

## ANALYSIS OF PROJECTION METHODS FOR SOLVING LINEAR SYSTEMS WITH MULTIPLE RIGHT-HAND SIDES\*

TONY F. CHAN<sup>†</sup> AND W. L. WAN<sup>†</sup>

**Abstract.** We analyze a class of Krylov projection methods but mainly concentrate on a specific conjugate gradient (CG) implementation by Smith, Peterson, and Mittra [*IEEE Transactions on Antennas and Propagation*, 37 (1989), pp. 1490–1493] to solve the linear system  $AX = B$ , where  $A$  is symmetric positive definite and  $B$  is a multiple of right-hand sides. This method generates a Krylov subspace from a set of direction vectors obtained by solving one of the systems, called the seed system, by the CG method and then projects the residuals of other systems orthogonally onto the generated Krylov subspace to get the approximate solutions. The whole process is repeated with another unsolved system as a seed until all the systems are solved. We observe in practice a superconvergence behavior of the CG process of the seed system when compared with the usual CG process. We also observe that only a small number of restarts is required to solve all the systems if the right-hand sides are close to each other. These two features together make the method particularly effective. In this paper, we give theoretical proof to justify these observations. Furthermore, we combine the advantages of this method and the block CG method and propose a block extension of this single seed method.

**Key words.** linear systems, multiple right-hand sides, Krylov space, Lanczos algorithm, conjugate gradient method

**AMS subject classifications.** 65F10, 65Y20

**PII.** S1064827594273067

**1. Introduction.** We want to solve the following systems:

$$(1) \quad AX = B,$$

where  $A$  is a real symmetric positive definite matrix of order  $n$  and  $B = [b^{(1)} \cdots b^{(N)}]$  is a number of right-hand sides to be solved. If we solve (1) using LU factorization, we first decompose  $A$  at a cost of  $O(n^3)$  operations and then solve for each right-hand side at a cost of  $O(n^2)$ . So, for direct methods, the main cost is the decomposition of  $A$  and each right-hand side is solved efficiently by making use of this decomposition. However, for iterative methods like CG, we may have to solve the  $N$  right-hand sides independently. This process can be speeded up if we can find good initial guesses for the unsolved systems and find some efficient refinement process to correct the initial guesses.

If the right-hand sides are arbitrary, we have nearly no hope of doing that. For example, if the right-hand sides are eigenvectors, there is hardly any information shareable among the right-hand sides when the systems are solved by the CG. Yet in practice, for example, in the wave-scattering problem [18], time-marching methods for PDEs [2], or structural mechanics problems [1], the right-hand sides are not arbitrary.

---

\*Received by the editors August 19, 1994; accepted for publication (in revised form) February 20, 1996. This research was partially supported by Office of Naval Research grant N00014-92-J-1890, National Science Foundation grant ASC92-01266, and Army Research Office grant DAAL03-91-G-150.

<http://www.siam.org/journals/sisc/18-6/27306.html>

<sup>†</sup>Department of Mathematics, University of California at Los Angeles, Los Angeles, CA 90024 (chan@math.ucla.edu, wlwan@math.ucla.edu).

They are usually discrete values of some function  $b(t)$ . In other words,

$$b^{(j)} = b(t_j), \quad j = 1, \dots, N.$$

Assume that  $b(t) \in C^k$  and let  $x^{(j)}$  be the solution of  $Ax^{(j)} = b^{(j)}$ . Then  $x^{(j)}$  can be shown to be discrete values of some function  $x(t) \in C^k$ . So, if the  $b^{(j)}$ 's are close, we may also expect the  $x^{(j)}$ 's to be close. For these situations there are several approaches to solve (1). An obvious approach is to solve the right-hand side one-by-one using the CG method with initial guess given by extrapolation of previous solutions if the right-hand sides are values of a known  $C^k$  function  $b(t)$ . This method is effective only when the  $b^{(j)}$ 's are close, which may impose a severe restriction on the step size of  $t$ . Another approach is to select one seed system and solve it by the Lanczos or the CG method. Then one performs a Galerkin projection of the residuals onto the Krylov subspace generated by the seed to obtain approximate solutions for the unsolved ones. The approximate solutions are then refined by the Lanczos or the CG again with, hopefully, fewer steps. This Lanczos–Galerkin scheme is discussed by Parlett [9] and Saad [12]. In addition, they modify the refinement process so that each refinement continues the previous Lanczos run instead of starting a new Lanczos process each time. However, this method requires a lot of memory storage for the Lanczos vectors.

Based on this projection idea, various approaches have been introduced to get rid of the storage requirement if the right-hand sides are assumed to be available simultaneously. Papadrakakis and Smerou [8] still use the Lanczos vectors to do the projection but derive a recursive update for the solution of each right-hand side so that no storage of the Lanczos vectors is needed. Instead of using the Lanczos vectors, van der Vorst [20] uses the residual vectors generated from CG to do the projection. Smith, Peterson, and Mittra [18] give a simplified CG version by using the direction vectors to do the projection. They also introduce a systematic implementation of the seed method and discuss some seed selection strategies. Similar ideas are also discussed by Smith [17] and Joly [5]. Mathematically, all these methods are equivalent. They only differ in implementation.

We may also use the block CG method, discussed in detail by O'Leary [7], to solve (1). In fact, we need not assume the  $b^{(j)}$ 's to be close in this case. However, if the  $b^{(j)}$ 's are really close, linear dependence may arise among the right-hand sides. This makes the block CG method ineffective and the method may even break down. Special treatment is needed to handle this situation. For example, Nikishin and Yeregin [6] proposed to use variable block CG method, where the current block size can be reduced at each block iteration. Besides, one may also use block Lanczos method [3], but then we have to face the storage problem again.

Simoncini and Gallopoulos [15, 16] combine the idea of the seed method and hybrid techniques to solve (1) when  $A$  is nonsymmetric and they present numerous comparisons with other methods together with comprehensive numerical experiments.

Among all these methods, we observe in practice that the single seed method proposed by Smith, Peterson, and Mittra [18] is a very effective implementation of the projection method. There are several practical advantages for this method. First of all, we observe a superconvergence behavior of the CG process of the seed system when compared with the usual CG process. Another advantage is that if the right-hand sides are close, it usually takes very few restarts to solve all the systems. Moreover, it requires no storage for the Lanczos or the direction vectors doing the projection. These features, especially the first two, make this method very effective.

To the best of our knowledge, no theoretical analysis has been given to explain these phenomena. In this paper, we analyze these two properties and give analytical explanations for them. Furthermore, we combine the advantages of the seed method and the block CG method and propose a block generalization of this single seed method. Instead of selecting a single seed, we select several systems as seed. This block seed method enjoys the fast convergence of the block CG while preserving the basic properties of the single seed method.

In sections 2 and 3, we describe the algorithm of the single seed method and present an analysis of it. In sections 4 and 5, we describe the algorithm of the block seed method and give analogous results for it. Numerical experiments are given in section 6 to justify the analysis in sections 3 and 5. Finally, conclusions are made in section 7.

Throughout the paper we adopt the following notations. The eigenvalues and the normalized eigenvectors of  $A$  are denoted by  $\lambda_i$  and  $z_i$ , respectively, and  $0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ . The acute angle between the vectors  $a$  and  $b$  is denoted by  $\angle(a, b)$  while the acute angle between the vector  $a$  and the vector subspace  $V$  is denoted by  $\angle(a, V)$ . We also denote  $O(h^s)$  to be a scalar, a vector, or a matrix whose norm is of order  $O(h^s)$  for convenience.

In this paper, we restrict our attention to symmetric positive definite  $A$ 's.

**2. Single seed method.** The idea of the single seed method is that for each restart a seed system is selected from the unsolved one and is then solved by the CG method. Meanwhile, an approximate solution of the nonseed system is obtained from the space of direction vectors  $p_i$  such that the residual is minimized in the  $A$ -norm in the direction  $p_i$ . In other words, the approximate solution is obtained by Galerkin projection of the residuals onto the Krylov subspace generated by the seed system. After the seed system is solved to desired accuracy, a new seed system is selected and the whole procedure is repeated. Details of the algorithm can be found in [18]. We summarize it in the following algorithm.

