LBNL-41382

INEXACT NEWTON PRECONDITIONING TECHNIQUES FOR LARGE SYMMETRIC EIGENVALUE PROBLEMS*

KESHENG WU†, YOUSEF SAAD‡, AND ANDREAS STATHOPOULOS§

Abstract. This paper studies a number of Newton methods and use them to define new secondary linear systems of equations for the Davidson eigenvalue method. The new secondary equations avoid some common pitfalls of the existing ones such as the correction equation and the Jacobi-Davidson preconditioning. We will also demonstrate that the new schemes can be used efficiently in test problems.

Key words. sparse matrix eigenvalue problem, Newton method, preconditioning for eigenvalue method

AMS subject classifications. 65F50, 65F15

1. Introduction.

This paper is concerned about improving the efficiency of computing a small number of eigenvalues of a large matrix A. The most common solution scheme is to first generate a basis for some subspace, and then apply the Rayleigh-Ritz projection on this basis to compute approximate solutions [6, 11, 27, 31]. The Lanczos method [21], the Arnoldi method [1] and the Davidson method [35] are examples of this type of procedures. In their simplest forms, e.g., using no preconditioning in the Davidson method, they generate Krylov subspace bases by repeatedly multiplying A with a vector. They are effective in computing the extreme and well separated eigenvalues. If the desired eigenvalues are not extreme or well separated, these methods may need to build very large Krylov subspaces to generate good approximate solutions.

Some of the most robust schemes used to reduce the basis of the Krylov subspace size are the spectrum transformation schemes [14, 18, 23, 29], for example, the shift-and-invert scheme. To compute eigenvalues near σ, the shift-and-invert scheme computes the extreme eigenvalues of $(A - \sigma I)^{-1}$ instead. If the wanted eigenvalues are close to σ, the extreme eigenvalues of $(A-\sigma I)^{-1}$ are well separated and can be easily found by the Lanczos method or the Arnoldi method. When using the shift-and-invert scheme, we need to multiply $(A - \sigma I)^{-1}$ with a sequence of vectors. This operation is usually implemented as solving a series of linear systems involving the matrix $(A-\sigma I)$. The shift-and-invert scheme requires accurate solutions to these linear systems. Since accurate solutions are often hard to compute, it is natural to consider a modfied scheme which uses approximate solutions.

There are many ways of using such approximate solutions. When the solutions are fairly accurate, one can simply use them as if they are accurate. Golub et al.[17] have studied this case for symmetric matrices. When the solutions are less accurate or the accuracy cannot be easily controlled, one must explicitly cope with the arbitrariness in the solutions. The Davidson method deals with this by allowing any approximate solution to be used [8]. For convenience of discussion, we call the linear systems to be solved in the eigenvalue calculations *the secondary equations*, and the matrices of the secondary equations *the secondary matrices*. Typically, the secondary equations are approximately solved using one of the preconditioning techniques for solving large linear system, such as incomplete LU (ILU) factorizations

*Work supported by National Science Foundation under contract No. NSF/DMR-95 25885, NSF/ASC-95 04038, by the Minnesota Supercomputer Institute, and in part by the Director, Office of Energy Research, Office of Laboratory Policy and Infrastructure Management, of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098.

†Lawrence Berkeley National Laboratory, Berkeley, CA. Email: kwu@lbl.gov.

‡Computer Science Department, University of Minnesota, Minneapolis, MN. Email: saad@cs.umn.edu.

§Computer Science Department, College of William and Mary, Williamsburg, VA. Email: andreas@cs.wm.edu.

[24], approximate inverse schemes [5, 7, 20], and others preconditioners [4, 2, 32]. For this reason, utilizing a secondary equation while solving an eigenvalue problem is sometimes called preconditioning. In many tests, the Davidson method is found to be very effective in taking advantage of the approximate solutions [40]. Other eigenvalue methods that utilize secondary equations include the approximate rational Krylov method [22] and the Jacobi-Davidson method [33]. Many of these methods can be viewed as the Davidson method with different matrices or right-hand sides in the secondary equations. For this reason, we only consider the new secondary equations for the Davidson method.

When Davidson first proposed his method, he proposed $\text{diag}(A - \lambda I)z = r$ as the secondary equation [9], where λ is the current Ritz value, x is the corresponding Ritz vector, and r is the residual vector $r \equiv (A - \lambda I)x$. This equation is the result of applying the diagonal preconditioning technique on the correction equation,

$$(A - \lambda I)z = r, \tag{1.1}$$

This correction equation was designed for enhancing the accuracy of eigenvalue solutions [12]. It is known to have a number of pitfalls [12]. For example, when λ is close to an exact eigenvalue, the matrix $A - \lambda I$ is ill-conditioned and possibly indefinite. For such matrices, most of the inexact solution techniques, e.g., ILU, are not reliable. Another pitfall is that if the correction equation is solved exactly, the solution is x. Since x is contained in the current basis, the Davidson method stagnates. Many variations of the Davidson method can mitigate some of these pitfalls. For example, both Olsen's preconditioning scheme [26] and the Jacobi-Davidson method [33] avoid stagnation by generating a z that is orthogonal to x. However, there is no analysis indicating the linear systems involved in these schemes are easier to solve than the correction equation. When computing extreme eigenvalues of symmetric matrices, a biased shift $\bar{\lambda}$ can be used instead of λ [36]. Using this biased shift, the secondary matrix is more likely to be definite. However, this biased shift is only defined for symmetric matrices and the matrix $A - \bar{\lambda} I$ may still be ill-conditioned near convergence.

