ELSEVIER

Journal of Computational and Applied Mathematics 123 (2000) 155–175

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.nl/locate/cam

The trace minimization method for the symmetric generalized eigenvalue problem☆

Ahmed Sameh *, Zhanye Tong

Department of Computer Sciences, Purdue University, West Lafayette, IN 47907, USA

Received 9 June 1999; received in revised form 3 February 2000

Abstract

In this paper, the trace minimization method for the generalized symmetric eigenvalue problems proposed by Sameh and Wisniewski [35] is reviewed. Convergence of an inexact trace minimization algorithm is established and a variant of the algorithm that uses expanding subspaces is introduced and compared with the block Jacobi–Davidson algorithm.

MSC: 65F15

Keywords: Trace minimization; Jacobi–Davidson scheme; Eigenvalue; Eigenvector

1. Introduction

The generalized eigenvalue problem

$$Ax = \lambda Bx, \tag{1.1}$$

where A and B are $n \times n$ real symmetric matrices with B being positive definite, arises in many applications, most notably in structural mechanics [1,2] and plasma physics [17,19]. Usually, A and B are large, sparse, and only a few of the eigenvalues and the associated eigenvectors are desired. Because of the size of the problem, methods that rely only on operations like matrix–vector multiplications, inner products, and vector updates, that utilize only high-speed memory are usually considered.

Many methods fall into this category (see, for example [42,43]). The basic idea in all of these methods is building a sequence of subspaces that, in the limit, contain the desired eigenvectors. Most

☆ This work was partially supported by NSF grant CCR-9619763.

* Corresponding author.

E-mail addresses: sameh@cs.purdue.edu (A. Sameh), tong@cs.purdue.edu (Z. Tong).

0377-0427/00/$ - see front matter
PII: S0377-0427(00)00391-5

of the early methods iterate on a single vector, i.e., using one-dimensional subspaces, to compute one eigenpair at a time. If several eigenpairs are needed, a deflation technique is frequently used. Another alternative is to use block analogs of the single vector methods to obtain several eigenpairs simultaneously. The well-known *simultaneous iteration* [31], or *subspace iteration* [28], is a block analog of the power method. Simultaneous iteration is originally developed by Bauer [3] under the name *treppeniteration*. It was extensively studied in the late 1960s and early 1970s [5,24,31,41,42].

Let A be symmetric positive definite and assume that the smallest p eigenpairs are the ones we desire to obtain, where $1 \leqslant p \ll n$. In simultaneous iteration, the sequence of subspaces of dimension p is generated by the following recurrence:

$$X_{k+1} = A^{-1}BX_k, \quad k = 0, 1, \ldots, \tag{1.2}$$

where X_0 is an $n \times p$ matrix of full rank. The eigenvectors of interest are magnified at each iteration step, and will eventually dominate X_k. The downside of simultaneous iteration is that linear systems of the form $Ax = b$ have to be solved repeatedly which is a significant challenge for large problems. Solving these linear systems inexactly often compromises global convergence. A variant of simultaneous iteration, called *the trace minimization method*, was proposed in 1982 by Sameh and Wisniewski [35] in an attempt to avoid this difficulty. Let X_k be the current approximation to the eigenvectors corresponding to the p smallest eigenvalues where $X_k^{\mathrm{T}}BX_k = I_p$. The idea of the trace minimization scheme is to find a correction term Δ_k that is B-orthogonal to X_k such that

$$\mathrm{tr}(X_k - \Delta_k)^{\mathrm{T}}A(X_k - \Delta_k) < \mathrm{tr}(X_k^{\mathrm{T}}AX_k).$$

It follows that, for any B-orthonormal basis X_{k+1} of the new subspace $\mathrm{span}\{X_k - \Delta_k\}$, we have

$$\mathrm{tr}(X_{k+1}^{\mathrm{T}}AX_{k+1}) < \mathrm{tr}(X_k^{\mathrm{T}}AX_k),$$

i.e., $\mathrm{span}\{X_k - \Delta_k\}$ gives rise to a better approximation of the desired eigenspace than $\mathrm{span}\{X_k\}$. This trace reduction property can be maintained without solving any linear systems exactly.

Just as simultaneous iteration is accelerated by the use of Chebyshev polynomials, the trace minimization method is accelerated via shifting strategies. The introduction of shifts, however, may compromise the robustness of the trace minimization scheme. Various techniques have been developed to prevent *unstable convergence* (see Section 3.2 for details). A simple way to get around this difficulty is to utilize expanding subspaces. This, in turn, places the trace minimization method into a class of methods that includes the Lanczos method [23], Davidson's method [10], and the Jacobi–Davidson method [12,37,38].

The Lanczos method has become increasingly popular since the ground-breaking analysis by Paige [27], and many practical algorithms are known today [7,14,30,36] (see [8,15] for an overview). The original Lanczos algorithm was developed for handling the standard eigenvalue problem only, i.e., $B = I$. Extensions to the generalized eigenvalue problem [11,21,16] require solving a linear system of the form $Bx = b$ at each iteration step, or factorizing matrices of the form $A - \sigma B$ during the iteration. Davidson's method can be regarded as a preconditioned Lanczos method. It was intended to be a practical method for standard eigenvalue problems in quantum chemistry where the matrices involved are diagonally dominant. In the past two decades, Davidson's method has gone through a series of significant improvements [6,25,26,40,44]. A recent development is the Jacobi–Davidson method [38], published in 1996, which is a variant of Davidson's original scheme and the well-known Newton's method. The Jacobi–Davidson algorithm for the symmetric eigenvalue problem may be regarded as a

generalization of the trace minimization scheme that uses expanding subspaces. Both utilize an idea that dates back to Jacobi [20]. As we will see in Section 5, the current Jacobi–Davidson scheme can be further improved by the techniques developed in the trace minimization method.

In this paper, we give a detailed account of the trace minimization method including the derivation of the scheme, its convergence theory, acceleration techniques, and some implementation details. Some of this material is new. The outline of the paper is as follows. In Section 2, we "derive" the trace minimization method and describe the basic algorithm. In Section 3, we prove convergence of the basic algorithm under the assumption that the inner systems are solved inexactly. Shifting techniques are introduced in Section 4, and a Davidson-type generalization is given in Section 5.

Throughout the paper, the eigenpairs of the eigenvalue problem (1.1) are denoted by (x_i, λ_i), $1 \leqslant i \leqslant n$. The eigenvalues are always arranged in ascending order. The following eigenvalue problem of order 100:

$$\begin{aligned} A &= \operatorname{diag}(1 \times 0.1, 2 \times 0.2, 3 \times 0.3, \ldots, 100 \times 10.0), \\ B &= \operatorname{diag}(0.1, 0.2, 0.3, \ldots, 10.0), \end{aligned} \tag{1.3}$$

will be used in Sections 2–5 to illustrate the techniques discussed in the paper. All numerical experiments for this small eigenvalue problem are performed with MATLAB on a SUN SPARC 5. The initial guesses are generated by the MATLAB function RAND, and the eigenpairs are accepted when the 2-norm of the residual vectors are less than 10^{-10}. Numerical experiments for large problems are performed on SGI/Origin 2000. The results are presented in Section 5.3.

2. The trace minimization method

In this section, we derive the trace minimization method originally presented in [35]. We assume that A is positive definite, otherwise problem (1.1) can be replaced by

$$(A - \mu B)x = (\lambda - \mu)Bx$$

with $\mu < \lambda_1 < 0$, that ensures a positive definite $(A - \mu B)$.

The trace minimization method is motivated by the following theorem.

Theorem 2.1 (Beckenbach and Bellman [4], Sameh and Wisniewski [35]). *Let A and B be as given in problem* (1.1), *and let X^* be the set of all $n \times p$ matrices X for which $X^{\mathrm{T}}BX = I_p$, $1 \leqslant p \leqslant n$. Then*

$$\min_{X \in X^*} \operatorname{tr}(X^{\mathrm{T}}AX) = \sum_{i=1}^{p} \lambda_i. \tag{2.1}$$

where $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_n$ are the eigenvalues of problem (1.1). *The equality holds if and only if the columns of the matrix X, which achieves the minimum, span the eigenspace corresponding to the smallest p eigenvalues.*

If we denote by E/F the matrix EF^{-1} and $\mathscr{X}$ the set of all $n \times p$ matrices of full rank, then (2.1) is equivalent to

$$\min_{X \in \mathscr{X}} \operatorname{tr}\left(\frac{X^{\mathrm{T}}AX}{X^{\mathrm{T}}BX}\right) = \sum_{i=1}^{p} \lambda_i.$$

