GENERALIZATIONS OF DAVIDSON'S METHOD FOR COMPUTING EIGENVALUES OF SPARSE SYMMETRIC MATRICES

Ronald B. Morgan[1] and David S. Scott

Department of Computer Sciences
University of Texas at Austin
Austin, TX 78712

TR-84-25 September 1984

Abstract

This paper analyzes Davidson's method for computing a few eigenpairs of large sparse symmetric matrices. An explanation is given for why Davidson's method often performs well but occasionally performs very badly. Davidson's method is then generalized to a method which offers a powerful way of applying preconditioning techniques developed for solving systems of linear equations to solving eigenvalue problems.

Work of the second author was partially supported by the Applied Mathematical Sciences Research Program, Office of Energy Research, U.S. Department of Energy, under contract W-7405-eng-26.

[1]Mathematics Department, University of Texas at Austin.

Table of Contents

List of Tables

1. Introduction

The Lanczos algorithm is a powerful technique for computing a few eigenvalues of a symmetric matrix A. If the matrix (A - σ) can be factored for one or more values of σ near the desired eigenvalues then the Lanczos algorithm can be used with the inverted operator and convergence will be very rapid. Otherwise the Lanczos algorithm can be used with the original matrix A but convergence can be very slow.

Slow convergence can also plague the conjugate gradients method for solving systems of linear equations which is an analog of the Lanczos algorithm. Convergence of the conjugate gradients algorithm can be accelerated by computing and using an approximate inverse (preconditioner). Much work has been done developing effective preconditioning techniques ([3], [5], [8], [1]). Unfortunately preconditioning cannot be directly applied to eigenvalue problems. If the preconditioner multiplies both sides of the equation $Az = \lambda z$ then the problem becomes a generalized eigenvalue problem which is no easier to solve. If only A is multiplied by the preconditioner then the answers are changed.

One approach for using preconditioners on eigenvalue problems is to convert them to linear equation problems by using inverse iteration or the Rayleigh quotient iteration and then use preconditioned conjugate gradients (actually SYMMLQ [6] since the matrices involved will be indefinite). This approach was investigated by Szyld [9]. A different approach is Davidson's method [2] which can be interpreted as a method for using diagonal preconditioning in solving eigenvalue problems. This paper analyzes Davidson's method from this point of view and then generalizes it to obtain a method which uses more powerful types of preconditioners.

2. Davidson's Method

Davidson [2] introduced a new method for computing a few eigenvalues of sparse symmetric matrices arising in quantum chemistry calculations. The standard solution technique for such problems is the Lanczos algorithm ([7] chap. 13) which is a clever implementation of the Rayleigh-Ritz procedure applied to a Krylov subspace. Davidson's method also uses the Rayleigh-Ritz procedure (see [7] p. 213) but on a non-Krylov subspace. Formally Davidson's method is as follows.

Davidson's Method

Initialize: Let A be the matrix of interest and let D be the diagonal of A. Choose an initial trial space $H_k = \text{span}(h_1, h_2, \ldots h_k)$ and compute (y_k, θ_k) the best approximation to the eigenpair of interest using the Rayleigh-Ritz procedure and compute the residual vector

$$r_k = Ay_k - y_k\theta_k.$$

Then FOR $j = k+1, k+2, \ldots$ until convergence DO 1 to 4

1. Compute $h_j = (D - \theta_{j-1})^{-1}r_{j-1}$
2. Set $H_j = \text{span}(H_{j-1}, h_j)$
3. Compute (y_j, θ_j) from H_j using the Rayleigh-Ritz procedure
4. Compute the residual $r_j = (A - \theta_j)y_j$

Convergence is measured by the norm of the residual vector.