The objective of this paper is to seek well-conditioned secondary equations that can avoid stagnation. Since the correction equation is a Newton method for eigenvalue problems, we seek other forms of Newton methods and study their suitability as secondary equations. A number of different Newton schemes are reported in Section 2. Most of them give rise to linear systems with desired properties. To demonstrate how the new schemes may be used, a small set of tests is presented in Section 3. The tests also demonstrate the effectiveness of various secondary equations. We conclude with a summary that recounts the advantages and disadvantages of the various methods.

2. Newton methods for eigenvalue problems.

In this section we describe a number of Newton methods for eigenvalue problems. We show that these Newton methods are well defined. Since well defined Newton methods will converge quadratically, if the Davidson method solves the linear systems from these Newton methods accurately, it will also converge quadratically. Naturally, there is no stagnation problem. We will also study under what conditions the Jacobian matrices of these Newton methods become singular. This will help us identify when not to use them. We start by discussing how Equation (1.1) can be viewed as a Newton method for eigenvalue problem in subsection (2.1). The main body of this section (2.2, 2.3) concentrates on a set of constrained Newton methods. We close this section by showing a concrete example of how various Newton methods work.

To simplify the discussion, we only consider real symmetric matrices. It is straightforward to extend the results to the nonsymmetric case.

2.1. A second look at the existing schemes.

The Davidson method was proposed as a combination of the Lanczos method and the Newton method for minimizing the Rayleigh quotient [9]. In fact, the first two derivatives of the Rayleigh quotient, $\lambda = x^T Ax/x^T x$, are[1]

$$\frac{\mathrm{d}\lambda}{\mathrm{d}x} = \frac{2Ax}{x^T x} - \frac{2x^T Axx}{(x^T x)^2}$$
$$\frac{\mathrm{d}^2\lambda}{\mathrm{d}x^2} = \frac{2A}{x^T x} - \frac{2(2Axx^T + x^T AxI + 2xx^T A)}{(x^T x)^2} + \frac{8x^T Ax\, xx^T}{(x^T x)^3}.$$

The linear system defined by the Newton method is

$$\frac{d^2\lambda}{dx^2} z = \frac{d\lambda}{dx}. \tag{2.1}$$

Had Davidson not made any further simplification, this equation would be used in his eigenvalue method. Since the Jacobian matrix is singular and fairly complicated, Davidson proposed to use the diagonal part of $A-\lambda I$, i.e., 2diag$(A-\lambda I)$, instead. For the quantum chemistry problems that he was interested in, this simple scheme works well because the matrices are diagonally dominant. Many researchers consider the Jacobian matrix to be $2(A - \lambda I)$ [8, 10, 25], because the resulting linear system is the correction equation (see Equation (1.1)) [12].

The correction equation is regarded as the linear system from the Newton method that computes an eigenvector when the eigenvalue is known. For symmetric matrices the eigenvalue usually converges faster than the eigenvector, it is reasonable to consider λ as "accurate". The Newton method to solve equation $(A - \lambda I)x = 0$ with an initial guess x_0 is

$$x_{i+1} = x_i - (A - \lambda I)^{-1} r_i, \tag{2.2}$$

where $r_i = (A - \lambda I)x_i$. It is clear that this is not a well defined Newton method. However, it can be corrected if λ is the exact eigenvalue λ^* and we replace the usual inverse with pseudoinverse [16],

$$x_{i+1} = x_i - (A - \lambda^* I)^+ r_i. \tag{2.3}$$

By definition, $AA^+A = A$, and thus the change in x_i still satisfies the correction equation:

$$\begin{aligned}(A - \lambda^* I)(x_i - x_{i+1}) &= (A - \lambda^* I)(A - \lambda^* I)^+ r_i \\ &= (A - \lambda^* I)(A - \lambda^* I)^+(A - \lambda^* I)x_i \\ &= (A - \lambda^* I)x_i \equiv r_i.\end{aligned}$$

If the initial guess x_0 is not orthogonal to the exact eigenvector x^*, then x_1 computed by Equation (2.3) is a non-trivial solution of $r = 0$. In other words, x_1 is the desired eigenvector.

The Jacobi-Davidson method uses the following secondary equation to replace the correction equation,

$$(I - xx^T)(A - \lambda I)(I - xx^T)z = r. \tag{2.4}$$

Since the matrix is singular, we would need to use the pseudoinverse as well. Usually, a Krylov subspace based iterative method is used to solve the above equation. Since the solutions generated by the the Krylov methods are in the range of the matrix, they are approximations to the pseudoinverse solutions. The Jacobi-Davidson method is related to a Newton method as well [34].

[1]Davidson did not show these equations in his paper.

In short, we have briefly described three Newton methods for eigenvalue problems, the Newton method for minimizing Rayleigh quotient, the Newton method to compute an eigenvector when an eigenvalue is known, and the Newton method associated with the Jacobi-Davidson method. Among the three Jacobian matrices, two are singular by construction, and the other can become ill-conditioned near convergence. The Krylov subspace methods can be used to approximate pseudoinverse solutions for these problems. To effectively use other preconditioning techniques such as ILU, a well-conditioned secondary equation is needed. Next we present some Newton recurrences with well-conditioned linear systems.