#### Single Seed Method for Solving Multiple Right-Hand Sides

```

for k=0, 1, 2 ... until all the systems are solved
  Select the k + 1th system as seed
  for i=0, 1, 2, ..., m_{k+1}                                % CG iteration
    for j=k + 1, k + 2, k + 3, ..., N                      % each remaining unsolved RHS
      if j=k + 1 then perform usual CG steps
         $\delta_i^{k,k} = (r_i^{k,k})^T r_i^{k,k} / (r_{i-1}^{k,k})^T r_{i-1}^{k,k}$ 
         $p_i^{k,k} = r_i^{k,k} + \delta_i^{k,k} p_{i-1}^{k,k}$ 
         $\sigma_i^{k,k} = (r_i^{k,k})^T r_i^{k,k} / (p_i^{k,k})^T A p_i^{k,k}$ 
         $x_{i+1}^{k,k} = x_i^{k,k} + \sigma_i^{k,k} p_i^{k,k}$ 
         $r_{i+1}^{k,k} = r_i^{k,k} - \sigma_i^{k,k} A p_i^{k,k}$ 
      else perform Galerkin projection
         $\eta_i^{k,j} = (p_i^{k,k})^T r_i^{k,j} / (p_i^{k,k})^T A p_i^{k,k}$ 
         $x_{i+1}^{k,j} = x_i^{k,j} + \eta_i^{k,j} p_i^{k,k}$ 
         $r_{i+1}^{k,j} = r_i^{k,j} - \eta_i^{k,j} A p_i^{k,k}$ 
      end if
    end for
  end for
end for
end for

```

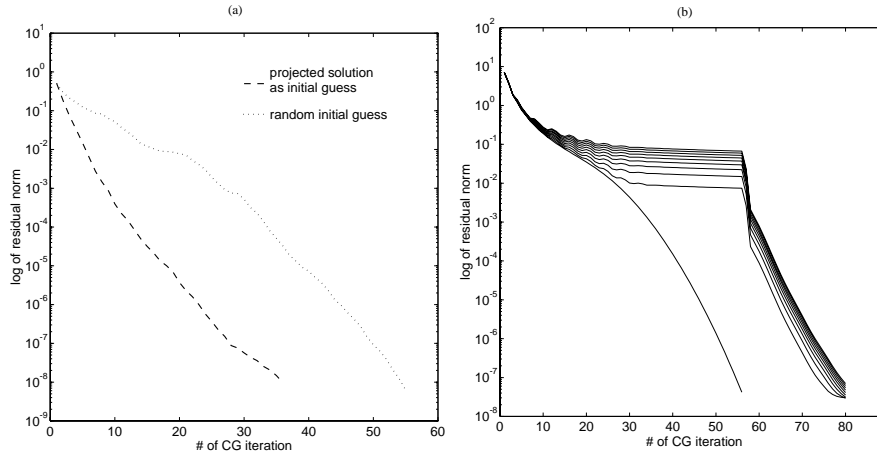


FIG. 1. (a) The convergence behaviors of the CG process with projected solution as initial guess and with random vector as initial guess. (b) The convergence behaviors of all the systems when the right-hand sides are the cyclic rotation of its elements.

We now make a few remarks about the algorithm. We assume, without loss of generality, that the  $k + 1$ st system is the seed for the  $k$ th restart. We use the first superscripts to denote the  $k$ th restart, the second superscripts to denote the  $j$ th system, and the subscripts to denote the  $i$ th step of the CG method. We also assume that  $m_k$  steps are used to solve the  $k$ th system within some given tolerance. These notations are used throughout the following sections.

What is so special about this method is the interesting phenomena shown in Fig. 1. We apply the above algorithm to solve  $Ax = b^{(1)}$  and  $Ax = b^{(2)}$ , where  $A = \text{diag}(1, \dots, 100)$  and  $b^{(1)}, b^{(2)}$  are just some random vectors. Suppose the first system is solved and an approximate solution for the second system is computed by the Galerkin projection onto the previous Krylov subspace. The result of using this projected solution as initial guess and that of using a random initial guess having the same residual norm is shown in Fig. 1(a). We see from the plot that the one using the projected solution as the initial guess converged much faster than the other. This is the superconvergence phenomenon that we referred to earlier. Essentially, the reason is that the projection process kills off the extreme eigenvector components of the initial error. More precisely, we know that the Ritz values approach the extreme eigenvalues rapidly in a few steps of the Lanczos process [10]. Usually the Krylov subspace generated by the first few steps also contains the extreme eigenvectors well. As a result, after the Galerkin projection, the effective spread of the spectrum of  $A$  is narrower which in turn increases the rate of convergence. We shall prove this claim in section 3.2.

Another special phenomenon is the small number of restarts needed to solve all the systems. Let us solve the linear systems with  $A$  as before and  $b_i^{(j)}(t) = \sin(t + (i + j - 2)\Delta t)$ ,  $i = 1, \dots, 100$ , and  $j = 1, \dots, 10$ , where  $\Delta t = 2\pi/100$ . So  $b^{(j)}$  is just obtained by shifting the components of  $b^{(j-1)}$  by one position and the first component is replaced by the last one. The result is shown in Fig. 1(b). The single seed method only needs one restart to solve all the systems. The explanation is that if  $B$  is rank  $k$ , then the single seed method only needs about  $k - 1$  restarts to solve all the systems. In fact, one can easily prove that the right-hand sides  $B$  generated by cyclic rotation of the components is only rank 2. We shall prove this claim in section 3.3.

**3. Analysis of the single seed method.** In this section, we analyze two properties of the single seed method. The first one is to prove the superconvergence of the seed system in our previous example. Before we go on to our analysis, we state some facts about the Lanczos algorithm which will be used in the following sections. Details of the following can be found in [4].

**3.1. Lanczos connection.** Suppose we solve  $Ax = b$  by the Lanczos algorithm. Let the columns of  $V_i = [v_1 \cdots v_i]$  be the orthonormal Lanczos vectors of the  $i$ -dim Krylov subspace  $K_i$  generated by  $i$  steps of the algorithm. Then we have the following well-known recurrence:

$$(2) \quad AV_m = V_m T_m + \beta_{m+1} v_{m+1} e_m^T, \quad m = 1, \dots, n-1,$$

where  $T_m = V_m^T A V_m$  is a tridiagonal matrix,  $e_m$  is the  $m$ th column of the identity matrix, and  $\beta_{m+1}$  is a scalar. Moreover, the solution at step  $m$  is given by

$$x_m = x_0 + V_m T_m^{-1} V_m^T r_0,$$

where  $x_0$  is the initial guess and  $r_0 = b - Ax_0$ . Suppose we have another system:  $A\tilde{x} = \tilde{b}$ . Then the solution obtained from projection onto  $K_m$  is given by

$$(3) \quad \tilde{x}_m = \tilde{x}_0 + V_m T_m^{-1} V_m^T \tilde{r}_0,$$

where  $\tilde{x}_0$  is some initial guess and  $\tilde{r}_0 = \tilde{b} - A\tilde{x}_0$  (see [12] for proof). The following is a useful lemma for our later analysis.

LEMMA 3.1. *The projected solution of the nonseed systems given by the formula in the single seed algorithm is*

$$x_i^{k,j} = x_0^{k,j} + V_i^k (T_i^k)^{-1} (V_i^k)^T r_0^{k,j},$$

where  $V_i^k$  are the Lanczos vectors generated by  $i$  steps of the Lanczos algorithm if the  $k$ th system was solved by the Lanczos algorithm instead.

*Proof.* Smith, Peterson, and Mitra [18] proved that the formula given in the algorithm computes the projected solution in the subspace generated by the direction vectors  $\{p_i^{k,k}\}$ . But this subspace is exactly the subspace spanned by the columns of  $V_i^k$  [4]. So, the result follows immediately from the formula (3).  $\square$

In the following sections, the superscript  $k$  for  $V_i^k$  and  $T_i^k$  is assumed to be the restart number and so it is dropped.