Thus the new trial vector is just $(D - \theta)^{-1}(A - \theta)y$. Except for the cost of forming the matrix vector product (which is the same for either method), this algorithm is much more expensive per step than the Lanczos algorithm since a full Gram-Schmidt process is needed to compute an orthogonal basis for the space H and a full (rather than tridiagonal) reduced matrix is generated by the Rayleigh-Ritz procedure. Despite this higher overhead, Davidson reported favorable results for his method compared to the Lanczos algorithm for problems arising in molecular energy calculations. In one example of dimension 372, Davidson's method reduced the residual norm to 1e-6 in 10 iterations while 28 iterations of Lanczos reduced the residual norm to only 2e-2. The improvement is due entirely to the premultiplication by the matrix $(D - \theta_j)^{-1}$ in step 1, since without this perturbation, Davidson's method would just reduce to a very expensive way of implementing the Lanczos algorithm (provided $k = 1$).

The new trial vector obtained by Davidson's method is the correction which would be obtained by one step of the Jacobi method for solving the system of equations

$$(A - \theta)x = 0$$

with y_i as the initial guess for x. This looks a little strange since in general $A - \theta$ is not singular and the only solution to the system of equations is $x = 0$. A more satisfying explanation of the algorithm is as follows. Let (y, θ) be the current approximation to the desired eigenpair. For a given coordinate i, the best improvement which can be made in y by perturbing its i(th) component can be determined by the Rayleigh-Ritz procedure. The best approximations to eigenpairs of A obtainable from the space spanned by $\{y, e_i\}$ are obtained by solving the 2x2 generalized eigenvalue problem $H_i - \alpha W_i$ where

$$H_i = \begin{bmatrix} y^T A y & y^T A e_i \\ e_i^T A y & e_i^T A e_i \end{bmatrix} = \begin{bmatrix} \theta & (Ay)_i \\ (Ay)_i & a_{ii} \end{bmatrix}$$

$$W_i = \begin{bmatrix} y^T y & y^T e_i \\ e_i^T y & e_i^T i \end{bmatrix} = \begin{bmatrix} 1 & y_i \\ y_i & 1 \end{bmatrix}$$

of the 2x2 problem or equivalently (setting $\alpha = \theta$) the matrix

$$H_i - \theta W_i = \begin{bmatrix} 0 & (Ay - y\theta) \\ (Ay - y\theta)_i & a_{ii} - \theta \end{bmatrix}$$

$$= \begin{bmatrix} 0 & r_i \\ r_i & d_i - \theta \end{bmatrix}$$

is close to singular where r_i is the ith component of the residual vector and d_i is the ith diagonal element of A. If we are near convergence then the residual will be small, and the small eigenvector of H_i-θW will be approximately

$$s = (1, \ -(d_i-\theta)^{-1} r_i)^T.$$

Thus the best linear combination of y and e_i is (approximately)

$$z = y \ -(d_i-\theta)^{-1} r_i e_i.$$

Davidson's method just lumps all of these perturbations into one vector and adds this composite vector to the trial space. (The dropping of the minus sign is of no account since only the trial space is important for the Rayleigh-Ritz procedure not the particular basis chosen.)

Example 1.

To compare Davidson's method to the Lanczos algorithm, the smallest eigenpair of a symmetric matrix A of order 20 was computed using both methods. A was tridiagonal except that a_{1n} and a_{n1} were nonzero. $a_{ii} = i$ for all i while all other nonzero elements were 1. The starting vector was $p_1 = (1, 0.1, 0.1, ..., 0.1)^T$ which is moderately but not exceptionally accurate. Table 2-1 illustrates the behavior of the methods. The Ritz value for Davidson's method was accurate to 9 decimal digits at step 10 while the Lanczos value at step 10 was accurate only to 2 digits. This behavior is very similar to that reported by Davidson [2]. Note that the Lanczos algorithm has better global convergence than Davidson's method (better for the first five steps).

In the next section we examine our derivation of Davidson's method in more detail.