2.2. Constrained Newton Recurrence. The eigenvectors of A are the solutions to the following nonlinear equation with a norm constraint,

$$Ax - xx^T Ax = 0, \qquad \|x\| = 1. \tag{2.5}$$

In this form, Tapia's algorithm for constrained optimization [38] can be directly applied after the Jacobian matrix is evaluated. By denoting $\lambda \equiv x^T Ax$, the Jacobian matrix can be written as

$$J_C \equiv A - \lambda I - xx^T(A + A^T). \tag{2.6}$$

Given an initial guess x_0, the constrained Newton recurrence to solve Equation (2.5) is,

$$J_C z = \quad r_i, \qquad (r_i = Ax_i - \lambda_i x_i, \quad \lambda_i = x_i^T Ax_i), \tag{2.7}$$

$$x_{i+1} = \frac{x_i - z}{\|x_i - z\|}, \qquad \lambda_{i+1} = x_{i+1}^T Ax_{i+1}. \tag{2.8}$$

It is easy to show that the following is true.

LEMMA 2.1. *If λ^* is a nonzero simple eigenvalue of symmetric matrix A and x^* is the corresponding eigenvector, then $J_C \equiv A - \lambda^* I - 2x^* x^{*T} A$ is nonsingular*[2].

The detailed proof of this lemma can be found elsewhere [40, Lemma 3.5]. The key is to show that only a zero vector y can satisfy $J_C y = 0$. The proof is similar to that used by Peters and Wilkinson on the augmented Newton recurrence [28].

Since J_C is a polynomial of x, J_C is nonsingular when x is sufficiently close to an eigenvector. Based on a theorem of Tapia [38, Theorem 3.3], the above Newton recurrence should converge quadratically near a nonzero simple eigenvalue. In addition, it is also easy to show that this Newton recurrence is mathematically equivalent to the Rayleigh Quotient iteration [40, Lemma 3.6],

$$x_{i+1} = \frac{(A - \lambda_i I)^{-1} x_i}{\|(A - \lambda_i I)^{-1} x_i\|}, \qquad (\lambda_i = x_i^T Ax_i).$$

LEMMA 2.2. *If at step i the constrained Newton scheme and the Rayleigh quotient iteration have the same solutions (λ_i, x_i), λ_i is not zero or an exact eigenvalue and J_C is not singular, the solutions at the next step will also be the same.*

To prove this lemma we need to show that $x_i - z$ (see Equations (2.7) and (2.8)) is parallel to $(A - \lambda_i I)^{-1} x_i$. Because these two vectors are scaled to unit length before assigned to x_{i+1}, if they are nonzero and parallel to each other, then x_{i+1} produced by the two methods must be the same. The detailed proof can be found elsewhere [40, Lemma 3.6].

Although near convergence J_C is well behaved, far away from convergence it is possible that it may become singular. The following lemma states the condition under which this may be true.

[2] If A is not symmetric, the real part of λ^* must not be zero.

LEMMA 2.3. *Given an arbitrary unit vector x, let $(\lambda = x^T Ax)$ be different from any eigenvalue of A, the matrix $J_C \equiv A - \lambda I - 2xx^T A$ is singular if and only if the following is true*[3],

$$x^T A(A - \lambda I)^{-1} x = 1/2. \tag{2.9}$$

Proof. If J_C is singular, there exists a nonzero vector y such that $J_C y = 0$. This equation can be rewritten as

$$(A - \lambda I)y = 2xx^T Ay.$$

The vector y satisfying this equation must be of the form $y = \zeta(A - \lambda I)^{-1}x$ where ζ is a nonzero constant. Substituting this expression of y into the above equation yields Equation (2.9).

If Equation (2.9) is satisfied, the vector $y = (A - \lambda I)^{-1}x$ can satisfy $J_C y = 0$. The vector y is not zero because Equation (2.9) can be written as $x^T Ay = 1/2$. The matrix J_C is singular. □

From these three lemmas, we see that the constrained Newton recurrence defined by Equations (2.7) and (2.8) is well behaved for almost any x. Starting with any x that does not satisfy Equation (2.9), Equation (2.7) will generate a solution that is not parallel to x. When used to extend the basis, the Davidson method should make progress. The lemmas do not give a formula for the condition number of J_C. Later in this paper we give a numerical example that is representative of the relative sizes of the condition numbers from the different schemes.

2.3. Augmented Newton Recurrence. Another formulation of the eigenvalue problem is to treat it as an $(n+1)$-dimensional unconstrained optimization problem. A Newton method can be applied to solve this problem [28]. Similar formulations of the eigenvalue problem have been given in the past, using a variety of normalization schemes. We choose to normalize the eigenvectors using the 2-norm. The eigenvalue problem can be restated as

$$\left\{ \begin{array}{rcl} (A - \lambda I)x & = & 0, \\ -\frac{1}{2}x^T x + \frac{1}{2} & = & 0. \end{array} \right. \tag{2.10}$$

This is an unconstrained quadratic problem. Given an initial guess (x_0, λ_0), the Newton recurrence is

$$\begin{pmatrix} x_{i+1} \\ \lambda_{i+1} \end{pmatrix} = \begin{pmatrix} x_i \\ \lambda_i \end{pmatrix} - J_A^{-1} \begin{pmatrix} (A - \lambda_i I)x_i \\ -\frac{1}{2}x_i^T x_i + \frac{1}{2} \end{pmatrix}, \tag{2.11}$$

$$J_A = \begin{pmatrix} A - \lambda_i I & -x_i \\ -x_i^T & 0 \end{pmatrix}. \tag{2.12}$$

LEMMA 2.4. *If λ^* is a simple eigenvalue of A, and x^* is the corresponding eigenvector,* $J_A \equiv \begin{pmatrix} A - \lambda^* I & -x^* \\ -(x^*)^T & 0 \end{pmatrix}$ *is nonsingular.*

Although we have formulated this augmented Newton recurrence differently, the proof of the above lemma follows closely what is used in an earlier formulation [28]. Because of continuity, when λ_i is close to a simple eigenvalue, J_A is nonsingular as well.