**3.2. Rate of convergence of the seed system.** In this section, we shall prove that if the previous Krylov subspace contains the extreme eigenvectors well, then the initial convergence rate of the seed is increased in a way as if the extreme ends of the spectrum of  $A$  are cut off. The technique of the proof is similar to that in [19] which estimates the rate of convergence of the CG process. We will compare the error reduction after  $i$  steps of the CG process with the error reduction of another CG process starting with an initial error vector obtained from the error vector of the original process by deleting the first  $l$  eigenvector components.

*Remark.* Saad also discussed the convergence of the seed system but from a different point of view (see [12] for details).

LEMMA 3.2. *Consider only the first two systems:  $Ax^{(1)} = b^{(1)}$  and  $Ax^{(2)} = b^{(2)}$ . Suppose the first system is solved in  $m_1$  CG steps. Let  $x_0^{1,2}$  be the solution obtained by the projection of  $b^{(2)}$  on  $K_{m_1}$  and let  $\tilde{x}_0^{1,2}$  be such that  $x^{(2)} - \tilde{x}_0^{1,2}$  is the projection*

of  $x^{(2)} - x_0^{1,2}$  on  $\text{span}\{z_k : k \in I\}^\perp$ , where  $z_k$  is the unit eigenvector corresponding to the eigenvalue  $\lambda_k$  and  $I = \{1, \dots, l\}$ . Let  $\bar{x}_i^{1,2}$  be the  $i$ th iterate of the CG process for  $Ax^{(2)} = b^{(2)}$  with  $\bar{x}_0^{1,2}$  as initial guess. Then, for any  $i$ , we have

$$(4) \quad \|x^{(2)} - x_i^{1,2}\|_A^2 \leq \|x^{(2)} - \bar{x}_i^{1,2}\|_A^2 + \delta_1,$$

where

$$\delta_1 = \|P_{m_1}^\perp x^{(2)}\|^2 \sum_{k \in I} \lambda_k \bar{p}_i^2(\lambda_k) \sin^2 \angle(z_k, K_{m_1}),$$

$P_{m_1}^\perp \equiv I - V_{m_1} T_{m_1}^{-1} V_{m_1}^T A$  is the  $A$ -orthogonal projection onto  $K_{m_1}^\perp$ , and  $\bar{p}_i$  is a polynomial of degree at most  $i$  and constant term 1.

*Proof.* Using  $x_0^{0,2} = 0$  as the initial guess for the second system, by Lemma 3.1 we get

$$x_0^{1,2} \equiv x_{m_1}^{0,2} = V_{m_1} T_{m_1}^{-1} V_{m_1}^T b^{(2)}.$$

Since  $x^{(2)}$  solves the second system  $Ax^{(2)} = b^{(2)}$ , we have

$$(5) \quad \begin{aligned} x^{(2)} - x_0^{1,2} &= (I - V_{m_1} T_{m_1}^{-1} V_{m_1}^T A)x^{(2)} \\ &= P_{m_1}^\perp x^{(2)}. \end{aligned}$$

Let the eigendecomposition of  $x^{(2)} - x_0^{1,2}$  be

$$(6) \quad x^{(2)} - x_0^{1,2} = \sum_{k=1}^n \phi_k z_k.$$

Therefore, by the definition of  $\bar{x}_0^{1,2}$

$$x^{(2)} - \bar{x}_0^{1,2} = \sum_{k=l+1}^n \phi_k z_k.$$

It is well known [4] that there exists a polynomial  $\bar{p}_i(t)$  of degree at most  $i$  and constant term 1 such that

$$\begin{aligned} x^{(2)} - \bar{x}_i^{1,2} &= \bar{p}_i(A)(x^{(2)} - \bar{x}_0^{1,2}) \\ &= \sum_{k=l+1}^n \bar{p}_i(\lambda_k) \phi_k z_k. \end{aligned}$$

This implies that

$$\|x^{(2)} - \bar{x}_i^{1,2}\|_A^2 = \sum_{k=l+1}^n \bar{p}_i^2(\lambda_k) \phi_k^2 \lambda_k.$$

Since  $\bar{p}_i(t)$  is a polynomial of degree at most  $i$ , by the minimization property of the CG iterate we have

$$(7) \quad \|x^{(2)} - x_i^{1,2}\|_A^2 = \min_{p_i} \|p_i(A)(x^{(2)} - x_0^{1,2})\|_A^2$$

$$\begin{aligned}
 (8) \quad & \leq \|\bar{p}_i(A)(x^{(2)} - x_0^{1,2})\|_A^2 \\
 & = \left\| \sum_{k=1}^n \bar{p}_i(\lambda_k) \phi_k z_k \right\|_A^2 \\
 & = \sum_{k=1}^n \bar{p}_k^2(\lambda_k) \phi_k^2 \lambda_k \\
 & = \sum_{k=l+1}^n \bar{p}_i^2(\lambda_k) \phi_k^2 \lambda_k + \sum_{k \in I} \bar{p}_i^2(\lambda_k) \phi_k^2 \lambda_k \\
 & = \|x^{(2)} - \bar{x}_i^{1,2}\|_A^2 + \sum_{k \in I} \bar{p}_i^2(\lambda_k) \phi_k^2 \lambda_k.
 \end{aligned}$$

From (5) and (6) we can calculate  $\phi_k$  to be

$$\begin{aligned}
 |\phi_k| & = |(P_{m_1}^\perp x^{(2)}) \cdot z_k| \\
 & = \|P_{m_1}^\perp x^{(2)}\| \|z_k\| |\cos \angle(z_k, P_{m_1}^\perp x^{(2)})| \\
 & \leq \|P_{m_1}^\perp x^{(2)}\| |\cos \angle(z_k, K_{m_1}^\perp)| \\
 & = \|P_{m_1}^\perp x^{(2)}\| |\sin \angle(z_k, K_{m_1})|.
 \end{aligned}$$

Substituting into (8), we obtain inequality (4).  $\square$

Lemma 3.2 tells us that if the last term of (4) is small, then the convergence rate will be increased in a way as if the first  $l$  eigenvalues of  $A$  have been removed. Since the Lanczos process captures the extreme eigenspaces, the Galerkin projection will kill off the extreme eigenvector components. So we may expect  $\angle(z_k, K_{m_1})$  to be small for  $k$  close to one. In fact, we can bound  $\sin \angle(z_k, K_{m_1})$  with the help of the estimates given by Parlett [10] and Saad [13].

LEMMA 3.3. *Let  $\theta_k = \angle(b^{(1)}, z_k)$  and  $\tau_k = \frac{\lambda_k - \lambda_{k+1}}{\lambda_{k+1} - \lambda_n}$ ,  $k = 1, \dots, l$ . Then*

$$(9) \quad \sin \angle(z_k, K_{m_1}) \leq \omega_k \tan \theta_k,$$

where

$$\omega_k = \frac{1}{T_{m_1-k}(1 + 2\tau_k)} \prod_{p=1}^{k-1} \frac{\lambda_p - \lambda_n}{\lambda_p - \lambda_k}, \quad k = 1, \dots, l,$$

and  $T_q(x)$  is the Chebyshev polynomial of degree  $q$ .

*Proof.* Since

$$\sin \angle(z_k, K_{m_1}) \leq \tan \angle(z_k, K_{m_1}),$$

the result follows from Theorem 12-4-1 in [10] and Theorem 6.3 in [13].  $\square$

First observe that  $\theta_k$  is fixed. So, Lemma 3.3 tells us that if  $\omega_k$  is small, then  $\sin \angle(z_k, K_{m_1})$  is small. If we assume the eigenvalues of  $A$  are distinct, then  $\tau_k > 0$ . So,  $T_{m_1-k}(1 + 2\tau_k)$  and  $T_{m_1+k-n+1}(1 + 2\tau_k)$  grow exponentially. This implies  $\omega_k$  decays exponentially if  $m_1$  is large enough. That means  $\sin \angle(z_k, K_{m_1})$  is very small for sufficiently large  $m_1$ .