Table 2-1: A Comparison of the Lanczos and Davidson Methods

step	Davidson Ritz value	Residual norm	Lanczos Ritz value	Residual norm
1	3.23529	5.27	3.23529	5.27
2	3.17006	3.17	1.21302	1.83
3	1.65718	1.80	.784054	1.34
4	1.48600	1.78	.476551	1.07
5	.291006	.953	.320862	.664
6	.223536	.0764	.2603809	.423
7	.222866	.01177	.2352622	.264
8	.222847	.00241	.2263713	.149
9	.222846	.000229	.2237563	.0783
10	.222846	.0000249	.2230518	.0381

3. Local Analysis of Davidson's Method

One major assumption in the derivation of Davidson's method is the form of s, the eigenvector of interest of the 2x2 problem H_i - αW_i. Let $\pi_i = r_i/(d_i - \theta)$ be the components of the Davidson vector p. If $\pi_i < 1$ then the eigenvalue near θ and corresponding eigenvector can be expanded in a power series in π_i.

$$\alpha = \theta - r_i\pi_i - 2r_iy_i\pi_i^2 + O(\pi_i^3)$$

and normalizing the first component of s to be 1,

$$s = (1,\ -\pi_i - y_i\pi_i^2 + O(\pi_i^3))^T.$$

(It is interesting to note that the lowest order terms do not depend on y_i.) Thus Davidson's method does implement the correct first order perturbation correction provided that $\pi_i < 1$.

Asymptotically r_i converges to zero while d_i-θ converges to d_i - λ, where λ is the desired eigenvalue. Thus π_i converges to zero for all i **unless** λ is equal to some diagonal element of A. If $\lambda = d_i$ for some i then the behavior of π_i depends on whether or not the corresponding eigenvector is e_i. In any case θ, the Rayleigh quotient of y, will satisfy

$$d_i - \theta = O(\|r\|^2)$$

(see [7] p. 222). If $z = e_i$ then $y_i \approx 1$, r_i will be just $(d_i - \theta)y_i$ and π_i will approach 1. Otherwise r_i will be some constant times $\|r\|$ and π_i will diverge to infinity. In either case Davidson's method may perform badly.

Example 2.

The matrix A from example 1 was modified so that $a_{12} = a_{21} = a_{1n} = a_{n1} = 0$. This made the smallest eigenvalue of A equal to 1 with e_1 as the corresponding eigenvector. The next smallest eigenvalue is 1.2538... . The same starting vector was used. By step 8 of the algorithm the second smallest eigenvalue had been computed to 8 decimal places but no approximation to the smallest eigenvalue had appeared at all. On step 9 a poor approximation to the smallest eigenvalue had appeared (1.21315) but had only converged to 1.0285 by step 16.

Example 3.

To obtain an example of the other kind of unusual behavior, the matrix A from Example 1 was modified by deleting its last row and column. The resulting matrix has 10 both as an eigenvalue and as a diagonal

element. The starting vector had all equal components except the tenth which was ten times larger. By step fourteen the desired eigenvector approximation had a residual of .000587. This is almost as fast as Example 1 and much faster than Example 2. This difference in behavior will be examined in the next section.

4. Convergence of Davidson's Method

Despite the dramatic results reported by Davidson for molecular energy calculations, Kalamboukis [4] reported that Davidson's method converged no faster than Lanczos on nuclear modeling problems. Since the overhead is much higher in Davidson's method, Kalamboukis recommended that Lanczos be used for this type of problem. Kalamboukis also suggested that the differing behavior of Davidson's algorithm could be explained by the degree of diagonal dominance of the matrices involved--the more diagonally dominant the matrix was, the better Davidson's method worked. This is not entirely true. If the diagonal of A is constant, then Davidson's method is equivalent to Lanczos regardless of the degree of diagonal dominence. More distressing is the fact that Davidson's method fails completely when A is a diagonal matrix since the new trial vector will just be y and the trial basis will become linearly dependent.