[3]If the matrix is nonsymmetric, the condition changes to $x^H(A + A^H)(A - \lambda I)^{-1}x = 1$.

In the Davidson method, where (λ_i, x_i) is a Ritz pair and x_i is normalized, the secondary equation derived from the augmented Newton recurrence is

$$\begin{pmatrix} A-\lambda_i I & -x_i \\ -x_i^T & 0 \end{pmatrix} \begin{pmatrix} z \\ \delta \end{pmatrix} = \begin{pmatrix} Ax_i - \lambda_i x_i \\ 0 \end{pmatrix}. \tag{2.13}$$

We use z to expand the basis and discard δ.

Based on this observation, we modify the augmented Newton recurrence into a constrained Newton recurrence; each time Equation (2.11) is solved, we scale the resulting x_{i+1} to have norm one and replace λ_{i+1} with the Rayleigh quotient (see Equation (2.8)). We call this modified scheme the normalized augmented Newton recurrence.

The matrices involved in Equations (2.11) and (2.13) are larger than A. If we prefer to work with a matrix that have size as A, we may factor Equation (2.13) symbolically and rewrite z as

$$z = (A-\lambda_i I)^{-1} \left(I - \frac{x_i x_i^T (A-\lambda_i I)^{-1}}{x_i^T (A-\lambda_i I)^{-1} x_i} \right) r_i. \tag{2.14}$$

In the Olsen scheme [26], $(A-\lambda_i I)^{-1}$ of the above equation is replaced with an approximation. In fact, we can replace $(A-\lambda_i I)^{-1}$ with any matrix M as long as $x_i^T M x_i$ is nonzero, the resulting z is orthogonal to x_i. Using Equation (2.14), it can be shown that $x_i - z$ is parallel to $(A-\lambda_i I)^{-1} x_i$; therefore, the normalized augmented Newton recurrence is equivalent to the Rayleigh quotient iteration [40, Lemma 3.2]:

LEMMA 2.5. *If the Rayleigh quotient iteration and the normalized augmented Newton iteration produce the same solutions at step i, $A-\lambda_i I$ is not singular and $x_i^T (A-\lambda_i I)^{-1} x_i$ is not zero, then the two methods produce the same solution at step $i+1$.*

Using the secondary matrix of the Jacobi-Davidson method, we can also define z as follows

$$z = \left((I - x_i x_i^T)(A-\lambda_i I)(I - x_i x_i^T) \right)^{+} r_i. \tag{2.15}$$

It is easy to see that Equation (2.14) and Equation (2.15) produce the same results. The recurrence formed from Equations (2.15) and (2.8) is equivalent to the Rayleigh quotient iteration [40].

The normalized augmented Newton recurrence is well defined for almost any vector x except under the conditioned established in the following lemma.

LEMMA 2.6. *If λ is not an eigenvalue of A, J_A is singular if and only if*

$$x^T (A-\lambda I)^{-1} x = 0, \qquad (\lambda = x^T A x). \tag{2.16}$$

Proof. If J_A is singular, there is a nonzero vector $\binom{y}{\zeta}$ such that

$$\begin{pmatrix} A-\lambda I & -x \\ -x^T & 0 \end{pmatrix} \begin{pmatrix} y \\ \zeta \end{pmatrix} = 0.$$

Since x is not an eigenvector, $A-\lambda I$ is not singular. The above equation leads to $y = \zeta (A-\lambda I)^{-1} x$ and $x^T y = 0$. In order for y to be a nonzero vector, Equation (2.16) must be true. If Equation (2.16) is true, the null space of J_A contains vector $\binom{(A-\lambda I)^{-1} x}{1}$; therefore J_A is singular. ☐

By varying the augmented Newton recurrence, we have generated a normalized augmented Newton recurrence and shown that the new variant is equivalent to the Rayleigh

TABLE 2.1
The number of steps and the final Ritz values computed by different Newton recurrences.

	Iterations	λ	r
Rayleigh quotient iteration	9	0.2094224436	8.26e-09
Constrained Newton recurrence	9	0.2094224436	2.97e-09
Normalized Augmented Newton	9	0.2094224435	3.16e-09
Inflated Newton recurrence	9	0.2094224435	2.28e-08
Augmented Newton recurrence	14	0.0680435715	4.58e-09

quotient iteration. In addition, the normalized augmented Newton recurrence is also equivalent to the "exact" form of the Olsen preconditioning scheme and the Newton recurrence in Jacobi-Davidson method. If we compute the accurate solution z from Equations (2.13), (2.14) and (2.15), it should not cause the Davidson method to stagnate. All above variations of the Newton method are derived with rigorous mathematical reasoning. On the other hand, the next scheme is generated based on a heuristic.