Using these two lemmas, we can now show that the bound for the convergence rate is in fact the classical CG bound but with the extreme ends of the spectrum of  $A$  being cut off, plus a small perturbation term.

THEOREM 3.4. *The bound for the A-norm of the error vector after i steps of the CG process is given by*

$$\|x^{(2)} - x_i^{1,2}\|_A^2 \leq 4\|x^{(2)} - \bar{x}_0^{1,2}\|_A^2 \left(\frac{\sqrt{\kappa_R} - 1}{\sqrt{\kappa_R} + 1}\right)^{2i} + \delta_2,$$

where

$$\delta_2 = \|P_{m_1}^\perp x^{(2)}\|^2 \sum_{k \in I} \lambda_k \omega_k^2 \tan^2 \theta_k,$$

and the bound for  $\delta_2$  is independent of the number of iterations  $i$ . Here,  $\kappa_R = \frac{\lambda_n}{\lambda_{l+1}}$  is the reduced condition number.

*Proof.* From Lemma 3.2 we have the following inequality:

$$(10) \quad \begin{aligned} \|x^{(2)} - x_i^{1,2}\|_A^2 &\leq \|x^{(2)} - \bar{x}_i^{1,2}\|_A^2 + \delta_1, \\ \delta_1 &= \|P_{m_1}^\perp x^{(2)}\|^2 \sum_{k \in I} \lambda_k \bar{p}_i^2(\lambda_k) \sin^2 \angle(z_k, K_{m_1}). \end{aligned}$$

The term  $\|x^{(2)} - \bar{x}_i^{1,2}\|_A^2$  can be bounded by the classical CG error estimate

$$(11) \quad \|x^{(2)} - \bar{x}_i^{1,2}\|_A^2 \leq 4\|x^{(2)} - \bar{x}_0^{1,2}\|_A^2 \left(\frac{\sqrt{\kappa_R} - 1}{\sqrt{\kappa_R} + 1}\right)^{2i}.$$

Now, we analyze the  $\delta_1$  term. First, from Lemma 3.3 we can bound  $\delta_1$  by

$$(12) \quad \begin{aligned} \delta_1 &= \|P_{m_1}^\perp x^{(2)}\|^2 \sum_{k \in I} \lambda_k \bar{p}_i^2(\lambda_k) \sin^2 \angle(z_k, K_{m_1}) \\ &\leq \|P_{m_1}^\perp x^{(2)}\|^2 \sum_{k \in I} \lambda_k \bar{p}_i^2(\lambda_k) \omega_k^2 \tan^2 \theta_k \\ &\leq \|P_{m_1}^\perp x^{(2)}\|^2 \max_{k \in I} \bar{p}_i^2(\lambda_k) \sum_{k \in I} \lambda_k \omega_k^2 \tan^2 \theta_k. \end{aligned}$$

Now, we bound the value of  $\max_{k \in I} \bar{p}_i^2(\lambda_k)$ . Recall that  $\bar{p}_i$  is the optimal CG polynomial when solving  $Ax^{(2)} = b^{(2)}$  using  $\bar{x}_0^{1,2}$  as initial guess (cf. proof of Lemma 3.2). It can be shown that  $\bar{p}_i(t)$  has the following form [19, property 2.8]:

$$\bar{p}_i(t) = \frac{(\bar{\theta}_1^{(i)} - t) \cdots (\bar{\theta}_i^{(i)} - t)}{\bar{\theta}_1^{(i)} \cdots \bar{\theta}_i^{(i)}},$$

where  $\bar{\theta}_1^{(i)} < \bar{\theta}_2^{(i)} < \cdots < \bar{\theta}_i^{(i)}$  are the Ritz values of  $A$  with respect to  $K_i$ , the Krylov subspace generated by  $\bar{r}_0^{1,2} = b - A\bar{x}_0^{1,2}$ . By property of Ritz values,  $\lambda_{l+1} \leq \bar{\theta}_j^{(i)} \leq \lambda_n$ ,  $j = 1, \dots, i$ . From the formula of  $\bar{p}_i(t)$  above, for  $0 < t < \lambda_{l+1}$ , we have  $0 < \bar{p}_i(t) < 1$ . Thus,  $\max_{k \in I} \bar{p}_i^2(\lambda_k) < 1$ . The desired inequality follows from substituting (11) and (12) into (10).  $\square$

Hence, if the Krylov subspace of the previous seed system contains the extreme eigenvectors well, then  $\omega_k$  is small. On the other hand, the quantity  $\|P_{m_1}^\perp x^{(2)}\|$  is bounded by  $\|x^{(2)}\|$ . (In fact, if we make additional assumptions on  $b^{(1)}$  and  $b^{(2)}$ , then we can estimate the bound for  $\|P_{m_1}^\perp x^{(2)}\|$ . This is discussed in section 3.4). So, the whole perturbation term is small which implies the initial rate of convergence can be



as fast as if the extreme ends of the spectrum of  $A$  are being cut off. This explains the superconvergence behavior shown in Fig. 1(a).

*Remark.* For the proof of the lemmas, we have made no assumptions about the right-hand sides. So the theorem is still true for arbitrary right-hand sides. This property suggests that the single seed method may be useful in solving general multiple right-hand sides.

**3.3. Rate of convergence of the nonseed system.** Now we come to our second analysis. We prove that the systems are almost solved after  $k - 1$  restarts of the CG process if the rank of  $B$  is  $k$ . This result is the central idea of the following analysis that explains why the single seed method only needs a few restarts to solve all the systems if the right-hand sides are close to each other.

**THEOREM 3.5.** *Suppose  $B = [b^{(1)} \dots b^{(N)}]$  and  $\text{rank}(B) = k < N$ . Then there exists  $\alpha > 0$ , independent of the iteration numbers  $m_p$ , such that the residual of the nonseed systems after  $k - 1$  restarts of the single seed method satisfies*

$$(13) \quad \|\tilde{r}_0^{k,j}\| \leq \alpha \sum_{p=1}^k |\beta_{m_p+1}|, \quad j = k + 1, \dots, N,$$

where  $\beta_{m_p+1}$  comes from the Lanczos recurrence formula (2) if the  $p$ th seed is thought of solved by the Lanczos algorithm,  $p = 1, \dots, k$ .

*Proof.* Let  $B = \sum_{i=1}^k \sigma_i u_i w_i^T$  be the outer product form of a singular value decomposition of  $B$ , where  $\sigma_i$ ,  $u_i$ , and  $w_i$  are the singular values and left and right singular vectors of  $B$ , respectively. So the  $j$ th system  $b^{(j)}$  can be written as

$$(14) \quad b^{(j)} = \sum_{i=1}^k \sigma_i (w_i^T e_j) u_i,$$

where  $e_j$  is the  $j$ th column of the identity matrix  $I$ . Let

$$(15) \quad \begin{aligned} \tilde{r}_0^{0,j} &= b^{(j)}, & \tilde{x}_0^{0,j} &= 0, & j &= 1, 2, \dots, N, \\ \zeta_i^{0,j} &= \sigma_i w_i^T e_j, & u_i^0 &= u_i, & i &= 1, 2, \dots, k. \end{aligned}$$

Notice that

$$(16) \quad |\zeta_i^{0,j}| \leq \sigma_i, \quad j = 1, \dots, N.$$

We take the first system as seed and suppose that it is solved in  $m_1$  CG steps. By Lemma 3.1, the approximate solution of the nonseed systems obtained from the Galerkin projection is

$$\tilde{x}_0^{1,j} = \tilde{x}_0^{0,j} + V_{m_1} T_{m_1}^{-1} V_{m_1}^T \tilde{r}_0^{0,j}, \quad j = 1, \dots, N,$$

and

$$(17) \quad \begin{aligned} \tilde{r}_0^{1,j} &= b^{(j)} - A \tilde{x}_0^{1,j} \\ &= (I - AV_{m_1} T_{m_1}^{-1} V_{m_1}^T) \tilde{r}_0^{0,j} \\ &= (I - V_{m_1} V_{m_1}^T - \beta_{m_1+1} v_{m_1+1} e_{m_1}^T T_{m_1}^{-1} V_{m_1}^T) \tilde{r}_0^{0,j} \\ &= (I - V_{m_1} V_{m_1}^T) \tilde{r}_0^{0,j} - \beta_{m_1+1} v_{m_1+1} e_{m_1}^T T_{m_1}^{-1} V_{m_1}^T \tilde{r}_0^{0,j}. \end{aligned}$$