The best way to understand the behavior of Davidson's method is to analyze the operator $N(\theta) = (D - \theta)^{-1}(A - \theta)$. Each new trial vector is $N(\theta)$ times some vector in the space. If θ were constant, the trial space would just be the Krylov space generated by N. (Unfortunately since N is nonsymmetric in general, it would not be possible to use the symmetric Lanczos algorithm on it.) Of course θ is not constant but it does converge to the desired eigenvalue λ and so the properties of $N(\theta)$ for values of θ near λ are crucial to the behavior of the algorithm. $N(\theta)$ is just the operator obtained by applying diagonal scaling to $(A - \theta)$. It is well known, from studying diagonal scaling as a preconditioner for conjugate gradients, that this diagonal scaling will tend to compress the spectrum of $(A - \theta)$ closer to 1. For conjugate gradients, this is the goal in itself since the compressed spectrum will have a lower condition number and conjugate gradients will converge faster. For eigenvalue problems we are interested in how rapidly the Krylov subspace generated by N will contain good approximations to the desired eigenvector.

The dominant term in the convergence rate for Krylov subspaces depends on the gap ratio of the desired eigenvalue which measures the **relative** separation of the desired eigenvalue from the rest of the spectrum (see [7] p. 244). Convergence to interior eigenvalues is also possible but usually implies earlier convergence to **all** of the eigenvalues on one side of the desired eigenvalue. Compression of the spectrum by itself is not sufficient to insure rapid convergence. Two additional conditions must be met. First the desired eigenvalue (the smallest eigenvalue of $(A - \theta)$) must be moved less than the rest of the spectrum so that the gap ratio of the desired eigenvalue is increased. Furthermore the preconditioning must not greatly perturb the desired eigenvector so that convergence to the eigenvector of N implies convergence to the desired eigenvector of **A**.

If the desired eigenvector is a coordinate vector, e_i say, then e_i is also an eigenvector of $N(\theta)$ for all θ and the second condition is satisfied. Unfortunately the corresponding eigenvalue is exactly 1 which lies right in the middle of the spectrum of N and so convergence is **very** slow. In the special case of a diagonal matrix A, all the eigenvalues of N are 1 and the method breaks down. Table 4-1 shows the spectrum of N(1.0001) for Example 2. Note that the desired eigenvalue lies in the middle of the spectrum.

The situation is rather different in Example 3. For θ's near the desired eigenvalue, $(D - \theta)$ is nearly singular and $N(\theta)$ will have a very large singular value. After one matrix multiply a vector very close to e_{10} will be in the trial space. Essentially one step is wasted obtaining the approximation to e_{10} and then the process procedes as if the large singular value did not exist.

Table 4-1: Spectrum of N(1.0001) for Example 2

0.1683	0.6377	0.7689	0.8304
0.8661	0.8902	0.9128	0.9392
0.9688	0.9999	**1.0000**	1.0311
1.0608	1.0872	1.1098	1.1339
1.1696	1.2311	1.3623	1.8317

In the regular case, $(D - \theta)^{-1}$ remains bounded as θ approaches λ, the desired eigenvalue N becomes zero, and the corresponding eigenvector of N converges to the desired eigenvector of A. Thus the method does converge to the desired eigenvector with a better (and perhaps much better) convergence rate. In Example 1 the gap ratio of the desired eigenvalue (λ_1) for the original matrix A is

$$(\lambda_2 - \lambda_1)/(\lambda_{20} - \lambda_2) = (1.854160 - .222851)/(20.41594 - 1.866517)$$
$$= .087943$$

For different values of θ, Table 4.2 gives the smallest eigenvalue of N(θ), its gap ratio, and the sine of the angle between the corresponding eigenvector of N(θ) and the smallest eigenvector of A.

Table 4-2: Gap Ratios and Sines for Different Values of θ

θ	Eigenvalue of N	Gap Ratio	Sine
0.9	-2.16723	0.722	0.3776
0.5	-0.30592	0.499	0.1010
0.3	-0.06670	0.447	0.0246
0.22	0.00227	0.431	0.0009
0.2	0.01788	0.427	0.0069