$X^{\mathrm{T}}AX/X^{\mathrm{T}}BX$ is called the generalized Rayleigh quotient. Most of the early methods that compute a few of the smallest eigenvalues are devised explicitly or implicitly by reducing the generalized Rayleigh quotient step by step. A simple example is the simultaneous iteration scheme for a positive definite matrix A where the current approximation X_k is updated by (1.2). It can be shown by the Courant–Fischer theorem [29, p. 206] and the Kantorovic inequality [22] that

$$\lambda_i\left(\frac{X_{k+1}^{\mathrm{T}}AX_{k+1}}{X_{k+1}^{\mathrm{T}}BX_{k+1}}\right)\leqslant\lambda_i\left(\frac{X_k^{\mathrm{T}}AX_k}{X_k^{\mathrm{T}}BX_k}\right),\quad 1\leqslant i\leqslant p. \tag{2.2}$$

The equality holds only when X_k is already an eigenspace of problem (1.1). Originally, the columns of X_{k+1} were taken as approximations to the desired eigenvectors. It was later found out that a Rayleigh–Ritz process on the subspace span$\{X_{k+1}\}$ yields more accurate approximations. A detailed treatment of simultaneous iteration can be found in [29, Chapter 14]. The following is an outline of the basic algorithm:

Algorithm 1. Simultaneous iteration.
Choose a block size $s\geqslant p$ and an $n\times s$ matrix V_1 of full rank such that $V_1^{\mathrm{T}}BV_1=I_s$.
For $k=1,2,\ldots$ until convergence, do

1. Compute $W_k=AV_k$ and the interaction matrix $H_k=V_k^{\mathrm{T}}W_k$.
2. Compute the eigenpairs (Y_k,Θ_k) of H_k. The eigenvalues are arranged in ascending order and the eigenvectors are chosen to be orthogonal.
3. Compute the corresponding Ritz vectors $X_k=V_kY_k$.
4. Compute the residuals $R_k=W_kY_k-BX_k\Theta_k$.
5. Test for convergence.
6. Solve the linear system
$$AZ_{k+1}=BX_k, \tag{2.3}$$
by an iterative method.
7. B-orthonormalize Z_{k+1} into V_{k+1} by the Gram–Schmidt process with reorthogonalization [9].

End for

In [35], simultaneous iteration was derived in a way that the trace minimization property is explicitly explored. At each iteration step, the previous approximation X_k, which satisfies $X_k^{\mathrm{T}}BX_k=I_s$ and $X_k^{\mathrm{T}}AX_k=\Theta_k$, is corrected with Δ_k that is obtained by

$$\begin{aligned}&\text{minimizing}\quad \mathrm{tr}(X_k-\Delta_k)^{\mathrm{T}}A(X_k-\Delta_k),\\ &\text{subject to}\quad X_k^{\mathrm{T}}B\Delta_k=0.\end{aligned} \tag{2.4}$$

As a result, the matrix $Z_{k+1}=X_k-\Delta_k$ always satisfies

$$\mathrm{tr}(Z_{k+1}^{\mathrm{T}}AZ_{k+1})\leqslant\mathrm{tr}(X_k^{\mathrm{T}}AX_k), \tag{2.5}$$

and

$$Z_{k+1}^{\mathrm{T}}BZ_{k+1}=I_s+\Delta_k^{\mathrm{T}}B\Delta_k, \tag{2.6}$$

which guarantee that

$$\mathrm{tr}(X_{k+1}^{\mathrm{T}}AX_{k+1})\leqslant\mathrm{tr}(X_k^{\mathrm{T}}AX_k) \tag{2.7}$$

for any B-orthonormal basis X_{k+1} of the subspace span$\{Z_{k+1}\}$. The equality in (2.7) holds only when $\Delta_k = 0$, i.e., X_k spans an eigenspace of (1.1) (see Theorem 3.3 for details).

Using Lagrange multipliers, the solution of the minimization problem (2.4) can be obtained by solving the saddle-point problem

$$\begin{bmatrix} A & BX_k \\ X_k^{\mathrm{T}}B & 0 \end{bmatrix} \begin{bmatrix} \Delta_k \\ L_k \end{bmatrix} = \begin{bmatrix} AX_k \\ 0 \end{bmatrix}, \tag{2.8}$$

where $2L_k$ represents the Lagrange multipliers. In [35], (2.8) is further reduced to the following positive-semidefinite system

$$(PAP)\Delta_k = PAX_k, \quad X_k^{\mathrm{T}}B\Delta_k = 0, \tag{2.9}$$

where P is the projector $P = I - BX_k(X_k^{\mathrm{T}}B^2X_k)^{-1}X_k^{\mathrm{T}}B$. This system is solved by the conjugate gradient method (CG) in which zero is chosen as the initial iterate so that the linear constraint $X_k^{\mathrm{T}}B\Delta_k^{(l)} = 0$ is automatically satisfied for any intermediate $\Delta_k^{(l)}$. This results in the following basic trace minimization algorithm:

Algorithm 2. The basic trace minimization algorithm.
Choose a block size $s \geqslant p$ and an $n \times s$ matrix V_1 of full rank such that $V_1^{\mathrm{T}}BV_1 = I_s$.
For $k = 1, 2, \ldots$ until convergence, do

1. Compute $W_k = AV_k$ and the interaction matrix $H_k = V_k^{\mathrm{T}}W_k$.
2. Compute the eigenpairs (Y_k, Θ_k) of H_k. The eigenvalues are arranged in ascending order and the eigenvectors are chosen to be orthogonal.
3. Compute the corresponding Ritz vectors $X_k = V_kY_k$.
4. Compute the residuals $R_k = AX_k - BX_k\Theta_k = W_kY_k - BX_k\Theta_k$.
5. Test for convergence.
6. Solve the positive-semidefinite linear system (2.9) approximately via the CG scheme.
7. B-orthonormalize $X_k - \Delta_k$ into V_{k+1} by the Gram–Schmidt process with reorthogonalization [9].

End for

From now on, we will refer to the linear system (2.9) in step (6) as the *inner system*(s). It is easy to see that the exact solution of the inner system is

$$\Delta_k = X_k - A^{-1}BX_k(X_k^{\mathrm{T}}BA^{-1}BX_k)^{-1}, \tag{2.10}$$

thus the subspace spanned by $X_k - \Delta_k$ is the same subspace spanned by $A^{-1}BX_k$. In other words, if the inner system (2.9) is solved exactly at each iteration step, the trace minimization algorithm above is mathematically equivalent to simultaneous iteration. As a consequence, global convergence of the basic trace minimization algorithm follows exactly from that of simultaneous iteration.

Theorem 2.2 (Rutishauser [32], Parlett [29], Sameh and Wisniewski [35]). *Let A and B be positive definite and let $s \geqslant p$ be the block size such that the eigenvalues of problem* (1.1) *satisfy* $0 < \lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_s < \lambda_{s+1} \leqslant \cdots \leqslant \lambda_n$. *Let also the initial iterate X_0 be chosen such that it has linearly independent columns and is not deficient in any eigen-component associated with the p smallest eigenvalues. Then the ith column of X_k, denoted by $x_{k,i}$, converges to the eigenvector x_i*

corresponding to λ_i *for* $i = 1, 2, \ldots, p$ *with an asymptotic rate of convergence bounded by* λ_i/λ_{s+1}. *More specifically, at each step, the error*

$$\phi_i = (x_{k,i} - x_i)^{\mathrm{T}} A (x_{k,i} - x_i) \tag{2.11}$$

is reduced asymptotically be a factor of $(\lambda_i/\lambda_{s+1})^2$.

The only difference between the trace minimization algorithm and simultaneous iteration is in step (6). If both (2.3) and (2.9) are solved via the CG scheme exactly, the performance of either algorithm is comparable in terms of time consumed, as observed in practice. The additional cost in performing the projection P at each CG step (once rather than twice) is not high because the block size s is usually small, i.e., $1 \leqslant s \ll n$. This additional cost is sometimes compensated for by the fact that PAP, when it is restricted to subspace $\{v \in R^n \mid Pv = v\}$, is better conditioned than A as will be seen in the following theorem.

Theorem 2.3. *Let* A *and* B *be as given in Theorem* 2.2 *and* P *be as given in* (2.9), *and let* v_i, μ_i, $1 \leqslant i \leqslant n$ *be the eigenvalues of* A *and* PAP *arranged in ascending order, respectively. Then, we have*

$$0 = \mu_1 = \mu_2 = \cdots = \mu_s < v_1 \leqslant \mu_{s+1} \leqslant \mu_{s+2} \leqslant \cdots \leqslant \mu_n \leqslant v_n.$$

Proof. The proof is a straightforward consequence of the Courant–Fischer theorem [29, p. 206], and hence omitted. □

3. Practical considerations

In computing practice, however, the inner systems (2.3) and (2.9) are always solved approximately, particularly for large problems. There are two reasons for this: (i) the error (2.11) in the ith column of X_k is reduced asymptotically by a factor of $(\lambda_i/\lambda_{s+1})^2$ at each iteration step. Thus we should not expect high accuracy in the early Ritz vectors even if the inner systems are solved to machine precision, and (ii) it is often too expensive to solve the inner systems to high-order accuracy by an iterative method. Numerical experiments have shown that, for simultaneous iteration, the inner system (2.3) has to be solved in a progressive way, i.e., the absolute stopping tolerance for the inner systems must be decreasing such that it is smaller than the specified error tolerance at the end of the outer iteration. On the contrary, for the trace minimization algorithm, the convergence is guaranteed if a constant relative residual tolerance is used for the inner system (2.9). Table 1 shows the behavior of both algorithms for example (1.3), where $\times$ indicates stagnation.