We want to express  $\tilde{r}_0^{0,j}$  in terms of  $\tilde{r}_0^{0,1}$  since we know that

$$(I - V_{m_1} V_{m_1}^T) \tilde{r}_0^{0,1} = 0.$$

Without loss of generality, we may assume that

$$(18) \quad |\zeta_1^{0,1}| = \max_{1 \leq i \leq k} |\zeta_i^{0,1}|.$$

From (14) and (16) we write  $\tilde{r}_0^{0,j}$  as

$$\tilde{r}_0^{0,j} = \zeta_1^{0,j} u_1^0 + \sum_{i=2}^k \zeta_i^{0,j} u_i^0.$$

In particular, we have

$$\tilde{r}_0^{0,1} = \zeta_1^{0,1} u_1^0 + \sum_{i=2}^k \zeta_i^{0,1} u_i^0.$$

Now, we rearrange the terms of  $\tilde{r}_0^{0,j}$  and rewrite it in terms of  $\tilde{r}_0^{0,1}$ :

$$(19) \quad \tilde{r}_0^{0,j} = \zeta_1^{0,j} (\zeta_1^{0,1})^{-1} \tilde{r}_0^{0,1} + \sum_{i=2}^k [\zeta_i^{0,j} - \zeta_1^{0,j} (\zeta_1^{0,1})^{-1} \zeta_i^{0,1}] u_i^0.$$

Let

$$\zeta_i^{1,j} = \zeta_i^{0,j} - \zeta_1^{0,j} (\zeta_1^{0,1})^{-1} \zeta_i^{0,1}, \quad i = 2, \dots, k.$$

Then, by (16) and (18) we have

$$|\zeta_i^{1,j}| \leq |\zeta_i^{0,j}| + |\zeta_i^{0,j}| |\zeta_i^{0,1} / \zeta_1^{0,1}|, \\ \leq 2\sigma_i.$$

Now, from (19) we get

$$(20) \quad \tilde{r}_0^{0,j} = \zeta_1^{0,j} (\zeta_1^{0,1})^{-1} \tilde{r}_0^{0,1} + \sum_{i=2}^k \zeta_i^{1,j} u_i^0.$$

Combining (17) and (20) we have

$$\tilde{r}_0^{1,j} = (I - V_{m_1} V_{m_1}^T) \sum_{i=2}^k \zeta_i^{1,j} u_i^0 + \gamma_1^{1,j} v_{m_1+1},$$

where

$$\gamma_1^{1,j} = -\beta_{m_1+1} e_{m_1}^T T_{m_1}^{-1} V_{m_1}^T \tilde{r}_0^{0,j} \\ = \alpha_1 \beta_{m_1+1},$$

where

$$\alpha_1 \equiv -e_{m_1}^T T_{m_1}^{-1} V_{m_1}^T \tilde{r}_0^{0,j}.$$

We now show that  $\alpha_1$  is independent of  $m_1$ . Obviously,  $e_{m_1}^T$  is bounded by one. Besides, it is well known [4] from the interlacing property of the eigenvalues of the consecutive  $T_k$ 's that the spectrums of  $T_k$ 's are bounded by the spectrum of  $A$ . Thus,

$$\|T_{m_1}^{-1}\| \leq \|A^{-1}\|.$$

Next, because the columns of  $V_{m_1}$  are orthogonal we have

$$\|V_{m_1}\| \leq 1.$$

Finally, it is clear that

$$\|\tilde{r}_0^{0,j}\| \leq \|B\|.$$

So,  $\alpha_1$  is bounded independent of  $m_1$ . In other words, it does not grow unboundedly as the iteration goes on.

Let

$$u_i^1 = (I - V_{m_1}V_{m_1}^T)u_i^0.$$

Then we further simplify  $\tilde{r}_0^{1,j}$  as

$$(21) \quad \tilde{r}_0^{1,j} = \sum_{i=2}^k \zeta_i^{1,j} u_i^1 + \gamma_1^{1,j} v_{m_1+1}.$$

So after one solve, the rank of the remaining columns of  $B$  is effectively reduced by one if  $\gamma_1^{1,j}$  is small. We proceed in the same manner for the other seeds. Inductively, after  $l - 1$  restarts the residual for the nonseed system is

$$(22) \quad \tilde{r}_0^{l,j} = \sum_{i=l+1}^k \zeta_i^{l,j} u_i^l + \sum_{p=1}^l \gamma_p^{l,j} \prod_{q=p+1}^l (I - V_{m_q}V_{m_q}^T) v_{m_p+1},$$

where

$$\begin{aligned} \zeta_i^{l,j} &= \zeta_i^{l-1,j} - \zeta_l^{l-1,j} (\zeta_l^{l-1,l})^{-1} \zeta_i^{l-1,l}, \\ u_i^l &= (I - V_{m_l}V_{m_l}^T)u_i^{l-1}, \\ \gamma_p^{l,j} &= \gamma_p^{l-1,j} - \zeta_l^{l-1,j} (\zeta_l^{l-1,l})^{-1} \gamma_l^{l-1,j}, \quad p = 1, \dots, l-1, \\ \gamma_l^{l,j} &= -\beta_{m_l+1} e_{m_l}^T T_{m_l}^{-1} V_{m_l}^T \tilde{r}_0^{l-1,j}. \end{aligned}$$

Notice that  $|\zeta_i^{l,j}| \leq 2^l \sigma_i$  since  $|\zeta_i^{l-1,j}| \leq 2^{l-1} \sigma_i$ , which can be derived inductively. In particular, when  $l = k$  we have from (22) that the first term vanishes and

$$(23) \quad \tilde{r}_0^{k,j} = \sum_{p=1}^k \gamma_p^{k,j} \prod_{q=p+1}^k (I - V_{m_q}V_{m_q}^T) v_{m_p+1},$$

where  $\gamma_p^{k,j}$  is a constant multiple of  $\beta_{m_p+1}$ . As explained before, these constants are independent of the  $m_p$ 's. So, we can find  $\alpha > 0$  such that

$$(24) \quad |\gamma_p^{k,j}| \leq \frac{\alpha}{k} |\beta_{m_p+1}|.$$

Moreover, for any  $q$ ,  $I - V_{m_q}V_{m_q}^T$  are projections and  $v_{m_p+1}$  is of unit length. So,

$$(25) \quad \left\| \prod_{q=p+1}^k (I - V_{m_q}V_{m_q}^T)v_{m_p+1} \right\| \leq 1.$$

We obtain (13) from (24), (25), and (23).  $\square$

If  $B$  has near-rank deficiency, then we have a similar result but with an extra perturbation term.

