big).$$

The switch is done if both (17) and (18) are satisfied.

In addition, we check to see if the eigenvectors are no longer useful. We use the test

$$(19)\quad -\log_{10}\Big(\frac{rne_{(k-1)}}{\text{largest element in matrix}}\Big) - \log_{10}||r_{m+k}|| > -\log_{10}(r\text{tol}),$$

where $r$tol is the desired residual norm for the linear equations problem and $rne_{(k-1)}$ is the residual norm of the eigenvector that is second to last in accuracy. This test roughly follows from Theorem 2. The idea is that the magnitudes of the components of the solution to the restarted problem are approximately $||r_{m+k}||$. If an approximate eigenvector is fully used, then the component in that direction will be reduced from this size by approximately the accuracy of the eigenvector. So the resulting size of the component is reflected in the left-hand side of inequality (19). Once these components have been reduced to the desired magnitude specified by $r$tol, the approximimate eigenvectors are no longer needed. The switch is done if both (17) and (19) are satisfied.

In the tests that follow, the method begins with $m = 21$ and $k = 4$, then switches to GMRES(25). However, we note that if storage is the limiting factor, the switch could have been to GMRES(29). For the problem in Example 1 with $r\text{tol} = 1.\text{e} - 9$, the switch is made when (17) and (19) are satisfied after eight runs. Then after 12 runs the residual norm is 0.26e-9. This is just as good as if eigenvectors are kept for all of the runs. See Table 9. For Example 2 with $r\text{tol} = 1.\text{e} - 6$, the switch is made after 11 of 12 runs, and the method does better on the last run without the eigenvalues. For Example 3 with $r\text{tol} = 1.\text{e} - 10$, the switch is after 11 runs. The residual norm after 20 runs is 0.47e-8, not as good as the residual norm of 0.54e-11 without switching. In this case the eigevectors are very important and the switch test is triggered too soon. For Example 5 with $n = 100$ and for Example 6 with $D = 1$ and $D = 41$, the switch is not particularly significant.

Next for Example 4, Example 5 with $n = 200$, and Example 6 with $D = (41)^2$, the eigenvectors are not particularly useful and the switch is done when (17) and (18) are both satisfied. For Example 5, this happens after just 11 of 40 runs, because the approximate eigenvectors are not improving. The switch also works well for Example 6 with $D = (41)^2$.

More complicated adaptive procedures can be implemented. One possibility is to adaptively choose the number of eigenvectors to be used. Also the eigenvectors could be released individually as they converge. However, even the simpler procedures described in this section may not work for all problems.

6. Conclusion. Forming and using approximate eigenvectors can improve the convergence of restarted GMRES. Even just a few eigenvectors can make a big difference if the matrix has both small and large eigenvalues. Once the eigenvectors converge, the corresponding eigenvalues are essentially removed or deflated from the spectrum. And the approximate eigenvectors can improve convergence even before they are accurate.

TABLE 9.
Discarding eigenvectors.

	rtol	total runs	switch after	res. norm	res. norm w/o switch
Ex. 1	1.e-9	12	8	0.26e-9	0.50e-9
Ex. 2	1.e-6	12	11	0.36e-8	0.17e-6
Ex. 3	1.e-10	20	11	0.47e-8	0.54e-11
Ex. 5, n=100	1.e-8	20	17	0.81e-8	0.16e-7
Ex. 6, D=1	1.e-10	8	6	0.30e-9	0.52e-10
Ex. 6, D=41	1.e-10	8	8	0.33e-9	0.33e-9
Ex. 4		15	9	0.84e-7	0.12e-6
Ex. 5, n=200		40	11	0.13e-3	0.44e-3
Ex. 6, D=$(41)^2$		20	5	0.94e-8	0.71e-8

This method is useful for any problem that is difficult because of small eigenvalues. However, there are several situations where it is particularly beneficial. If the matrix-vector product is expensive, approximate eigenvectors can be used with relatively little extra expense. The method is also particularly effective when the spectrum of the matrix is well-behaved except for a few eigenvalues, such as in the case of having only a few negative eigenvalues or only a few eigenvalues with negative real parts. Also, if GMRES is used with a problem that has more than one right-hand side, then the eigenvectors can be computed once and used efficiently for all of the right-hand sides.

The method is not really needed for easy problems where few restarts are used. It also may not help if the problem is hard because of eigenvalues scattered around the complex plane. Another possibly related situation is when the small eigenvalues are less separated from rest of the spectrum than the spectrum is separated from zero. Then the eigenvalue problem is tougher than the linear equations problem. If the eigenvectors are not converging, then they probably should be discarded.

Acknowledgments. The author would like to thank Wayne Joubert for helpful discussions and thank the referees for their many constructive comments.

REFERENCES

[1] W. E. ARNOLDI, *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quart. Appl. Math., 9 (1951), pp. 17–29.

[2] S. F. ASHBY, *Polynomial Preconditioning for Conjugate Gradient Methods*, Ph.D. thesis, Dept. of Computer Science, University of Illinois, Urbana, 1987.

[3] O. AXELSSON, *A survey of preconditioned iterative methods for linear systems of equations*, BIT, 25 (1985), pp. 166–187.