3.1. A convergence result

In this section, we prove the convergence of the trace minimization algorithm under the assumption that the inner systems in (2.9) are solved inexactly. We assume that, for each i, $1 \leqslant i \leqslant s$, the ith inner system in (2.9) is solved approximately by the CG scheme with zero as the initial iterate such that the 2-norm of the residual is reduced by a factor $\gamma < 1$. The computed correction matrix will be

Table 1
The basic trace minimization algorithm (Algorithm 2) versus simultaneous iteration. The inner systems are solved by the CG scheme which is terminated such that the 2-norm of the residual is reduced by a specified factor. The number of outer iterations (#its) and the number of matrix vector multiplications with A (A mults) are listed for different residual reduction factors

Methods	10^{-4}		10^{-2}		0.5		Dynamic	
	#its	A mults	#its	A mults	#its	A mults	#its	A mults
Simult	×		×		×		×	
Tracmn	59	6638	59	4263	77	4030	66	4479

denoted by $\Delta_k^c = \{d_{k,1}^c, d_{k,2}^c, \ldots, d_{k,s}^c\}$ to distinguish it from the exact solution $\Delta_k = \{d_{k,1}, d_{k,2}, \ldots, d_{k,s}\}$ of (2.9).

We begin the convergence proof with two lemmas. We first show that, at each iteration step, the columns of $X_k - \Delta_k^c$ are linearly independent, and the sequence $\{X_k\}_0^\infty$ in the trace minimization algorithm is well-defined. In Lemma 3.2, we show that the computed correction matrix Δ_k^c satisfies

$$\mathrm{tr}(X_k - \Delta_k^c)^T A(X_k - \Delta_k^c) \leqslant \mathrm{tr}(X_k^T A X_k).$$

This assures that, no matter how prematurely the CG process is terminated, the trace $\mathrm{tr}(X_k^T A X_k)$ always forms a decreasing sequences bounded from below by $\sum_{i=1}^s \lambda_i$.

Lemma 3.1. *For each $k = 0, 1, 2, \ldots, Z_{k+1} = X_k - \Delta_k^c$ is of full rank.*

Proof. Since $d_{k,i}^c$ is an intermediate approximation obtained from the CG process, there exists a polynomial $p(t)$ such that

$$d_{k,i}^c = p(PAP)(PAx_{k,i}),$$

where $x_{k,i}$ is the ith column of X_k and P is the projector in (2.9). As a consequence, for each i, $d_{k,i}^c$ is B-orthogonal to X_k, i.e., $X_k^T B d_{k,i}^c = 0$. Thus the matrix

$$Z_{k+1}^T B Z_{k+1} = I_s + \Delta_k^{cT} B \Delta_k^c$$

is nonsingular, and Z_{k+1} is of full rank. □

Lemma 3.2. *Suppose that the inner systems in* (2.9) *are solved by the CG scheme with zero as the initial iterate. Then, for each i, $(x_{k,i} - d_{k,i}^{(l)})^T A(x_{k,i} - d_{k,i}^{(l)})$ decreases monotonically with respect to step l of the CG scheme.*

Proof. The exact solution of the inner system (2.9),

$$\Delta_k = X_k - A^{-1}BX_k(X_k^T BA^{-1}BX_k)^{-1}$$

satisfies $P\Delta_k = \Delta_k$. For each i, $1 \leqslant i \leqslant s$, the intermediate $d_{k,i}^{(l)}$ in the CG process also satisfies $Pd_{k,i}^{(l)} = d_{k,i}^{(l)}$. It follows that

$$\begin{aligned}(d_{k,i}^{(l)} - d_{k,i})^T PAP(d_{k,i}^{(l)} - d_{k,i}) &= (d_{k,i}^{(l)} - d_{k,i})^T A(d_{k,i}^{(l)} - d_{k,i}) \\ &= (x_{k,i} - d_{k,i}^{(l)})^T A(x_{k,i} - d_{k,i}^{(l)}) - [(X_k^T BA^{-1}BX_k)^{-1}]_{ii}.\end{aligned}$$

Since the CG process minimizes the PAP-norm of the error $e_{k,i}^{(l)} = d_{k,i}^{(l)} - d_{k,i}$ on the expanding Krylov subspace [33, p. 130], we have $(d_{k,i}^{(l)} - d_{k,i})^{\mathrm{T}} PAP(d_{k,i}^{(l)} - d_{k,i})$ decreases monotonically. So does $(x_{k,i} - d_{k,i}^{(l)})^{\mathrm{T}} A(x_{k,i} - d_{k,i}^{(l)})$. □

Theorem 3.3. *Let X_k, Δ_k^{c}, and Z_{k+1} be as given for Lemma* 3.1. *Then we have* $\lim_{k\to\infty} \Delta_k^{\mathrm{c}} = 0$.

Proof. First, by the definition of Δ_k^{c}, we have

$$Z_{k+1}^{\mathrm{T}} B Z_{k+1} = I_s + \Delta_k^{\mathrm{cT}} B \Delta_k^{\mathrm{c}} \triangleq I_s + T_k.$$

Consider the spectral decomposition of $Z_{k+1}^{\mathrm{T}} B Z_{k+1}$

$$Z_{k+1}^{\mathrm{T}} B Z_{k+1} = U_{k+1} D_{k+1}^2 U_{k+1}^{\mathrm{T}},$$

where U_{k+1} is an $s \times s$ orthogonal matrix and $D_{k+1}^2 = \mathrm{diag}(\delta_1^{(k+1)}, \delta_2^{(k+1)}, \ldots, \delta_s^{(k+1)})$. It is easy to see that $\delta_i^{(k+1)} = 1 + \lambda_i(T_k) \geqslant 1$.

Second, by the definition of X_{k+1}, there exists an orthogonal matrix V_{k+1} such that

$$X_{k+1} = Z_{k+1} \cdot U_{k+1} D_{k+1}^{-1} V_{k+1}.$$

Denote by $z_i^{(k+1)}$ the diagonal elements of the matrix $U_{k+1}^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} A Z_{k+1} U_{k+1}$. It follows that

$$\begin{aligned}
\mathrm{tr}(X_{k+1}^{\mathrm{T}} A X_{k+1}) &= \mathrm{tr}(D_{k+1}^{-1}(U_{k+1}^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} A Z_{k+1} U_{k+1}) D_{k+1}^{-1}), \\
&= \frac{z_1^{(k+1)}}{\delta_1^{(k+1)}} + \frac{z_2^{(k+1)}}{\delta_2^{(k+1)}} + \cdots + \frac{z_s^{(k+1)}}{\delta_s^{(k+1)}}, \\
&\leqslant z_1^{(k+1)} + z_2^{(k+1)} + \cdots z_s^{(k+1)}, \\
&= \mathrm{tr}(Z_{k+1}^{\mathrm{T}} A Z_{k+1}), \\
&\leqslant \mathrm{tr}(X_k^{\mathrm{T}} A X_k),
\end{aligned}$$

which implies that

$$\cdots \geqslant \mathrm{tr}(X_k^{\mathrm{T}} A X_k) \geqslant \mathrm{tr}(Z_{k+1}^{\mathrm{T}} A Z_{k+1}) \geqslant \mathrm{tr}(X_{k+1}^{\mathrm{T}} A X_{k+1}) \geqslant \cdots.$$

Since the sequence is bounded from below by $\sum_{i=1}^{s} \lambda_i$, it converges to a positive number $t \geqslant \sum_{i=1}^{s} \lambda_i$. Moreover, the two sequences

$$\frac{z_1^{(k+1)}}{\delta_1^{(k+1)}} + \frac{z_2^{(k+1)}}{\delta_2^{(k+1)}} + \cdots + \frac{z_s^{(k+1)}}{\delta_s^{(k+1)}}, \quad k = 1, 2, \ldots$$

and

$$z_1^{(k+1)} + z_2^{(k+1)} + \cdots z_s^{(k+1)}, \quad k = 1, 2, \ldots$$

also converge to t. Therefore,

$$\left(\frac{z_1^{(k+1)} \lambda_1(T_k)}{1 + \lambda_1(T_k)}\right) + \left(\frac{z_2^{(k+1)} \lambda_2(T_k)}{1 + \lambda_2(T_k)}\right) + \cdots + \left(\frac{z_s^{(k+1)} \lambda_s(T_k)}{1 + \lambda_s(T_k)}\right) \to 0.$$