It is clear from Table 4-2 that N is not very sensitive to changes in θ. The gap ratio for N for θ anywhere near λ_1 is five times the gap ratio for A. This makes an enormous difference in the convergence rate of the algorithm. Furthermore the eigenvector is hardly perturbed. Thus rapid local convergence is to be expected. The first value of θ encountered in Example 1 is 3.23529. This lies between the third eigenvalue (2.95594) and the fourth eigenvalue (3.99522) of A. Some of the eigenvalues of N(3.23529) are complex (which is not disastrous by itself) but the desired eigenvector is not well represented in the extreme eigenvalues of N. This explains why the first few Davidson steps in Example 1 were not as effective as the first few Lanczos steps. Davidson's method was happily trying to converge to the wrong eigenvector. It is only after θ is closer to the desired eigenvalue than to any of the others that the method starts converging rapidly. The gap ratio for the desired eigenvalue of N is at least a factor of five larger than the gap ratio for A for all values of θ near λ. Since the convergence rate depends exponentially on the gap ratio this makes an enormous difference in the convergence rate of the algorithm.

5. Generalizing Davidson's Method

Unfortunately Davidson's method does not always increase the gap ratio. If the diagonal of a matrix A is constant, then Davidson's method reduces to the Lanczos algorithm. If the diagonal is almost constant then Davidson's method may be slightly faster than Lanczos but it will probably not be worth the higher overhead. This was the conclusion of Kalamboukis [4] when he investigated using Davidson's method on nuclear modeling problems.

Interpreting the operator $(D - \theta_{j-1})^{-1}$ as a preconditioner for $(A - \theta_{j-1})$, the obvious way to improve Davidson's method is to use a better preconditioner. The generalized algorithm requires modification of

only step one of the original algorithm as follows:

$$1.\ \text{Compute } p_j = (M - \theta_{j-1})^{-1} r_{j-1},$$

where $M - \theta_{j-1}$ is some easily inverted approximation to $(A - \theta_{j-1})$. One potential advantage of this use of preconditioners over conjugate gradients is that here there is no requirement that the preconditioner be positive definite. This allows the preconditioner to more closely approximate the indefinite matrix $(A - \sigma)$. In what follows the generalized algorithm will be refered to as the GD algorithm.

Obviously the effectiveness of the GD algorithm depends on how well $(M - \theta)^{-1}$ approximates $(A - \theta)^{-1}$. The GD algorithm was applied to Example 1 using the preconditioner $(T - \theta)^{-1}$ where T is the tridiagonal part of A. Table 5-1 gives the sequence of eigenvalue approximations obtained by the GD algorithm (starting with the same vector).

Table 5-1: Behavior of the GD algorithm on Example 1

step	θ	residual norm
1	3.23529	5.274
2	2.58389	3.777
3	1.54362	1.286
4	1.49082	1.121
5	.38969	1.024
6	.22286	.0151
7	.22285	.1e-7
8	.22285	.6e-13

As can be seen from Table 5-1, the first four steps are very similar to the behavior of the original algorithm with convergence to the wrong eigenvalues. Once θ is closer to the desired eigenvalue than to the rest of the spectrum, convergence is almost immediate. As before the behavior of the algorithm can be understood by examining the spectrum of $N(\theta) = (T - \theta)^{-1}(A - \theta)$ for values of θ near the desired eigenvalue.

Table 5-2: Gap Ratios for Different Values of θ for GD

θ	Eigenvalue of N	Gap Ratio	Cosine
.9	1.0 + .171i	1.0	.4761
.2	0.2388	1.0	.0165
.222846097	.494e-7	1.0	.839e-6

As before, for all values of θ near the desired eigenvalue there is a well separated eigenvalue of $N(\theta)$ with a corresponding eigenvector which is nearly parallel to the desired eigenvector of A. The extremely rapid convergence obtained with the tridiagonal preconditioner is due to more than just the increased gap ratio. For all values of θ near the desired eigenvalue, the spectrum of $N(\theta)$ with tridiagonal preconditioning consists of a cluster of 18 eigenvalues within 1e-16 of 1 and two other eigenvalues symmetrically located around 1. This distribution is guaranteed to converge after only 3 steps.