[4] O. AXELSSON AND G. LINDSKOG, *On the rate of convergence of the preconditioned conjugate gradient method*, Numer. Math., 48 (1986), pp. 499–523.

[5] A. K. CLINE, *Several Observations on the Use of Conjugate Gradient Methods*, ICASE Report 76-22, NASA Langley Research Center, Hampton, VA, 1976.

[6] P. CONCUS, G. H. GOLUB, AND G. MEURANT, *Block preconditioners for the conjugate gradient method*, SIAM J. Sci. Statist. Comput., 6 (1985), pp. 220–252.

[7] P. CONCUS, G. H. GOLUB, AND D. P. O'LEARY, *A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations*, Sparse Matrix Computations, J. R. Bunch and D. J. Rose, eds., Academic Press, New York, 1976, pp. 309–332.

[8] E. CRAIG, *The N-step iteration procedure*, J. Math. Phys., 34 (1955), pp. 64–73.

[9] H. C. ELMAN, *Iterative Methods for Large Sparse Nonsymmetric Systems of Linear Equations*, Ph.D. thesis, Computer Science Dept., Yale University, New Haven, CT, 1982.

[10] H. C. ELMAN, Y. SAAD, AND P. E. SAYLOR, *A hybrid Chebyshev Krylov subspace algorithm*

for solving nonsymmetric systems of linear equations, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 840–855.

[11] R. W. FREUND, *A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems*, SIAM J. Sci. Statist. Comput., 14 (1993), pp. 470–482.

[12] R. W. FREUND AND N. NACHTIGAL, QMR*: a quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., 60 (1991), pp. 315–339.

[13] G. H. GOLUB AND D. P. O'LEARY, *Some history of the conjugate gradient and Lanczos methods*, SIAM Rev., 31 (1989), pp. 50–100.

[14] M. GUTKNECHT, *A completed theory of the unsymmetric Lanczos process and related algorithms, Part* I, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 594–639.

[15] L. A. HAGEMAN AND D. M. YOUNG, *Applied Iterative Methods*, Academic Press, New York, 1981.

[16] M. R. HESTENES, *The conjugate gradient method for solving linear systems*, in Proc. Symp. Appl. Math., Vol. 6, Numerical Analysis, American Mathematical Society, New York, 1956.

[17] M. R. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 409–436.

[18] W. JOUBERT, *Lanczos methods for the solution of nonsymmetric systems of linear equations*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 926–943.

[19] C. LANCZOS, *An iterative method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Standards, 45 (1950), pp. 255–282.

[20] ———, *Solution of systems of linear equations by minimized iterations*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 33–53.

[21] J. G. LEWIS AND D. J. PIERCE, *Recent research in iterative methods at Boeing*, Comput. Phys. Comm., 53 (1989), pp. 213–221.

[22] T. A. MANTEUFFEL, *The Tchebychev iteration for nonsymmetric linear systems*, Numer. Math., 28 (1977), pp. 307–327.

[23] J. A. MEIJERINK AND H. A. VAN DER VORST, *An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix*, Math. Comp., 31 (1977), pp. 148–162.

[24] R. B. MORGAN, *Computing interior eigenvalues of large matrices*, Linear Algebra Appl., 154-156 (1991), pp. 289–309.

[25] R. B. MORGAN AND M. ZENG, *Estimates for interior eigenvalues of large nonsymmetric matrices*, submitted.

[26] N. M. NACHTIGAL, S. C. REDDY, AND L. N. TREFETHEN, *How fast are nonsymmetric matrix iterations?*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 778–795.

[27] A. NAVARRA, *An application of* GMRES *to indefinite linear problems in meteorology*, Comput. Phys. Comm., 53 (1989), pp. 321–327.

[28] C. C. PAIGE AND M. A. SAUNDERS, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.

[29] B. N. PARLETT, D. R. TAYLOR, AND Z. A. LIU, *A look-ahead Lanczos algorithm for unsymmetric matrices*, Math. Comp., 44 (1985), pp. 105–124.

[30] Y. SAAD, *Variations on Arnoldi's method for computing eigenelements of large unsymmetric matrices*, Linear Algebra Appl., 34 (1980), pp. 269–295.

[31] Y. SAAD, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comp., 37 (1981), pp. 105–126.

[32] Y. SAAD AND M. H. SCHULTZ, GMRES*: a generalized minimum residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.

[33] B. T. SMITH, J. M. BOYLE, Y. IKEBE, B. C. KLEMA, AND C. B. MOLER, *Matrix Eigensystems Routines: EISPACK Guide*, 2nd ed., Springer-Verlag, New York, NY, 1970.

[34] P. SONNEVELD, CGS*, a fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 10 (1989), pp. 36–52.

[35] A. VAN DER SLUIS AND H. A. VAN DER VORST, *The rate of convergence of conjugate gradients*, Numer. Math., 48 (1986), pp. 543–560.

[36] H. A. VAN DER VORST, Bi-CGSTAB*: A fast and smoothly converging variant of* Bi-CG *for the solution of nonsymmetric linear equations*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 631–644.

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