Observing that for any $i, 1 \leqslant i \leqslant s$,

$$\begin{aligned} z_i^{(k+1)} &\geqslant \lambda_1(U_{k+1}^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} A Z_{k+1} U_{k+1}), \\ &= \lambda_1(Z_{k+1}^{\mathrm{T}} A Z_{k+1}), \\ &= \min_{y \neq 0} \frac{y^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} A Z_{k+1} y}{y^{\mathrm{T}} y}, \\ &= \min_{y \neq 0} \left(\frac{y^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} A Z_{k+1} y}{y^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} B Z_{k+1} y} \right) \cdot \left(\frac{y^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} B Z_{k+1} y}{y^{\mathrm{T}} y} \right), \\ &\geqslant \min_{y \neq 0} \frac{y^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} A Z_{k+1} y}{y^{\mathrm{T}} Z_{k+1}^{\mathrm{T}} B Z_{k+1} y}, \\ &\geqslant \lambda_1(A, B), \\ &> 0, \end{aligned}$$

we have

$$\lambda_1(T_k) \to 0, \quad i = 1, 2, \ldots, s,$$

i.e., $\lim_{k \to \infty} \Delta_k^c = 0$. □

Theorem 3.4. *If, for each* i, $1 \leqslant i \leqslant s$, *the CG process for the* i*th inner system*

$$(PAP) d_{k,i} = P A x_{k,i}, \quad d_{k,i}^{\mathrm{T}} B X_k = 0,$$

in (2.9) *is terminated such that the* 2*-norm of the residual is reduced by a factor* $\gamma < 1$, *i.e.*,

$$||P A x_{k,i} - (PAP) d_{k,i}^{\mathrm{c}}||_2 \leqslant \gamma ||P A x_{k,i}||_2, \tag{3.1}$$

then columns of X_k *converge to* s *eigenvectors of problem* (1.1).

Proof. Condition (3.1) implies that

$$||P A x_{k,i}||_2 - ||P A d_{k,i}^{\mathrm{c}}||_2 \leqslant \gamma ||P A x_{k,i}||_2,$$

and consequently

$$||P A x_{k,i}||_2 \leqslant \frac{1}{1-\gamma} ||P A d_{k,i}^{\mathrm{c}}||_2.$$

It follows from Theorem 3.3 that $\lim_{k \to \infty} PAX_k = 0$, i.e.,

$$\lim_{k \to \infty} (AX_k - BX_k[(X_k^{\mathrm{T}} B^2 X_k)^{-1} X_k^{\mathrm{T}} B A X_k]) = 0.$$

This shows that $span\{X_k\}$ converges to an eigenspace of problem (1.1). □

3.2. Randomization

Condition (3.1) in Theorem 3.4 is not essential because the constant γ can be arbitrarily close to 1. The only deficiency in Theorem 3.4 is that it does not establish ordered convergence in the sense that the ith column of X_k converges to the ith eigenvector of the problem. This is called *unstable*

convergence by Rutishauser. In computing practice, roundoff errors usually turn unstable convergence into delayed stable convergence. In [32], Rutishauser introduced a randomization technique to prevent unstable convergence in simultaneous iteration; it can be incorporated into the trace minimization algorithm as well: *After step* (6) *of Algorithm* 2, *we append a random vector to* X_k *and perform the Ritz processes* (1)–(2) *on the augmented subspace of dimension* $s+1$. *The extra Ritz pair is discarded after step* (2).

Randomization slightly improves the convergence of the first s Ritz pairs [31]. Since it comes with additional cost, it should be used only in the first few steps and when a Ritz pair is about to converge.

3.3. Terminating the CG process

Theorem 3.4 gives a sufficient condition for the convergence of the trace minimization algorithm. However, the asymptotic rate of convergence of the trace minimization algorithm will be affected by the premature termination of the CG processes. Table 3.1 shows how differently the trace minimization algorithm behaves when the inner systems are solved inexactly. It is not clear how the parameter γ should be chosen to avoid performing excessive CG iterations while maintaining the asymptotic rate of convergence. In [35], the CG processes are terminated by a heuristic stopping strategy.

Denote by $d_{k,i}^{(l)}$ the approximate solution at the lth step of the CG process for the ith column of X_k, and $d_{k,i}$ the exact solution. The heuristic stopping strategy in [35] can be outlined as follows:

1. From Theorem 2.2, it is reasonable to terminate the CG process for the ith column of Δ_k when the error
$$\varepsilon_{k,i}^{(l)} = [(d_{k,i}^{(l)} - d_{k,i})^{\mathrm{T}} A (d_{k,i}^{(l)} - d_{k,i})]^{1/2},$$
is reduced by a factor of $\tau_i = \lambda_i/\lambda_{s+1}$, called *error reduction factor*.
2. The quantity $\varepsilon_{k,i}^{(l)}$ can be estimated by
$$[(d_{k,i}^{(l)} - d_{k,i}^{(l+1)})^{\mathrm{T}} A (d_{k,i}^{(l)} - d_{k,i}^{(l+1)})]^{1/2},$$
which is readily available from the CG process.
3. The error reduction factor $\tau_i = \lambda_i/\lambda_{s+1}, 1 \leqslant i \leqslant s$, can be estimated by $\tau_{k,i} = \theta_{k,i}/\theta_{k,s+1}$. Since $\theta_{k,s+1}$ is not available, $\theta_{k-1,s}$ is used instead and is fixed after a few steps because it will eventually converge to λ_s rather than λ_{s+1}.

This strategy has worked well in practice. The last column of Table 1 shows the result obtained with this stopping strategy.

4. Acceleration techniques

The algorithm discussed in Section 3 effectively reduces the work at each iteration step. It requires, however, about the same number of outer iteration steps as the simultaneous iteration. For problems in which the desired eigenvalues are poorly separated from the remaining part of the spectrum, the algorithm converges too slowly. Like other inverse iteration schemes, the trace minimization

Table 2
The trace minimization algorithm with various shifting strategies

Safe shift		Single shift		Multiple shifts	
#its	A mults	#its	A mults	#its	A mults
46	4153	22	3619	18	3140

algorithm can be accelerated by shifting. Actually, the formulation of the trace minimization algorithm makes it easier to incorporate shifts. For example, if eigenpairs $(x_i, \theta_i), 1 \leqslant i \leqslant i_0$, have been accepted and $\theta_{i_0} < \theta_{i_0+1}, \theta_{i_0}$ can be used as a shift parameter for computing subsequent eigenpairs. Due to the deflation effect, the linear systems

$$[P(A - \theta_{i_0}B)P]d_{k,i} = PAx_{k,i}, \quad X_k^{\mathrm{T}}Bd_{k,i} = 0, \quad i_0 + 1 \leqslant i \leqslant s,$$

are consistent and can still be solved by the CG scheme. Moreover, the trace reduction property still holds. The first column of Table 2 shows the result of the trace minimization scheme with such a conservative shifting strategy, which we call *safe shift*. The performance is obviously improved over that of the basic trace minimization algorithm shown in Table 1. In the following, we introduce two more efficient shifting techniques which improve further the performance of the trace minimization algorithm.

4.1. Single shift

We know from Section 2 that global convergence of the trace minimization algorithm follows from the monotonic reduction of the trace, which in turn depends on the positive definiteness of A. A simple and robust shifting strategy would be finding a scalar σ close to λ_1 from below and replace A with $A - \sigma B$ in step (6) of the algorithm. After the first eigenvector is converged, find another σ close to λ_2 from below and continue until all the desired eigenvectors are obtained. If both A and B are explicitly available, it is not hard to find a σ satisfying $\sigma \leqslant \lambda_1$. Gerschgorin disks [13], for example, provide reliable bounds on the spectrum of (1.1). These bounds, however, are usually too loose to be useful.

In the trace minimization algorithm, the subspace spanned by X_k converges to the invariant subspace V_s corresponding to the s smallest eigenvalues. If the subspace spanned by X_k is close enough to V_s, a reasonable bound for the smallest eigenvalue can be obtained. More specifically, let Q be a B-orthonormal matrix obtained by appending $n - s$ columns to X_k, i.e., $Q = (X_k, Y_k)$ and $Q^{\mathrm{T}}BQ = I_n$. Then problem (1.1) is reduced to the standard eigenvalue problem

$$(Q^{\mathrm{T}}AQ)u = \lambda u. \tag{4.1}$$

Since

$$Q^{\mathrm{T}}AQ = \begin{bmatrix} \Theta_k & X_k^{\mathrm{T}}AY_k \\ Y_k^{\mathrm{T}}AX_k & Y_k^{\mathrm{T}}AY_k \end{bmatrix} = \begin{bmatrix} \Theta_k & C_k^{\mathrm{T}} \\ C_k & Y_k^{\mathrm{T}}AY_k \end{bmatrix}, \tag{4.2}$$

by the Courant–Fischer theorem, we have

$$\lambda_1 \geqslant \lambda_{\min}\begin{bmatrix}\Theta_k & 0\\ 0 & Y_k^{\mathrm{T}}AY_k\end{bmatrix} + \lambda_{\min}\begin{bmatrix}0 & C_k^{\mathrm{T}}\\ C_k & 0\end{bmatrix}$$
$$\geqslant \min\{\theta_1, \lambda_1(Y_k^{\mathrm{T}}AY_k)\} - ||C_k||_2.$$