We observe that the matrix in front of r_i in Equation (2.14) closely resembles the Sherman-Morrison-Woodbury formula for the inverse of $A - \lambda_i I + \alpha x_i x_i^T$ where α is an arbitrary constant [16, Equation (2.1.4)],

$$(A - \lambda_i I + \alpha x_i x_i^T)^{-1} = (A - \lambda_i I)^{-1} \left(I - \frac{\alpha x_i x_i^T (A - \lambda_i I)^{-1}}{1 + \alpha x_i^T (A - \lambda_i I)^{-1} x_i} \right).$$

Based on this observation, we can define yet another recurrence by combining the following equation and Equation (2.8),

$$z = J_I^{-1} r_i, \qquad J_I = A - \lambda_i I + \alpha x_i x_i^T. \tag{2.17}$$

The same iteration matrix J_I has been used before in a so-called Inflated Inverse Iteration [15]. Thus we refer to this new recurrence as the Inflated Newton Recurrence. This recurrence is well defined since it is always possible to choose a value for α to make J_I nonsingular. It is easy to verify that the eigenpairs are the stationary points of the recurrence.

2.4. An example. We will end this section by giving one example. Among the various Newton schemes, only the Inflated Newton Recurrence is defined procedurally and does not optimize a known quantity. This example will show that it actually behaves like a Newton method. During the Newton recurrences, we also computed the condition numbers since the theoretical analysis does not show exactly how large they are.

The test matrix we choose is called EX2. It is generated from solving a fully coupled Navier-Stokes equation using the FIDAP package and is available from MatrixMarket[4]. It is a symmetric 441×441 matrix with only simple eigenvalues ranging from -7×10^8 to 3×10^6. There are 28 well-separated negative eigenvalues and 160 eigenvalues between zero and one. The condition number of the matrix is roughly 10^{10}. The initial guesses used in all tests are $[1, 1, \ldots, 1]^T$. The tests are carried out using Matlab and the linear systems are solved with the matlab operator "\".

The Newton recurrences was run until the residual norms do not further decrease. The numbers of iterations taken, the final Ritz values and the residual norms are recorded in Table 2.1. The Ritz values from the first four methods listed in the table agree with each other to better accuracy than $\epsilon \|A\| \sim 7 \times 10^{-8}$. The augmented Newton recurrence converges to a

[4]MatrixMarket URL: `http://math.nist.gov/MatrixMarket/`.

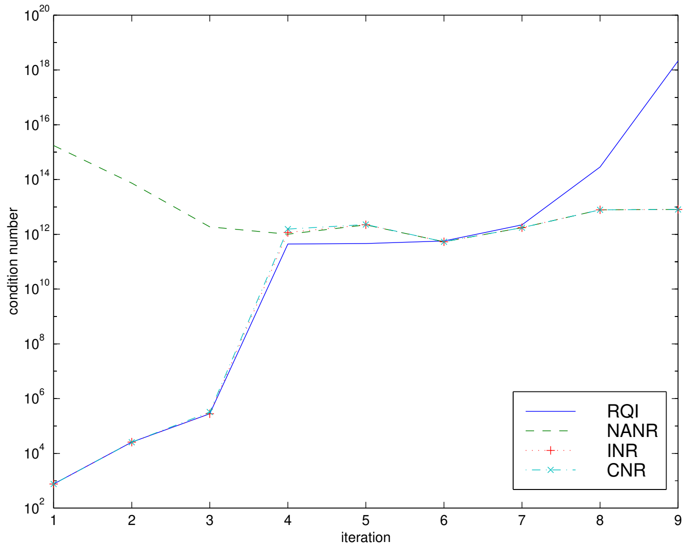

FIG. 2.1. *Condition numbers of the Jacobian matrices from different Newton recurrences.*

different eigenvalue. The final residual norms from all methods can be regarded as negligible because they are less than $\epsilon\|A\|$.

In theory, the Rayleigh quotient iteration, the constrained Newton method and the normalized augmented Newton method are equivalent to each other. Indeed they produce the same results at every step. Surprisingly, the inflated Newton method generates almost the same results as well. This is another reason we regard it as a Newton method.

Figure 2.1 shows the condition numbers of the Jacobian matrices used in the four recurrences, namely, the Rayleigh quotient iteration (RQI), the normalized augmented Newton recurrence (NANR), the inflated Newton recurrence (INR), and the constrained Newton recurrence (CNR). The condition numbers become large in two cases: near convergence for the Rayleigh quotient iteration and at the start of the normalized augmented Newton recurrence. In most other cases, the condition numbers of the iteration matrices are less than 10^{12} which is 100 times the condition number of EX2.

The motivation of this paper is to find well conditioned secondary matrices for the Davidson method. The matrix J_I seems to be well conditioned in both theory and in this example. In theory, both J_C and J_A are nonsingular near convergence, this example provides a reference point as how large their condition numbers actually are. It is possible for both J_C and J_A to become singular when the solutions are far away from any eigenpair. The example shows that the vector $[1, 1, \ldots, 1]$ can make J_A very close to singular.

3. Suitability as secondary equations. In the previous section, we have identified a number of well-behaved Newton methods for eigenvalue problems. In this section we will explain and show how the linear systems from these Newton methods may be used as secondary equations for the Davidson method. Equations (1.1), (2.7), (2.13), (2.14), (2.15) and (2.17) each describe a scheme to compute z that can be used to extend the basis in the Davidson method. Among the six, Equations (2.7), (2.13) and (2.17) have not been studied in this context before. Typically, the secondary equations are approximately solved using preconditioning techniques developed for solving linear systems. Our attempt to find well-conditioned secondary equations can be regarded as an attempt to find more efficient ways of

TABLE 3.1
The test matrices from HB collection.