6. Global Convergence

As seen above, both Davidson's Method and GD do not converge rapidly to the desired eigenvalue as long as θ is far away. If the desired eigenvalue is specified as the eigenvalue of A closest to a given number σ, then it is possible to modify the algorithm to improve the global convergence. Instead of using $(M - \theta)^{-1}$ as the preconditioner, $(M - \sigma)^{-1}$ can be used until θ has started converging to the desired eigenvalue. Tables 6-1, 6-2, and 6-3 give the behavior, for different values of σ, of both diagonal preconditioning and tridiagonal preconditioning on Example 1 when the preconditioners are fixed at σ until θ has settled down.

Table 6-1: $\sigma = .9$

	Diagonal		Tridiagonal	
step	θ	residual norm	θ	residual norm
1	3.2352	5.2740	3.2352	5.2740
2	.9007	1.3130	.5190	1.5320
3	.3321*	.5493	.2276*	.2001
4	.2347	.2184	.2229	.0331
5	.2236	.0613	.2228	.0002
6	.2229	.0130	.2228	.3813e-11
7	.2228	.0023		

*Switch from σ to θ

Table 6-2: $\sigma = .5$

	Diagonal		Tridiagonal	
step	θ	residual norm	θ	residual norm
1	3.2352	5.2740	3.2352	5.2740
2	.7455	1.1730	.2911	.9275
3	.3055	.4406	.2229*	.0168
4	.2318*	.1978	.2228	.0022
5	.2234	.0494	.2228	.1154e-5
6	.2229	.0117	.2228	.3836e-13

*Switch from σ to θ

As can be seen from the tables, this modification improves the global convergence of the algorithm and does not require a particularly accurate value for σ.

7. Conclusions

The success of Davidson's method on some types of eigenvalue problems shows the potential power of diagonal preconditioning. Generalizing Davidson's method allows for more powerful preconditioners to be used which makes the method effective for a much wider class of matrices. Using $(M - \sigma)^{-1}$ instead of $(M - \theta)^{-1}$ as the preconditioner shows that it is possible to force global convergence to a particular eigenvalue. If the formation of a matrix-vector product is very cheap then the overhead required in the GD algorithm will not be cost effective, but if the matrix-vector product is expensive then the GD algorithm will significantly reduce the number of matrix-vector products required and thus will be significantly cheaper

Table 6-3: $\sigma = .2$

	Diagonal		Tridiagonal	
step	θ	residual norm	θ	residual norm
1	3.2352	5.2740	3.2352	5.2740
2	.7054	1.1160	.2493	.7077
3	.2987	.4254	.2230	.0294
4	.2308	.1854	.2228*	.7790e-4
5	.2233	.0462	.2228	.2244e-7
6	.2228*	.0109	.2228	.4518e-13

*Switch from σ to θ

than the Lanczos algorithm.

References

[1] Axelsson, O.
On Preconditioning and Convergence Acceleration in Sparse Matrix Problems.
Technical Report CERN 74-10, CERN, Geneva, Switzerland, 1974.

[2] Davidson, E. R.
The Iterative Calculation of a Few of the Lowest Eigenvalues and Corresponding Eigenvectors of Large Real-Symmetric Matrices.
J. of Comp. Phys. 17:87-94, 1975.

[3] Gustafsson, I.
A Class of Fisrt Order Factorization Methods.
BIT 18:142-156, 1978.

[4] Kalamboukis, T. Z.
Davidson's Algorithm with and without Perturbation Corrections.
J. Phys. A: Math. Gen. 13:57-62, 1980.

[5] Meijerink, J. A., and H. A. van der Vorst.
An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix is a Symmetric M-matrix.
Math. of Comp. 31:148-162, 1977.

[6] Paige, C. C., and M. A. Saunders.
Solution of Sparse Indefinite Systems of Linear Equations.
SIAM J. Numer. Anal. 12:617-629, 1975.

[7] Parlett, B. N.
The Symmetric Eigenvalue Problem.
Prentice-Hall, 1980.

[8] Stone, H. L.
Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations.
SIAM J. of Numer. Anal. 5:530-558, 1968.

[9] Szyld, D. B.
A Two-level Iterative Method for Large Sparse Generalized Eigenvalue Calculations.
PhD thesis, Courant Institute, New York University, October, 1983.