Similar to [29, p. 241], it is easy to derive $||C_k||_2 = ||R_k||_{B^{-1}}$, in which $R_k = AX_k - BX_k\Theta_k$ is the residual matrix. If

$$\theta_{k,1} \leqslant \lambda_1(Y_k^{\mathrm{T}}AY_k), \tag{4.3}$$

we get

$$\lambda_1 \geqslant \theta_{k,1} - ||R_k||_{B^{-1}}. \tag{4.4}$$

In particular, if (4.3) holds for the orthonormal complement of $x_{k,1}$, we have

$$\lambda_1 \geqslant \theta_{k,1} - ||r_{k,1}||_{B^{-1}}. \tag{4.5}$$

This heuristic bound for the smallest eigenvalue suggests the following shifting strategy (we denote $-\infty$ by λ_0):

If the first i_0, $i_0 \geqslant 0$, eigenvalues have converged, use $\sigma = \max\{\lambda_{i_0}, \theta_{k,i_0+1} - ||r_{k,i_0+1}||_{B^{-1}}\}$ as the shift parameter. If θ_{k,i_0+1} lies in a cluster, replace r_{k,i_0+1} by the residual matrix corresponding to the cluster containing θ_{k,i_0+1}.

4.2. Multiple dynamic shifts

In [35], the trace minimization algorithm is accelerated with a more aggressive shifting strategy. At the beginning of the algorithm, a single shift is used for all the columns of X_k. As the algorithm proceeds, multiple shifts are introduced dynamically and the CG process is modified to handle possible breakdown. This shifting strategy is motivated by the following theorem.

Theorem 4.1 (Parlett [29, p. 357]). *For an arbitrary nonzero vector u and scalar σ, there is an eigenvalue λ of (1.1) such that*

$$|\lambda - \sigma| \leqslant ||(A - \sigma B)u||_{B^{-1}}/||Bu||_{B^{-1}}.$$

We know from the Courant–Fischer theorem that the targeted eigenvalue λ_i is always below the Ritz value $\theta_{k,i}$. Further, from Theorem 4.1, if $\theta_{k,i}$ is already very close to the targeted eigenvalue λ_i, then λ_i must lie in the interval $[\theta_{k,i} - ||r_{k,i}||_{B^{-1}}, \theta_{k,i}]$. This observation leads to the following shifting strategy for the trace minimization algorithm. At step k of the outer iteration, the shift parameters $\sigma_{k,i}$, $1 \leqslant i \leqslant s$, are determined by the following rules (Here, $\lambda_0 = -\infty$ and the subscript k is dropped for the sake of simplicity):

1. If the first i_0, $i_0 \geqslant 0$, eigenvalues have converged, choose

$$\sigma_{k,i_0+1} = \begin{cases}\theta_{i_0+1} & \text{if } \theta_{i_0+1} + ||r_{i_0+1}||_{B^{-1}} \leqslant \theta_{i_0+2} - ||r_{i_0+2}||_{B^{-1}},\\ \max\{\theta_{i_0+1} - ||r_{i_0+1}||_{B^{-1}}, \lambda_{i_0}\} & \text{otherwise.}\end{cases}$$

2. For any other column j, $i_0+1<j\leqslant p$, choose the largest θ_l such that

$$\theta_l<\theta_j-||r_j||_{B^{-1}}$$

as the shift parameter σ_j. If no such θ_l exists, use θ_{i_0+1} instead.
3. Choose $\sigma_i=\theta_i$ if θ_{i-1} has been used as the shift parameter for column $i-1$ and

$$\theta_i<\theta_{i+1}-||r_{i+1}||_{B^{-1}}.$$

4. Use σ_{i_0+1} as the shift parameters for other columns if any.

This heuristic shifting strategy turns out to be quite efficient and robust in practice. Table 2 shows the results for the shifting strategies discussed in this section. Since A is positive definite, zero is a good shift parameter. In our experiments, however, we did not take advantage of this fact and selected the shift parameters according to the strategies described above with B^{-1}-norms replaced by 2-norms. We see that both the number of outer iteration steps and the number of matrix vector multiplications with A are reduced considerably by the multiple dynamic shifting strategy. The number of matrix vector multiplications with B is not shown in the table because it is almost identical to that with A.

4.3. Solving the inner systems

With multiple shifts, the inner systems in (2.9) become

$$[P(A-\sigma_{k,i}B)P]d_{k,i}=PAx_{k,i},\quad X_k^{\mathrm{T}}Bd_{k,i}=0,\quad 1\leqslant i\leqslant s \tag{4.6}$$

with $P=I-BX_k(X_k^{\mathrm{T}}B^2X_k)^{-1}X_k^{\mathrm{T}}B$. Clearly, the linear systems can be indefinite, and the CG processes for such systems are numerically unstable and may break down. A simple way to get around this problem is terminating the CG process when a near breakdown is detected. In [35], the CG process is also terminated when the error $(x_{k,i}-d_{k,i}^{(l)})^{\mathrm{T}}A(x_{k,i}^{(l)}-d_{k,i}^{(l)})$, increases by a small factor. This helps maintain global convergence which is not guaranteed in the presence of shifting.

Due to the deflation effect, the inner systems in (4.6) are usually not ill-conditioned when restricted to the subspace $\{v\in R^n\,|\,Pv=v\}$ unless some of the gap ratios $(\lambda_{s+1}-\lambda_i)/(\lambda_n-\lambda_i)$, $1\leqslant i\leqslant p$, are small. In this case, the inner systems have to be preconditioned. Suppose $\hat{A}=CC^{\mathrm{T}}$ is a symmetric positive definite preconditioner of $A-\sigma_{k,i}B$ (for example, an approximate incomplete Cholesky factorization of $A-\sigma_{k,i}B$). The ith indefinite system in (4.6) can be written as

$$[\tilde{P}(\tilde{P}-\sigma_{k,i}\tilde{B})\tilde{P}]\tilde{d}_{k,i}=\tilde{P}\tilde{P}\tilde{x}_{k,i},\quad \tilde{X}_k^{\mathrm{T}}\tilde{B}\tilde{d}_{k,i}=0, \tag{4.7}$$

with

$$\tilde{A}=C^{-1}AC^{-\mathrm{T}},\quad \tilde{B}=C^{-1}BC^{-\mathrm{T}},\quad \tilde{d}_{k,i}=C^{\mathrm{T}}d_{k,i},\quad \tilde{X}_k=C^{\mathrm{T}}X_k,\quad \tilde{x}_{k,i}=C^{\mathrm{T}}x_{k,i},$$

and

$$\tilde{P}=I-\tilde{B}\tilde{X}_k(\tilde{X}_k^{\mathrm{T}}\tilde{B}^2\tilde{X}_k)^{-1}\tilde{X}_k^{\mathrm{T}}\tilde{B}.$$

Since it is usually difficult to construct a symmetric positive-definite preconditioner for a symmetric indefinite matrix, we suggest that a fixed preconditioner be used for all the matrices $A-\sigma_{k,i}B$.

In the presence of shifting, the asymptotic error reduction factor for the ith Ritz vector becomes $(\lambda_i - \sigma_{k,i})/(\lambda_{s+1} - \sigma_{k,i})$. As a consequence, the CG process is now terminated when the error

$$\varepsilon_{k,i}^{(l)} = [(d_{k,i}^{(l)} - d_{k,i})^{\mathrm{T}}(A - \sigma_{k,i}B)(d_{k,i}^{(l)} - d_{k,i})]^{1/2}$$

is reduced by a factor of

$$\tau_i = \begin{cases} (\theta_{k,i} - \sigma_{k,i})/(\theta_{k,s+1} - \sigma_{k,i}), & \theta_{k,i} \neq \theta_{k,i}, \\ (\theta_{k-1,i} - \sigma_{k,i})/(\theta_{k,s+1} - \sigma_{k,i}), & \theta_{k,i} = \theta_{k,i} \end{cases} \tag{4.8}$$

and $\theta_{k,s+1}$ is estimated as in Section 3.3. In practice, we have terminated the CG process when the 2-norm of the residual is reduced by a factor of τ_i.

5. A Davidson-type generalization

The shifting strategies described in Section 4 improve the performance of the trace minimization algorithm considerably. Although the randomization technique, the shifting strategy, and the roundoff error actually make the algorithm surprisingly robust for a variety of problems, further measures to guard against unstable convergence are necessary for problems in which the desired eigenvalues are clustered. A natural way to maintain stable convergence is by using expanding subspaces, with which the trace reduction property is automatically maintained.