NAME	N	NNZ	Description
BCSSTK09	1083	9760	stiffness matrix, square plate clamped
1138BUS	1138	2596	admittance matrix, power system
BCSSTM13	2003	11973	mass matrix, fluid flow
ZENIOS	2873	15032	air-traffic control model

utilizing these techniques for computing eigenvalues. Here are some possible choices.

- **Incomplete factorizations of J_A**. It is not hard to see how one might modify an ILU routine for A to compute an ILU factorization for J_A. One caveat is that the last row and the last column of J_A probably need special attention.
- **Approximate inverses**. Approximate inverses of J_A, J_C and J_I may be computed relatively easily [7, 19, 20], especially if the approximate inverse can be computed using only matrix-vector multiplications with the original matrix [7].
- **Iterative solvers**. An iterative method can be used to solve the secondary equations as is usually done in the Jacobi-Davidson method. In the context of linear system solution, there are many schemes of using inner-outer iterations [3, 30, 39]. These studies show many schemes to control the accuracy and the maximum number of iterations of the inner iterations. Generally, this is a robust methodology. If there is only a matrix-vector multiplication routine available, the Davidson method with an iterative linear system solver can usually find the desired solutions in less time than the one without the iterative solver.

In our numerical tests we use iterative solvers because this scheme is fairly easy to implement, yet it is quite effective. We only implemented a very simple version of inner-outer iteration scheme. Each time the solver is invoked it iterates until the residual norm is decreased by a factor of 10^{-4}, or until a maximum of 200 matrix-vector multiplications is reached. The tests attempt to compute five smallest eigenvalues of a selected set of symmetric matrices from the Harwell-Boeing collection [13] (see Table 3.1). The Davidson method with thick-restart [37] is used for the tests. The solutions are declared converged if the residual norms are less than $10^{-12}\|A\|_F$. The total number of matrix-vector multiplications allowed to both the Davidson method itself and the iterative linear system solvers is 300,000. Like many inner-outer iteration schemes, this limit is fairly high. However, if the matrix-vector multiplication is indeed the only way to access the matrix, this limit may be reasonable.

We want to use the best available variations of the Davidson method so that the comparisons are relevant to the actual use. For this reason, we used the biased shift $\bar{\lambda}$ [36] instead of λ in the secondary matrices. With this modification, the correction equation behaves like the Rayleigh quotient iteration in the Davidson method [36]. In this case, if the secondary equations are solved accurately, the four schemes to be tested, the correction equation, the inflated Newton method, the constrained Newton method and the Jacobi-Davidson method, should generate the same solutions. Since we are compute the smallest eigenvalue, the biased shift also make the matrix $A - \bar{\lambda}I$ almost always positive definite. This makes the iterative linear system solvers more effective.

We have applied CG, BiCG and GMRES(10) to solve four different secondary equations, Equations (1.1), (2.4), (2.7) and (2.17). The results reported here only contain those from using CG, because it minimizes the time used on the test problems. Part of the reason why CG uses less time is that it uses less arithmetic operations per step. The matrix J_C is not symmetric; however, this did not cause CG to break-down. In most cases, the solutions

TABLE 3.2

The total number of matrix-vector multiplications used by the Davidson method with CG preconditioner.

	BCSSTK09	1138BUS	BCSSTM13	ZENIOS
$A - \bar{\lambda}I$	209514	12685	9180	9396
$A - \bar{\lambda}I + xx^T$	123154	12683	8374	7489
$A - \bar{\lambda}I - 2x(Ax)^T$	291153	13347	12000	2051
$(I - xx^T)(A - \bar{\lambda}I)(I - xx^T)$	35971	13000	10200	>324000

TABLE 3.3

Time (seconds on a SGI Challenge) used by the Davidson method with CG preconditioner.

	BCSSTK09	1138BUS	BCSSTM13	ZENIOS
$A - \bar{\lambda}I$	163	5.2	10.0	14.7
$A - \bar{\lambda}I + xx^T$	99	5.6	9.9	12.5
$A - \bar{\lambda}I - 2x(Ax)^T$	236	6.0	14.1	3.6
$(I - xx^T)(A - \bar{\lambda}I)(I - xx^T)$	31	6.5	12.8	>575

computed by the iterative methods are not accurate. What appears to be important is that the solutions contain significant new components outside of the space already explored by the Davidson method. In this regard, CG appears to be just as effective as BiCG and GMRES(10).

Table 3.2 and 3.3 summarizes matrix-vector multiplication and time used to compute the five smallest eigenvalues of the four test problems. The matrix-vector multiplications (MATVEC) reported include those used by the Davidson method itself and the iterative linear system solvers. The timing results are obtained on a SGI Challenge. The four problems are selected to accentuate each of the four schemes. In terms of reducing the total CPU time used by the Davidson method, solving the correction equation in the Davidson method is most effective for the 1138BUS test problem; solving the linear system from the inflated Newton recurrence is most effective for BCSSTM13; solving the linear system from the constrained Newton recurrence is most effective for ZENIOS and the Jacobi-Davidson method is most effective for BCSSTK09.