COROLLARY 3.6. *Suppose the singular values of  $B$  are such that*

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k > \epsilon \geq \sigma_{k+1} \geq \dots \geq \sigma_N \geq 0.$$

*Then there exists  $\alpha' > 0$ , independent of the  $m_p$ 's, so that*

$$\|\tilde{r}_0^{k,j}\| \leq \alpha' \left( \sum_{p=1}^k |\beta_{m_p+1}| + \epsilon \right).$$

*Proof.* By (22) with  $k = N$ ,  $l = k$ , we have

$$(26) \quad \tilde{r}_0^{k,j} = \sum_{i=k+1}^N \zeta_i^{k,j} u_i^k + \sum_{p=1}^k \gamma_p^{k,j} \prod_{q=p+1}^k (I - V_{m_q}V_{m_q}^T)v_{m_p+1}.$$

Since  $|\zeta_i^{k,j}| \leq 2^k \sigma_i$  and  $\sigma_i \leq \epsilon$ ,  $i = k + 1, \dots, N$ , by Theorem 3.5 we get

$$\begin{aligned} \|\tilde{r}_0^{k,j}\| &\leq \sum_{i=k+1}^N |\zeta_i^{k,j}| \|u_i^k\| + \alpha \sum_{p=1}^k |\beta_{m_p+1}| \\ &\leq (N - k)2^k \epsilon + \alpha \sum_{p=1}^k |\beta_{m_p+1}|. \end{aligned}$$

The result follows if we let  $\alpha' = \max(\alpha, (N - k)2^k)$ .  $\square$

*Remarks.*

1. In practice, a stopping criterion of the following form is used:

$$\|r_{m_p}^{p,p}\| \leq \text{tol} \|b^{(p)}\|.$$

It is well known [4] that

$$\|r_{m_p}^{p,p}\| = |\beta_{m_p+1}| |e_{m_p}^T T_{m_p}^{-1} e_1| \|b^{(p)}\|.$$

Hence,  $|\beta_{m_p+1}|$  can be bounded as

$$|\beta_{m_p+1}| |e_{m_p}^T T_{m_p}^{-1} e_1| \leq \text{tol}.$$

If  $|e_{m_p}^T T_{m_p}^{-1} e_1|$  is not small, then  $|\beta_{m_p+1}|$  is the same order of tol. So, Theorem 3.5 really means that the systems would be solved to the order of accuracy of tol in only  $k - 1$  restarts if  $\text{rank}(B)=k$ .

2. Consider the special case that  $B = [b^{(1)} \dots b^{(N)}]$ ,  $b^{(j)} = b(t_j)$ , and

$$b(t) = c_1(t)a_1 + \dots + c_k(t)a_k,$$

where  $c_i(t)$  are some bounded functions, not necessarily continuous, and  $a_i$  are some linearly independent vectors. Then Theorem 3.5 implies that  $B$  is almost solved after  $k - 1$  restarts. In particular, if  $b(t)$  is a polynomial of degree  $k - 1$ , then  $B$  could be solved in about  $k - 1$  restarts.

3. In general, if  $b(t)$  is not a polynomial, Corollary 3.6 ensures that we can still have a similar result for  $b(t)$  a continuous function. We rewrite this remark into the following theorem.

**THEOREM 3.7.** *If  $b(t)$  is a continuous function on  $[t_1, t_N]$ , then given  $\epsilon > 0$ , there exist integers  $k \leq N$  and  $\alpha'$  such that*

$$\|\tilde{r}_0^{k,j}\| \leq \alpha' \left( \sum_{p=1}^k |\beta_{m_p+1}| + \epsilon \right).$$

*Proof.* Since  $b(t)$  is continuous, by the Weierstrass approximation theorem [14] there exists a polynomial  $f(t)$  of degree  $(k - 1)$  such that

$$(27) \quad \|b(t) - f(t)\|_2 < \epsilon/\sqrt{N} \quad \forall t \in [t_1, t_N].$$

Notice that we can always find a polynomial of degree  $(N - 1)$  such that (27) is true since  $B$  has only  $N$  columns. So, we always have  $k \leq N$ . We now estimate the size of  $\sigma_{k+1}$ .

Let

$$\tilde{B} = [f(t_1) \cdots f(t_N)].$$

Then  $\text{rank}(\tilde{B})=k$ . By the minimization property of  $\sigma_{k+1}$  (see [4]) we have

$$\sigma_{k+1} \leq \|B - \tilde{B}\|_2.$$

But by (27) we have  $\|B - \tilde{B}\|_2 \leq \|B - \tilde{B}\|_F \leq \epsilon$ . The result follows from Corollary 3.6.  $\square$

This theorem points out an important property. If  $b(t)$  is continuous and the  $b^{(j)}$ 's are close to each other, we may expect that  $\text{rank}(B) < N$ . That means we can find a low-order polynomial to approximate  $b(t)$ . Theorem 3.7 tells us that the single seed method will automatically exploit this fact. One may suggest using a QR factorization of  $B$  and then solving the linearly independent right-hand sides. But this is only effective when the rank is much smaller than  $N$ . Otherwise it is costly to do the factorization and reconstruction. Furthermore, in practice, it is hard to tell the numerical rank of  $B$ . The usefulness of this method is that we do not need to determine the rank by ourselves and it will automatically take care of it.

The number  $k$  somehow reveals the effective rank of  $B$ . If the  $b^{(j)}$ 's are close to each other, this number is usually much smaller than  $N$ . That is why, in practice, the single seed method only takes a few restarts to solve all the systems.

**3.4. Error of the initial guess.** In this section, we want to estimate the error of the initial guess given by the Galerkin projection, assuming that  $b(t)$  is smooth. Saad [12] also estimated the error of the projected solution but in a more general context. Since the right-hand sides of the nonseed systems differ from that of the seed system by  $O(h)$ , we can show that the initial guess given by projection also has  $O(h)$  error.

**THEOREM 3.8.** *Assume that  $b = b(t)$  and  $b^{(j)} = b(t_1 + (j - 1)h)$ . Suppose we have applied  $m_1$  CG steps to the first system:  $Ax^{(1)} = b^{(1)}$  and obtain  $K_{m_1}$ . Let  $\tilde{x}_0^{1,j}$*

be the projected solution of  $Ax^{(j)} = b^{(j)}$  onto  $K_{m_1}$  and  $x_{m_1}^{0,j}$  be the solution given by applying  $m_1$  CG steps to the  $j$ th system with initial guess zero. Then

$$\|\tilde{x}_0^{1,j} - x_{m_1}^{0,j}\| = O(h).$$

*Proof.* Since  $b(t)$  is smooth, the approximation of  $b^{(j)}$  by  $b^{(1)}$  satisfies

$$b^{(j)} = b^{(1)} + O(h).$$

Then, by Lemma 3.1, the projected solution of  $b^{(j)}$  with zero initial guess is

$$\begin{aligned} \tilde{x}_0^{0,j} &= V_{m_1} T_{m_1}^{-1} V_{m_1}^T b^{(j)} \\ &= x_{m_1}^{0,1} + O(h). \end{aligned}$$

Since  $x_{m_1}^{0,1} - x_{m_1}^{0,j} = O(h)$ , the theorem is proved.  $\square$

This theorem shows that  $\|P_{m_1}^\perp x^{(2)}\|$  in Theorem 3.4 is only  $O(h)$  if the right-hand sides have smooth dependence.

**4. Block seed method.** In this section, we propose a block generalization of the single seed method. In the last section, we note that the initial guess given by the Galerkin projection requires  $h$  to be very small in order to have a good approximation since the approximation is only of first order. So the performance of the single seed method will be improved when the step size of  $b(t)$  is small. Practically, we may need to take large steps which may deteriorate the performance of the single seed method. A natural solution to this problem is to select more than one system as seed so that the subspace generated by the seed is larger and the initial guess obtained from the projection onto this subspace is hopefully better. Moreover, if we use the block CG method to solve the seed systems, the convergence rate is also improved. These statements will be made more precise in the following sections.

The block seed algorithm is exactly the same as the single seed method except the seed is a block of  $s$  systems. The seed is solved by the block CG method [7] while an approximate solution for the nonseed system is obtained by the Galerkin projection onto the the subspace generated by the seed. The algorithm of the block seed method is as follows.

Note that the second superscript for a matrix denotes the  $j$ th subblock of  $B$ .

A drawback of the block method is that it may break down when singularity of the matrices  $(R_{i-1}^{k,k})^T R_{i-1}^{k,k}$  and  $(P_i^{k,k})^T A P_i^{k,k}$  occurs. On the other hand, in addition to the fast convergence inherited from the block CG, the block seed method possesses all the basic properties of the single seed method discussed in previous sections. In fact, the bounds obtained are better than those for the single seed method. Analogous theorems are stated in the following sections.