The best-known method that utilizes expanding subspaces is that of Lanczos. It uses the Krylov subspaces to compute an approximation of the desired eigenpairs, usually the largest. This idea was adopted by Davidson, in combination with the simultaneous coordinate relaxation method, to obtain what he called the "compromise method" [10], known as Davidson's method today. In this section, we generalize the trace minimization algorithm described in the previous sections by casting it into the framework of the Davidson method. We start by the Jacobi–Davidson method, explore its connection to the trace minimization method, and develop a Davidson-type trace minimization algorithm.

5.1. The Jacobi–Davidson method

As was mentioned in Section 1, the Jacobi–Davidson scheme is a modification of the Davidson method. It uses the same ideas presented in the trace minimization method to compute a correction term to a previous computed Ritz pair, but with a different objective. In the Jacobi–Davidson method, for a given Ritz pair (x_i, θ_i) with $x_i^{\mathrm{T}}Bx_i = 1$, a correction vector d_i is sought such that

$$A(x_i + d_i) = \lambda_i B(x_i + d_i), \quad x_i^{\mathrm{T}}Bd_i = 0, \tag{5.1}$$

where λ_i is the eigenvalue targeted by θ_i. Since the targeted eigenvalue λ_i is not available during the iteration, it is replaced by an approximation σ_i. Ignoring high-order terms in (5.1), we get

$$\begin{bmatrix} A - \sigma_i B & Bx_i \\ x_i^{\mathrm{T}}B & 0 \end{bmatrix} \begin{bmatrix} d_i \\ l_i \end{bmatrix} = \begin{bmatrix} -r_i \\ 0 \end{bmatrix}, \tag{5.2}$$

where $r_i = Ax_i - \theta_i Bx_i$ is the residual vector associated with the Ritz pair (x_i, θ_i). Note that replacing r_i with Ax_i does not affect d_i. In [37,38], the Ritz value θ_i is used in place of σ_i at each step. A block Jacobi–Davidson algorithm, described in [37], is outlined as follows:

Algorithm 3. The block Jacobi–Davidson algorithm.
Choose a block size $s \geqslant p$ and an $n \times s$ matrix V_1 such that $V_1^{\mathrm{T}} B V_1 = I_s$.
For $k = 1, 2, \ldots$ until convergence, do

1. Compute $W_k = AV_k$ and the interaction matrix $H_k = V_k^{\mathrm{T}} W_k$.
2. Compute the s smallest eigenpairs (Y_k, Θ_k) of H_k. The eigenvalues are arranged in ascending order and the eigenvectors are chosen to be orthogonal.
3. Compute the corresponding Ritz vectors $X_k = V_k Y_k$.
4. Compute the residuals $R_k = W_k Y_k - BX_k \Theta_k$.
5. Test for convergence.
6. for $1 \leqslant i \leqslant s$, solve the indefinite system

$$\begin{bmatrix} A - \theta_i B & Bx_{k,i} \\ x_{k,i}^{\mathrm{T}} B & 0 \end{bmatrix} \begin{bmatrix} d_{k,i} \\ l_{k,i} \end{bmatrix} = \begin{bmatrix} r_{k,i} \\ 0 \end{bmatrix}, \tag{5.3}$$

or preferably its projected form

$$[P_i(A - \theta_{k,i} B)P_i] d_{k,i} = P_i r_{k,i}, \quad x_{k,i}^{\mathrm{T}} B d_{k,i} = 0, \tag{5.4}$$

approximately, where $P_i = I - Bx_{k,i}(x_{k,i}^{\mathrm{T}} B^2 x_{k,i})^{-1} x_{k,i}^{\mathrm{T}} B$ is an orthogonal projector, and $r_{k,i} = Ax_{k,i} - \theta_{k,i} Bx_{k,i}$ is the residual corresponding to the Ritz pair $(x_{k,i}, \theta_{k,i})$.

7. If $\dim(V_k) \leqslant m - s$, then

$$V_{k+1} = \mathrm{Mod}\, GS_B(V_k, \Delta_k),$$

else

$$V_{k+1} = \mathrm{Mod}\, GS_B(X_k, \Delta_k).$$

Here, Mod GS_B stands for the Gram–Schmidt process with reorthogonalization [9] with respect to B-inner products, i.e. $(x, y) = x^{\mathrm{T}} B y$.

End for

This algorithm can be regarded as a trace minimization algorithm with expanding subspaces. The performance of the block Jacobi–Davidson algorithm depends on how good the initial guess is and how efficiently and accurately the inner system (5.3) is solved.

If the right-hand side of (5.3) is taken as the approximate solution to the inner system (5.3), the algorithm is reduced to the Lanczos method. If the inner system (5.3) is solved to high-order accuracy, it is reduced to simultaneous Rayleigh quotient iteration (RQI, see [28]) with expanding subspaces, which converges cubically. If the inner system (5.3) is solved crudely, the performance of the algorithm is in-between. Cubic convergence has been observed for some test problems [38]. In practice, however, the stage of cubic convergence is often reached after many iterations. Fig. 1 shows the convergence history of the block Jacobi–Davidson algorithm for the sample problem (1.3), where four eigenpairs are computed with $m = 20$ and only the errors in the first Ritz value are plotted. The algorithm always "stagnates" at the beginning and increasing the number of iteration steps for the inner systems makes little difference or, in some cases, even derails convergence to the desired eigenpairs. This can be explained by the following. On the one hand, the Ritz shifting strategy in the block Jacobi–Davidson algorithm forces the algorithm to converge to eigenvalues closest to the Ritz values that are often far away from the desired eigenvalues at the beginning of the iteration. On

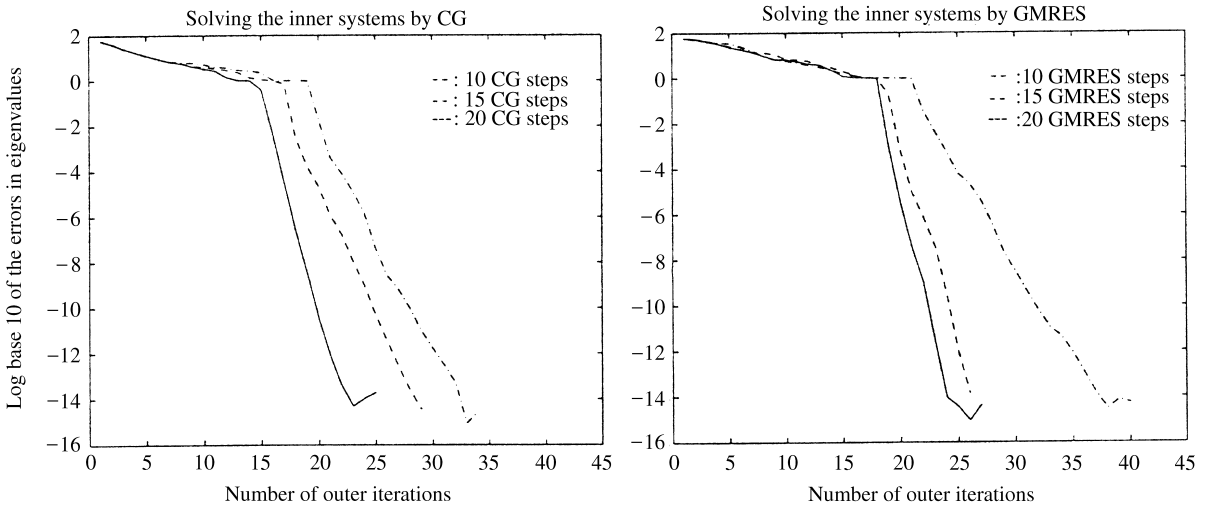

Fig. 1. The block Jacobi–Davidson algorithm.

the other hand, since the subspace is expanding, the Ritz values are decreasing and the algorithm is forced to converge to the smallest eigenpairs.

Another problem with the block Jacobi–Davidson algorithm is ill-conditioning. At the end of the Jacobi–Davidson iteration, when a Ritz value approaches a multiple eigenvalue or a cluster of eigenvalues, the inner system (5.4) becomes poorly conditioned. This makes it difficult for an iterative solver to compute even a crude approximation to the solution of the inner system.

All these problems can be partially solved by the techniques developed in the trace minimization method, i.e., the multiple dynamic shifting strategy, the implicit deflation technique ($d_{k,i}$ is required to be B-orthogonal to all the Ritz vectors obtained in the previous iteration step), and the dynamic stopping strategy. We call the modified algorithm *the Davidson-type trace minimization algorithm* [34].

5.2. The Davidson-type trace minimization algorithm

Let $s \geqslant p$ be the block size, $m \geqslant s$ be a given integer that limits the dimension of the subspaces. The Davidson-type trace minimization algorithm is as follows.

Algorithm 4. The Davidson-type trace minimization algorithm.
Choose a block size $s \geqslant p$ and an $n \times s$ matrix V_1 such that $V_1^{\mathrm{T}} B V_1 = I_s$.
For $k = 1, 2, \ldots$ until convergence, do

1. Compute $W_k = AV_k$ and the interaction matrix $H_k = V_k^{\mathrm{T}} W_k$.
2. Compute the s smallest eigenpairs (Y_k, Θ_k) of H_k. The eigenvalues are arranged in ascending order and the eigenvectors are chosen to be orthogonal.
3. Compute the corresponding Ritz vectors $X_k = V_k Y_k$.
4. Compute the residuals $R_k = W_k Y_k - B X_k \Theta_k$.
5. Test for convergence.