There are many different ways of controlling the inner-outer iteration schemes, we have only tested a very simple scheme. The exact time used by different secondary equations may change if a different inner-outer iteration scheme is used. However, there is no indication that the relative performance of the four secondary equations tested here will change significantly when a different inner-outer iteration scheme. These four test cases can be regarded as representative cases for our comparison. Overall, none of the secondary equations is consistently better than others. As a rule of thumb, the first two secondary matrices, $A - \bar{\lambda}I$ and $A - \bar{\lambda}I + xx^T$, use roughly about the same number of matrix-vector multiplications and time. The other two secondary matrices, $A - \bar{\lambda}I - 2x(Ax)^T$ and $(I - xx^T)(A - \bar{\lambda}I)(I - xx^T)$, behave rather differently in some cases. For example, for BCSSTK09, the Jacobi-Davidson method is effective but the constrained Newton scheme used about 8 times the matrix-vector multiplication and time. For ZENIOS, the constrained Newton method uses the least amount of time and the Jacobi-Davidson scheme fails to compute five smallest eigenvalues with 300,000 matrix-vector multiplications.

Since none of the secondary equations tested is superior than others on all test problems, it might be reasonable to choose the one that is least time-consuming to apply. In our implementation of the iterative methods, the matrix-vector multiplication routines for $A - \bar{\lambda}I + xx^T$, $A - \bar{\lambda}I - 2x(Ax)^T$ and $(I - xx^T)(A - \bar{\lambda}I)(I - xx^T)$ are built on top of a matrix-vector multiplication routine for $A - \bar{\lambda}I$. Multiplying with $A - \bar{\lambda}I + xx^T$ and $A - \bar{\lambda}I - 2x(Ax)^T$

each need one additional dot-product and one additional SAXPY operations. Multiplying $(I - xx^T)(A - \bar{\lambda}I)(I - xx^T)$ needs two additional dot-product and two additional SAXPY operations. The extra arithmetic operations may not be significant. However, on distributed computing environments, the extra synchronization caused by computing the dot-products is usually not negligible. Under this circumstance using the Krylov methods to solve the correction equation should be the first choice.

4. Summary. We have studied a number of Newton methods for eigenvalue problems and tested some of them as secondary equations in the Davidson method. The theoretical advantages of the new secondary equations are that they do not become singular near convergence and solving these linear systems accurately will lead to quadratic convergence.

In theory, most of the Newton methods described here are equivalent to each other. If we solve the corresponding secondary equations accurately, the Davidson method should generate the same solution no matter which one is used. However, when used as secondary equations, we do not solve them accurately. Because the secondary matrices are different, the approximate solutions are different as well. When the secondary matrix has smaller condition number, it is usually easier to generate more accurate solutions which in turn will cause the Davidson method to converge faster. In most cases, the Ritz values computed by the Davidson method are not extremely close to any eigenvalue. For example, in many engineering applications, 3 – 6 decimal point accuracy is sufficient. In Figure 2.1, this corresponds to stopping the recurrences after 3 or 4 iterations. During the first few iterations, the condition numbers of $A - \lambda I$ and $A - \lambda I + xx^T$ are relatively small. Most approximation schemes can be used to solve the corresponding linear systems reasonably well. In short, the pitfall that $A - \lambda I$ becomes ill-conditioned near convergence is not a serious problem in practice. This partly justifies why using CG to solve the correction equation with biased shift is competitive against three other Newton schemes. Because the matrix $A - \bar{\lambda}I$ is often cheaper to multiply, we suggest that the correction equation should be the first choice as the secondary equation for the Davidson method.

Though the theoretical advantages of the new schemes did not turn out to be the determining factor in the actual performance of the Davidson method, we did find some cases where the Davidson method with the constrained Newton scheme and the inflated Newton scheme use less time than with the correction equation and the Jacobi-Davidson preconditioning. In fact, on one of test problems, the constrained Newton scheme was considerably more effective than others schemes. For this reason, the new schemes are worthwhile alternatives if the existing schemes are not very successful on some particular problems.

REFERENCES

[1] W. E. Arnoldi. The principle of minimized iteration in the solution of the matrix eigenvalue problem. *Quarterly of Applied Mathematics*, 9:17–29, 1951.

[2] O. Axelsson and P. S. Vassilevski. A survey of multilevel preconditioned iterative methods. *BIT*, 29(4):769–793, 1989.

[3] O. Axelsson and P. S. Vassilevski. A block generalized Conjugate Gradient solver with inner iterations and variable step preconditioning. *SIAM J. Matrix Anal. Appl.*, 12:625–644, 1991.

[4] Owe Axelsson. A survey of preconditioned iterative methods for linear systems of equations. *BIT*, 25:166–187, 1985.

[5] M. Benzi, C. D. Meyer, and M. Tuma. A sparse approximate inverse preconditioner for the conjugate gradient method. *SIAM J. Sci. Comput.*, 17(5):1135–1149, 1996.

[6] F. Chatelin. *Eigenvalues of Matrices*. Wiley, New York, NY, 1993.

[7] E. Chow and Y. Saad. Approximate inverse preconditioners via sparse-sparse iterations. *SIAM J. Sci. Comput.*, 18(6):1657–1675, 1997.

[8] M. Crouzeix, B. Philippe, and M. Sadkane. The Davidson method. *SIAM J. Sci. Comput.*, 15:62–76, 1994.

[9] Ernest R. Davidson. The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices. *J. Comput. Phys.*, 17:87–94, 1975.

[10] Ernest R. Davidson. Super-matrix methods. *Computer Physics Communications*, 53:49–60, 1989.

[11] James W. Demmel. *Applied Numerical Linear Algebra*. SIAM, Philadelphia, PA, 1997.