From now on we adopt additional notations for our discussion of the block seed method. The block size is assumed to be  $s$  and the matrix  $A$  is assumed not to have eigenvalues with multiplicity greater than  $s$ . After  $m$  steps of the block CG, we let  $\mathcal{V}_m = [U_1 \cdots U_m]$  whose columns denote the orthonormal blocks of Lanczos vectors and  $\mathcal{K}_m$  denotes the subspace spanned by columns of  $\mathcal{V}_m$ . Let  $\mathcal{P}_m$  and  $\mathcal{P}_m^\perp$  be the  $A$ -orthogonal projection onto  $\mathcal{K}_m$  and  $\mathcal{K}_m^\perp$ , respectively. Lowercase letters denote vectors while uppercase letters denote matrices.

```

Block Seed Method for Solving Multiple Right-Hand Sides

for k=0, 1, 2 ... until all the systems are solved
  Select the ks + 1, ..., ks + sth systems as seed
  for i=0, 1, 2, ..., m_{k+1} % block CG iteration
    for j=k + 1, k + 2, k + 3, ..., [N/s] % each remaining unsolved block of RHS
      if j=k + 1 then perform usual block CG steps
         $\Delta_i^{k,k} = (R_i^{k,k})^T R_i^{k,k} ((R_{i-1}^{k,k})^T R_{i-1}^{k,k})^{-1}$ 
         $P_i^{k,k} = R_i^{k,k} + \Delta_i^{k,k} P_{i-1}^{k,k}$ 
         $\Sigma_i^{k,k} = (R_i^{k,k})^T R_i^{k,k} ((P_i^{k,k})^T A P_i^{k,k})^{-1}$ 
         $X_{i+1}^{k,k} = X_i^{k,k} + \Sigma_i^{k,k} P_i^{k,k}$ 
         $R_{i+1}^{k,k} = R_i^{k,k} - \Sigma_i^{k,k} P_i^{k,k}$ 
      else perform block Galerkin projection
         $H_i^{k,j} = (P_i^{k,k})^T R_i^{k,j} ((P_i^{k,k})^T A P_i^{k,k})^{-1}$ 
         $X_{i+1}^{k,j} = X_i^{k,j} + H_i^{k,j} P_i^{k,k}$ 
         $R_{i+1}^{k,j} = R_i^{k,j} - H_i^{k,j} A P_i^{k,k}$ 
      end if
    end for
  end for
end for

```

**5. Analysis of the block seed method.** In the following sections, we give the analogous results of the single seed case. Since the techniques of the proofs are similar to the nonblock case, we shall skip them.

**5.1. Rate of convergence of the seed system.** We show below that the bound for the convergence rate of the seed systems is superior to that of the single seed method. This is due to the fact that the block Lanczos method gives a better bound for the extreme eigenvalues than that of the classical Lanczos method. Yet the statement and the proof of the theorem is analogous to Theorem 3.4. Before we go on we state a fact about the minimization property of the block CG iterate proved by O'Leary [7].

**THEOREM 5.1.** *Let  $X_m$  be the approximate solution given by the block CG and  $X^*$  be the true solution. Then  $X_m$  minimizes  $\text{tr}[(X - X^*)^T A (X - X^*)]$  over all  $X$  such that  $X - X_0 \in \mathcal{K}_m$ .*

Since the minimization property of the block CG algorithm is on all the systems together, we can only estimate the convergence rate of the seed systems indirectly. That means we cannot bound the errors of the individual systems except by blocks. Like before, we show the effect of the extreme  $l$  eigenvalues.

**LEMMA 5.2.** *Consider two systems:  $AX^{(1)} = B^{(1)}$  and  $AX^{(2)} = B^{(2)}$ . Suppose the first systems are solved to desired accuracy after  $m_1$  steps of the block CG process. Let  $X_0^{0,2}$ ,  $\bar{X}_0^{0,2}$ ,  $X_i^{0,2}$ ,  $\bar{X}_i^{0,2}$ , and  $I$  be defined similarly as in Lemma 3.2 and  $x^{(j)}$  be the solution of the  $j$ th system of  $X^{(2)}$ . Then for any  $i$  we have*

$$(28) \quad \text{tr}[(X^{(2)} - X_i^{1,2})^T A (X^{(2)} - X_i^{1,2})] \leq \text{tr}[(X^{(2)} - \bar{X}_i^{1,2})^T A (X^{(2)} - \bar{X}_i^{1,2})] + \delta_3,$$

where

$$\delta_3 = \sum_{j=1}^s \|\mathcal{P}_{m_1}^\perp x^{(j)}\|^2 \sum_{k \in I} \lambda_k \bar{p}_j^2(\lambda_k) \sin^2 \angle(z_k, \mathcal{K}_{m_1}).$$

We can also estimate  $\sin \angle(z_k, \mathcal{K}_{m_1})$  by the bounds given by Saad [11].

LEMMA 5.3. *Suppose that  $\mathcal{P}_1 z_j$  are linearly independent for  $j = k, \dots, k + s - 1$ ,  $k \leq l$ . Let  $\hat{x}_k \in \mathcal{K}_1$  be defined such that*

$$z_j^T \hat{x}_k = \delta_{kj}$$

for  $j = k, \dots, k + s - 1$ ,  $k \leq l$ . Let  $\hat{\theta}_k = \angle(z_k, \hat{x}_k)$  and  $\hat{\tau}_k = \frac{\lambda_k - \lambda_{k+s}}{\lambda_{k+s} - \lambda_n}$ ,  $k = 1, \dots, l$ . Then

$$(29) \quad \sin \angle(z_k, \mathcal{K}_{m_1}) \leq \hat{\omega}_k \tan \hat{\theta}_k,$$

where

$$\hat{\omega}_k = \frac{1}{T_{m-k}(1 + 2\hat{\tau}_k)} \prod_{\lambda \in \Lambda_k} \frac{\lambda - \lambda_n}{\lambda - \lambda_k}, \quad k = 1, \dots, l.$$

Here,  $\Lambda_k$  is the set of the first  $k - 1$  distinct eigenvalues.

Remark. Since  $|\lambda_k - \lambda_{k+s}| \geq |\lambda_k - \lambda_{k+1}|$ , we have  $\hat{\tau}_k \geq \tau_k$ . This implies  $\hat{\omega}_k \leq \omega_k$ . Besides, the subspace  $\mathcal{V}_{m_1}$  is bigger than the subspace  $V_{m_1}$ , and so we may expect  $\|\mathcal{P}_{m_1}^\perp x^{(j)}\| \leq \|P_{m_1}^\perp x^{(j)}\|$ , too (cf. Lemmas 3.2 and 3.3).

By these two lemmas we may give an indirect bound similar to Theorem 3.4 for the convergence rate of the seed systems.

THEOREM 5.4. *The bound for the sum of the error vector in the A-norm after  $i$  steps of the block CG process is given by*

$$\text{tr}[(X^{(2)} - X_i^{1,2})^T A (X^{(2)} - X_i^{1,2})] \leq 4 \sum_{j=1}^s \rho_j \left( \frac{\sqrt{\kappa_R} - 1}{\sqrt{\kappa_R} + 1} \right)^{2i} + \delta_4,$$

where

$$\delta_4 = \sum_{j=1}^s \|\mathcal{P}_{m_1}^\perp x^{(j)}\|^2 \sum_{k \in I} \lambda_k \hat{\omega}_k^2 \tan^2 \hat{\theta}_k$$

for some constant  $\rho_j$ .

This bound looks almost the same as the bound in Theorem 3.4 except  $\omega_k$  and  $\theta_k$  change to  $\hat{\omega}_k$  and  $\hat{\theta}_k$ . But from the previous remark  $\hat{\omega}_k \leq \omega_k$  and  $\|\mathcal{P}_{m_1}^\perp x^{(j)}\| \leq \|P_{m_1}^\perp x^{(j)}\|$ . This shows that the bound for the convergence rate of the seed systems in the block case is better than that in the single seed case.

**5.2. Rate of convergence of the nonseed system.** We now state two other analogous theorems of the block seed method. It says that all the systems are almost solved in  $\lceil k/s \rceil - 1$  restart of the block seed method if  $\text{rank}(B) = k$ .