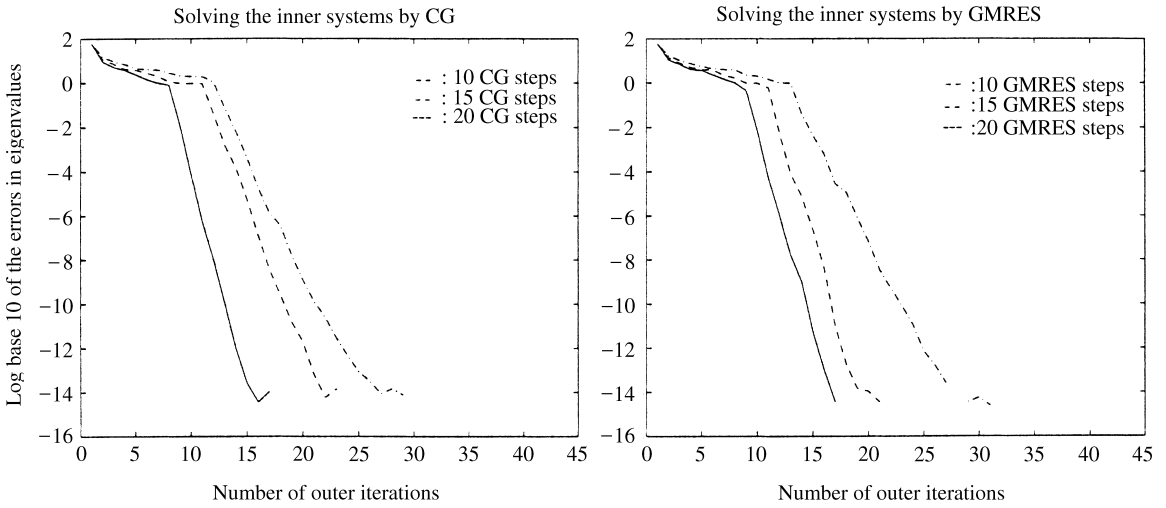

Fig. 2. The Davidson-type trace minimization algorithm.

6. For $1 \leqslant i \leqslant s$, solve the indefinite system

$$[P(A - \sigma_{k,i}B)P]d_{k,i} = Pr_{k,i}, \quad X_k^{\mathrm{T}} B d_{k,i} = 0, \tag{5.5}$$

to a certain accuracy determined by the stopping criterion described in Section 4.3. The shift parameters $\sigma_{k,i}$, $1 \leqslant i \leqslant s$, are determined according to the dynamic shifting strategy described in Section 4.2.

7. If $\dim(V_k) \leqslant m - s$, then

$$V_{k+1} = \mathrm{Mod}\, GS_B(V_k, \Delta_k)$$

else

$$V_{k+1} = \mathrm{Mod}\, GS_B(X_k, \Delta_k).$$

End for

The orthogonality requirement $d_i^{(k)} \perp_B X_k$ is essential in the original trace minimization algorithm for maintaining the trace reduction property (2.7). In the current algorithm, it appears primarily as an implicit deflation technique. A more efficient approach is to require $d_i^{(k)}$ to be B-orthogonal only to "good" Ritz vectors. Fig. 2 displays the convergence history of the Davidson-type trace minimization algorithm for the sample problem (1.3) where $d_i^{(k)}$ is only required to be B-orthogonal to $x_{k,i}$. The number of outer iterations is decreased compared to the trace minimization algorithm in Section 4, and compared to the block Jacobi–Davidson algorithm: 15 iterations vs. 18 and 22 iterations, respectively. Moreover, in the block Jacobi–Davidson algorithm, the number of outer iterations cannot be reduced further when the number of iterations for the inner systems reaches 30. On the contrary, in the Davidson-type trace minimization algorithm, the number of outer iterations decreases steadily even when the number of iterations for the inner systems reaches 50. Note that reducing the number of outer iterations is important in a parallel or distributed computing environment.

Table 3
Numerical results for the test problem in [6] with 4 processors

Inner iterations	Block Jacobi–Davidson			Davidson-type Tracemin		
	#its	A mults	Time	#its	A mults	Time
10	208	9368	28.5	216	9728	29.8
20	103	8760	19.2	76	6468	14.8
40	69	11392	19.0	34	5616	9.3
60	54	13236	21.0	27	6564	10.1
80	48	15608	22.0	24	7808	11.3
100	57	23065	31.3	20	8108	11.6
DS(MAX=120)	33	9653	17.0	23	7364	11.2

5.3. Numerical results

The block Jacobi–Davidson algorithm and the Davidson-type trace minimization algorithm have been coded in C with MPI [18] and PETSc [39]. Numerical experiments have been done on a variety of problems. In this section, we present some of the numerical results obtained on the SGI/Origin 2000.

We first show the results for an example used in [6]. This is a standard eigenvalue problem. The matrix A is defined by

$$a_{ij} = \begin{cases} i & \text{if } i = j, \\ 0.5 & \text{if } j = i+1 \text{ or } j = i-1, \\ 0.5 & \text{if } (i,j) \in \{(1,n),(n,1)\}, \\ 0 & \text{otherwise.} \end{cases}$$

The size of the problem is $n = 10,000$. We compute the four smallest eigenpairs with block size $s = 4$ and maximum subspace size $m = 20$. For both algorithms, the inner systems are solved approximately by the CG scheme. The eigenpairs are accepted when the relative residuals are less than 10^{-10}.

In Table 3, we list the number of outer iterations, the number of matrix vector multiplications with A, and the execution time (in seconds) as functions of the number of inner iteration steps. We see that the performance of both algorithms are very close if the inner systems are solved crudely. The difference becomes clear when we increase the number of inner iteration steps. The dynamic shifting strategy accelerates the algorithm significantly. When the number of inner iteration steps reaches 40, the number of outer iterations is almost half that of the Ritz shifting strategy. When the number of inner iteration steps reaches 80, the number of outer iterations starts increasing for the block Jacobi–Davidson algorithm, but continues to decrease for the Davidson-type trace minimization algorithm. There are plenty of examples for which the block Jacobi–Davidson algorithm actually converges to wrong eigenpairs when the inner systems are solved to high accuracy. The last row of the table shows the result with the dynamic stopping strategy, where the maximum number of inner iteration steps is set to 120. We see that the dynamic shifting strategy improves the performance of the block Jacobi–Davidson algorithm dramatically. In our experiments, the starting subspaces for both algorithms are identical and were chosen randomly. The results clearly show that the success of the block Jacobi–Davidson algorithm depends on good starting spaces.

Table 4
Numerical results for problems from the Harwell–Boeing collection with four processors

Problem	Maximum	Block Jacobi–Davidson			Davidson-type Tracemin		
	inner iterations	#its	A mults	Time	#its	A mults	Time
BCSST08	40	34	3954	4.7	10	759	0.8
BCSST09	40	15	1951	2.2	15	1947	2.2
BCSST11	100	90	30990	40.5	54	20166	22.4
BCSST21	100	40	10712	35.1	39	11220	36.2
BCSST26	100	60	21915	32.2	39	14102	19.6

In Table 4, we show the results obtained for a few generalized eigenvalue problems in the Harwell–Boeing collection. These problems are difficult because the gap ratios for the smallest eigenvalues are extremely small due to the huge span of the spectra. Without preconditioning, none of these problems can be solved with a reasonable cost. In our experiments, we use the incomplete Cholesky factorization (IC(0)) of A as the preconditioner for all the matrices of the form $A-\sigma B$. The Davidson-type trace minimization algorithm works better than the block Jacobi–Davidson algorithm for three of the five problems. For the other two, the performance for both algorithms is similar. Both the shifting strategy and the stopping strategy do not work very well for these two problems because the 2-norms of the residuals are too large to be useful in selecting the shifting parameters.

In all the experiments, for both algorithms, the inner systems are solved by the CG scheme that is terminated when either the specified condition is met or an abnormal case is detected. It is surprising that the CG scheme works well considering that the inner systems for both algorithms are indefinite. The performance with other solvers for the inner systems are similar to that with the CG scheme. For the first problem in Table 4, however, if the inner systems are solved by GMRES(20) with IC(0) pre-conditioning, the block Jacobi–Davidson algorithm returns

84.78615951, 84.78643355, 84.78643355, 85.53681115

while the smallest four eigenvalues are

6.90070261, 18.14202961, 18.14236644, 18.14236645,

which were correctly returned by the Davidson-type trace minimization algorithm using the same inner system solver. This indicates that the Davidson-type trace minimization algorithm is also more robust than the block Jacobi–Davidson algorithm for some problems.