[12] J. J. Dongarra, C. B. Moler, and J. H. Wilkinson. Improving the accuracy of computed eigenvalues and eigenvectors. *SIAM J. Numer. Anal.*, 20:23–45, 1983.

[13] I. S. Duff, R. G. Grimes, and J. G. Lewis. Sparse matrix test problems. *ACM Trans. Math. Soft.*, 15:1–14, 1989.

[14] T. Ericsson and A. Ruhe. The spectral transformation Lanczos method for numerical solution of large sparse generalized symmetric eigenvalue problems. *Math. Comp.*, 35:1251–1268, 1980.

[15] Albert T. Galick. *Efficient solution of large sparse eigenvalue problems in microelectronic simulation*. PhD thesis, University of Illinois at Urbana-Champaign, 1993.

[16] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD 21211, third edition, 1996.

[17] Gene H. Golub, Zhenyue Zhang, and Hongyuan Zha. Large sparse symmetric eigenvalue problems with homogeneous linear constraints: The lanczos process with inner-outer iterations. Technical Report CSE-96-050, Department of Computer Science and Engineering, The Pennsylvania State University, 1996. Submitted to SIAM Journal of Numerical Analysis.

[18] R. G. Grimes, J. G. Lewis, and H. D. Simon. A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems. *SIAM J. Matrix Anal. Appl.*, 15(1):228–272, 1994.

[19] M. J. Grote and T. Huckle. Parallel preconditioning with sparse approximate inverses. *SIAM J. Sci. Comput.*, 18(3):838–853, 1997.

[20] L. Yu. Kolotilina and A. Yu. Yeremin. Factorized sparse approximate inverse preconditionings I. theory. *SIAM J. Matrix Anal. Appl.*, 14(1):45–58, 1993.

[21] Cornelius Lanczos. An iterative method for the solution of the eigenvalue problem of linear differential and integral operators. *Journal of Research of the National Bureau of Standards*, 45:255–282, 1950.

[22] R. B. Lehoucq and K. Meerbergen. Using generalized Cayley transformations within an inexact rational Krylov sequence method. *SIAM J. Matrix Anal. Appl.*, 20(1):131–148, 1999.

[23] K. Meerbergen and A. Spence. Implicitly restarted arnoldi with purification for the shift-invert transformation. *Mathematics of Computation*, 66(218):667–689, April 1997.

[24] 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.*, 31:148–162, 1977.

[25] R. B. Morgan and D. S. Scott. Generalizations of Davidson's method for computing eigenvalues of sparse symmetric matrices. *SIAM J. Sci. Statist. Comput.*, 7:817–825, 1986.

[26] J. Olsen, P. Jorgensen, and J. Simons. Passing the one-billion limit in full configuration- interaction (FCI) calculations. *Chemical Physics Letters*, 169:463–472, 1990.

[27] Beresford N. Parlett. *The symmetric eigenvalue problem*. Classics in Applied Mathematics. SIAM, Philadelphia, PA, 1998.

[28] G. Peters and J. H. Wilkinson. Inverse iteration, ill-conditioned equations and Newton's method. *SIAM Review*, 21(3):339–360, July 1979.

[29] A. Ruhe. Rational Krylov sequence methods for eigenvalue computation. *Lin. Alg. Appl.*, 58:391–405, 1984.

[30] Yousef Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.*, 14(2):461–469, March 1993.

[31] Yousef Saad. *Numerical Methods for Large Eigenvalue Problems*. Manchester University Press, 1993.

[32] Yousef Saad. Highly parallel preconditioners for general sparse matrices. In G. Golub, A. Greenbaum, and M. Luskin, editors, *Recent Advances in Iterative Methods*, volume 60 of *IMA volumes in mathematics and its applications*, pages 165–199. Springer-Verlag, New York, NY, 1994.

[33] G. L. G. Sleijpen, A. G. L. Booten, D. R. Fokkema, and H. A. van der Vorst. Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems. *BIT*, 36(3):595–633, 1996.

[34] G. L. G. Sleijpen and H. A. van der Vorst. The Jacobi-Davidson method for eigenvalue problems as an accelerated inexact newton scheme. In *Iterative methods in linear algebra II : proceedings of the ImACS International Symposium on Iterative Methods in Linear Algebra, June 17-20, 1995, Blagoevgrad*, pages 377–389, Amsterdam ; New York, March 1995. Elsevier.

[35] G. L. G. Sleijpen and H. A. van der Vorst. A Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 17(2):401–425, 1996.

[36] A. Stathopoulos, Y. Saad, and C. F. Fischer. Robust preconditioning of large, sparse, symmetric eigenvalue problems. *Journal of Computational and Applied Mathematics*, 64:197–215, 1995.

[37] A. Stathopoulos, Y. Saad, and K. Wu. Dynamic thick restarting of the Davidson and the implicitly restarted Arnoldi methods. *SIAM J. Sci. Comput.*, 19(1):227–245, 1998.

[38] R. A. Tapia. Newton's method for optimization problems with equality constraints. *SIAM J. Numer. Anal.*, 11:874–886, 1974.

[39] Kesheng Wu. An experimental study of Krylov subspace accelerators. Technical Report UMSI 96/20, Min-

nesota Supercomputer Institute, University of Minnesota, 1996.

[40] Kesheng Wu. *Preconditioned Techniques for Large Eigenvalue Problems*. PhD thesis, University of Minnesota, 1997. An updated version also appears as Technical Report TR97-038 at the Computer Science Department.