6. Conclusions

In this paper, we presented a comprehensive overview of the trace minimization scheme, its variants, and comparisons with the block Jacobi–Davidson scheme. We demonstrated that, compared to a variant of the trace minimization scheme, the block Jacobi–Davidson algorithm depends more on a good initial subspace due to its choice of the Ritz values as the shift parameters. We showed that the Davidson-type trace minimization scheme can alleviate this dependence by adopting the dynamic shifting strategy and the stopping criterion developed for the original trace minimization algorithm. This variant of the trace minimization algorithm is not only more efficient but also more

robust than the block Jacobi–Davidson algorithm for symmetric generalized eigenvalue problems. Further research is needed, however, on how one can optimally precondition the indefinite systems that arise in both the Davidson-type trace minimization algorithm and the block Jacobi–Davidson algorithm. Our experience indicates that obtaining a positive-definite pre-conditioner for $A - \sigma B$, via an approximate Cholesky factorization that involves boosting of the diagonal elements, is a viable approach.

References

[1] K.J. Bathe, E.L. Wilson, Large eigenvalue problems in dynamic analysis, ASCE, J. Eng. Mech. Div. 98 (1972) 1471–1485.
[2] K.J. Bathe, E.L. Wilson, Solution methods for eigenvalue problems in structural mechanics, Internat. J. Numer. Methods Engrg 6 (1973) 213–226.
[3] F.L. Bauer, Das Verfahren der Treppeniteration und Verwandte Verfahren zur Lösung Algebraischer Eigenwertprobleme, Z. Angew. Math. Phys. 8 (1957) 214–235.
[4] E.F. Beckenbach, R. Bellman, Inequalities, Springer, New York, 1965.
[5] M. Clint, A. Jennings, The evaluation of eigenvalues and eigenvectors of real symmetric matrices by simultaneous iteration, Comput. J. 13 (1970) 76–80.
[6] M. Crouzeix, B. Philippe, M. Sadkane, The Davidson method, SIAM J. Sci. Comput. 15 (1994) 62–76.
[7] J. Cullum, R.A. Willoughby, Lanczos and the computation in specified intervals of the spectrum of large, sparse real symmetric matrices, in: I.S. Duff, G.W. Stewart (Eds.), Sparse Matrix Proceedings 1978, SIAM Publications, Philadelphia, PA, 1979.
[8] J. Cullum, R.A. Willoughby, Computing eigenvalues of very large symmetric matrices — an implementation of a Lanczos algorithm with no reorthogonalization, J. Comput. Phys. 44 (1984) 329–358.
[9] J. Daniel, W.B. Gragg, L. Kaufman, G.W. Stewart, Reorthogonalization and stable algorithms for updating the Gram-Schmidt QR factorization, Math. Comp. 33 (1976) 772–795.
[10] E.R. Davidson, The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices, J. Comput. Phys. 17 (1975) 817–825.
[11] T. Ericsson, A. Ruhe, The spectral transformation Lanczos method for the solution of large sparse generalized symmetric eigenvalue problems, Math. Comp. 35 (1980) 1251–1268.
[12] D.R. Fokkema, G.L.G. Sleijpen, H.A. van der Vorst, Jacobi–Davidson style QR and QZ algorithms for the reduction of matrix pencils, SIAM J. Sci. Comput. 20 (1999) 94–125.
[13] A. Gerschgorin, Über die Abgrenzung der Eigenwerte einer Matrix, Izv. Akad. Nauk SSSR Ser. Fiz.-Mat. 6 (1931) 749–754.
[14] G.H. Golub, R. Underwood, The block Lanczos method for computing eigenvalues, in: J.R. Rice (Ed.), Mathematical Software III, Academic Press, New York, 1977, pp. 361–377.
[15] G.H. Golub, C.F. van Loan, Matrix Computation, 3rd Edition, Johns Hopkins University Press, Baltimore, MD, 1993.
[16] R.G. Grimes, J.G. Lewis, H.D. Simon, A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems, SIAM J. Matrix Anal. Appl. 15 (1994) 228–272.
[17] R.C. Grimm, J.M. Greene, J.L. Johnson, Computation of the magnetohydrodynamic spectrum in axisymmetric toroidal confinement systems, in: Methods of Computational Physics 16, Academic Press, New York, 1976.
[18] W.D. Gropp, E. Lusk, A. Skjellum, Using MPI: Portable Parallel Programming with the Message Passing Interface, MIT Press, Boston, MA, 1994.
[19] R. Gruber, Finite hybrid elements to compute the ideal magnetohydrodynamic spectrum of an axisymmetric plasma, J. Comput. Phys. 26 (1978) 379–389.
[20] C.G.J. Jacobi, Über ein leichtes Verfahren die in der Theorie der Säculärstörungen vorkommenden Gleichungen numerisch aufzulösen, J. Reine Angew. Math 30 (1846) 51–94.
[21] T.Z. Kalamboukis, A Lanczos-type algorithm for the generalized eigenvalue problem $Ax = \lambda Bx$, J. Comput. Phys. 53 (1984) 82–89.

[22] L.V. Kantorovič, Funkcional'nyi analiz i prikladnaja mathematika, Uspekhi Mat. Nauk 3 (1948) 9–185.
[23] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Nat. Bur. Stand. Sect. B 45 (1950) 225–280.
[24] A.A. Levin, On a method for the solution of a partial eigenvalue problem, USSR J. Comput. Math. Math. Phys. 5 (1965) 206–212.
[25] B. Liu, The simultaneous expansion for the solution of several of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices, in: C. Moler, I. Shavitt (Eds.), Numerical Algorithms in Chemistry: Algebraic Method, Lawrence Berkeley Laboratory, University of California, California, 1978, pp. 49–53.
[26] R.B. Morgan, D.S. Scott, Generalizations of Davidson's method for computing eigenvalues of sparse symmetric matrices, SIAM J. Sci. Statist. Comput. 7 (1986) 817–825.
[27] C.C. Paige, The computation of eigenvalues of very large sparse matrices, Ph. D. Thesis, University of London, 1971.
[28] B.N. Parlett, The Symmetric Eigenvalue Problem, Prentice-Hall, Englewood Cliffs, NJ, 1980.
[29] B.N. Parlett, The Symmetric Eigenvalue Problem, SIAM, Philadelphia, 1998.
[30] B.N. Parlett, D.S. Scott, The Lanczos algorithm with selective orthogonalization, Math. Comp. 33 (1979) 217–238.
[31] H. Rutishauser, Computational aspects of F.L. Bauer's simultaneous iteration method, Numer. Math. 13 (1969) 4–13.
[32] H. Rutishauser, Simultaneous iteration method for symmetric matrices, Numer. Math. 13 (1970) 205–223.
[33] Y. Saad, Numerical Methods for Large Eigenvalue Problems, Halsted Press, New York, 1992.
[34] A. Sameh, Z. Tong, Trace minimization and Jacobi–Davidson-type algorithms for large symmetric eigenvalue problems, Tech. Rep. CS-98, Purdue University, West Lafayette, Indiana, 1998.
[35] A. Sameh, J.A. Wisniewski, A trace minimization algorithm for the generalized eigenvalue problem, SIAM J. Numer. Anal. 19 (1982) 1243–1259.
[36] H.D. Simon, The Lanczos algorithm with partial reorthogonalization, Math. Comp. 42 (1984) 115–142.
[37] G.L.G. Sleijpen, A.G.L. Booten, D.R. Fokkema, H.A. van der Vorst, Jacobi–Davidson type methods for generalized eigenproblems and polynomial eigenproblems, BIT 36 (1996) 595–633.
[38] G.L.G. Sleijpen, H.A. van der Vorst, A Jacobi–Davidson iteration method for linear eigenvalue problems, SIAM J. Matrix Anal. Appl. 17 (1996) 401–425.
[39] B.F. Smith, W.D. Gropp, L.C. McInnes, S. Balay, Petsc 2.0 users manual, Tech. Rep. ANL-95/11, Argonne National Laboratory, Chicago, IL, 1995.
[40] A. Stathopoulos, Y. Saad, C.F. Fischer, Robust preconditioning of large, sparse, symmetric eigenvalue problems, J. Comput. Appl. Math. (1995) 197–215.
[41] G.W. Stewart, Accelerating the orthogonal iteration for the eigenvalues of a hermitian matrix, Numer. Math. 13 (1969) 362–376.
[42] G.W. Stewart, A bibliographical tour of the large, sparse generalized eigenvalue problems, in: J.R. Banch, D.J. Rose (Eds.), Sparse Matrix Computations, Academic Press, New York, 1976, pp. 113–130.
[43] H.A. van der Vorst, G.H. Golub, 150 years old and still alive: Eigenproblems, in: I.S. Duff, G.A. Watson (Eds.), The State of the Art in Numerical Analysis, Clarendon Press, Oxford, 1997, pp. 93–119.
[44] K. Wu, Preconditioning techniques for large eigenvalue problems, Ph. D. Thesis, University of Minnesota, Minneapolis, MN, 1997.