THEOREM 5.5. *Suppose we want to solve the same problem as in Theorem 3.5 by the block seed method. Then there exists  $\alpha > 0$ , independent of the  $m_p$ 's, such that the residual of the nonseed systems after  $\lceil k/s \rceil - 1$  restarts satisfies*

$$\|\tilde{r}_0^{\lceil k/s \rceil, j}\| \leq \alpha \sum_{p=1}^{\lceil k/s \rceil} \|\hat{\beta}_{m_p+1}\|, \quad j = k + 1, \dots, N,$$

where  $\hat{\beta}_{m_p+1}$  comes from the recurrence of the block Lanczos algorithm [3],

$$A\mathcal{V}_{m_p} = \mathcal{V}_{m_p} \mathcal{T}_{m_p} + U_{m_p+1} \hat{\beta}_{m_p+1} E_{m_p}^T, \quad p = 1, \dots, \lceil k/s \rceil.$$



Finally, if  $b(t)$  is continuous, we can bound the residual in the same way as in Theorem 3.7.

**THEOREM 5.6.** *If  $b(t)$  is a continuous function in  $t$  on  $[t_1, t_N]$ , then given  $\epsilon > 0$ , there exist integers  $k$  and  $\alpha'$  such that the residual of the nonseed systems, after  $\lceil k/s \rceil - 1$  restart of the block seed method, satisfies*

$$\|\tilde{r}_0^{\lceil k/s \rceil, j}\| \leq \alpha' \left( \sum_{p=1}^{\lceil k/s \rceil} \|\hat{\beta}_{m_p+1}\| + \epsilon \right).$$

The results of Theorems 5.5 and 5.6 are not surprising since each solve of the block CG generates a subspace  $s$  times bigger than CG does. So the number of restarts should be reduced by a factor of  $s$  for the block seed method.

**5.3. Error of the initial guess.** We point out in section 4 that the initial guess given by the single seed method is good only if the right-hand sides are really close. In fact, the initial guess is only  $O(h)$  accurate to the  $m$ th iterate given by the CG process. We now prove that the initial guess given by the block seed method is  $O(h^s)$  accurate to the extrapolate approximation of  $x^{(j)}$  given by the solutions of the seed systems.

**THEOREM 5.7.** *Assume  $b^{(j)} = b(t_1 + (j - 1)h)$  as in Theorem 3.8. Suppose we have applied  $m_1$  block CG steps to the seed systems:  $AX^{(1)} = B^{(1)}$  and obtain  $\mathcal{K}_{m_1}$ . Let  $\tilde{x}_0^{1,j}$  be the projected solution of  $Ax^{(j)} = b^{(j)}$  onto  $\mathcal{K}_{m_1}$ . Then  $\tilde{x}_0^{1,j}$  is an  $s$ th-order approximation of the extrapolation of the columns of  $X_{m_1}^{0,1}$  which are obtained from applying  $m_1$  block CG steps to the first subblock of  $B$ .*

*Proof.* Since  $b(t)$  is smooth, the extrapolate approximation of  $b^{(j)}$  by  $b^{(i)}$ 's,  $i = 1, \dots, s$ , is of order  $s$ . So, there exist  $\chi_i$  such that

$$b^{(j)} = \sum_{i=1}^s \chi_i b^{(i)} + O(h^s).$$

Then, by the block version of Lemma 3.1 the projected solution of  $b^{(j)}$  with initial guess zero is

$$\begin{aligned} \tilde{x}_0^{1,j} &= \mathcal{V}_{m_1} \mathcal{T}_{m_1}^{-1} \mathcal{V}_{m_1}^T b^{(j)} \\ &= \sum_{i=1}^s \chi_i \mathcal{V}_{m_1} \mathcal{T}_{m_1}^{-1} \mathcal{V}_{m_1}^T b^{(i)} + O(h^s) \\ &= \sum_{i=1}^s \chi_i x_{m_1}^{(i)} + O(h^s). \end{aligned}$$

The result follows from the fact that  $\sum_{i=1}^s \chi_i x_{m_1}^{(i)}$  is actually the extrapolated approximation given by  $X_{m_1}^{0,1}$ .  $\square$

Higher order of accuracy means the size of  $h$  need not be too small in order to get a good initial guess. This suggests that the block seed method is useful in the case when the right-hand sides are not close to each other.

**6. Numerical experiments.** In this section, we present four examples to illustrate the analysis in sections 3 and 5 and also compare the performance of these methods applied to an acoustic scattering problem.

All the experiments were performed in MATLAB with machine precision  $10^{-16}$ . The stopping criterion is  $\|r_i^{k,j}\| < \text{tol} * \|b^{(j)}\|$ , where the tolerance,  $\text{tol}$ , is set to be  $10^{-8}$ .

*Example 1.* In this example, we show the effective cutoff of the spectrum of  $A$  and the convergence rate of the seed systems. More precisely, we are going to verify the analysis of Theorem 3.4 which tells us that the convergence of the CG process using projected solution as the initial guess behaves like that of using the same initial guess but with the extreme  $l$  eigenvector components of the initial error being cut off. Meanwhile, we also verify Theorem 5.4, the block counterpart of Theorem 3.4. In order to show that this property does not depend on the smoothness of  $b$ , we simply use random vectors for  $B$ . Our test problem is

$$AX = B,$$

where  $A = \text{diag}(1, 2, \dots, 100)$  and  $B$  is a random matrix with four columns.

Let  $b_j$  denote the  $j$ th column of  $B$ . For the single seed method we solve  $b_1$  by the CG method (in 55 iterations) and project the other three systems to obtain an initial guess. Then each one is solved by the CG again. We only consider the behavior of the last system  $b_4$ . Before discussing the convergence behavior, let us see what has happened to  $b_4$  after projection. Figure 2(a) shows the components of  $b_4$  before the projection while Fig. 2(b) shows the situation after the projection. Figure 2(b) clearly reveals that the first couple of eigenvector components of  $b_4$  are killed off by the projection. This illustrates that the effective spectrum of  $A$  is narrower than before which in turn increases the convergence rate. In fact, the result is shown in Fig. 3(a). The solid line denotes the convergence behavior of the last system using the projected solution as initial guess. The dotted line denotes the convergence behavior of the same system using the same projected solution as initial guess but whose error vector is such that the extreme  $l$  eigenvector components are cut off. We choose  $l = 5$  because Fig. 2(b) shows that the first five eigenvector components have essentially gone. Figure 3(a) indeed shows that the convergence behavior of both initial guesses are almost identical as predicted by Theorem 3.4.

For the block seed method we solve  $b_1$  and  $b_2$  by the block CG method (in 40 iterations) and project the other two systems to obtain an initial guess. Then the last two are solved by the block CG again. Again, we only concentrate on the last system. The situation is similar to the single seed case. Let us first see the effect of the block projection of  $b_4$ . Figure 2(c) shows that more eigenvector components are killed off by the block projection. This implies more extreme eigenspaces are being captured by one solve of the block CG than by the usual CG. The effect of this on the convergence behavior is shown in Fig. 3(b). It shows that the convergence behavior of the last system is actually slightly better than the case  $l=5$  and almost the same as the case  $l=10$ . This verifies that the block seed method has a better convergence bound than the single seed method as stated in the discussion after Theorem 5.4.

*Example 2.* In this example, we show the convergence rate of the nonseed systems. Our test problem is

$$Ax(t) = b(t),$$

where  $A = \text{diag}(1, 2, \dots, 100)$  and  $b(t) = a_0 + ta_1 + t^2a_2 + t^3a_3$ ;  $a_0, a_1, a_2$ , and  $a_3$  are some linearly independent vectors of unit length. So,  $\text{rank}(B) = 4$  in this case. We choose  $t_1 = 1$ ,  $\Delta t = 0.1$ , and  $t_j = t_1 + (j - 1)\Delta t$ ,  $j = 1, \dots, 10$ , and  $b_j = b(t_j)$ .

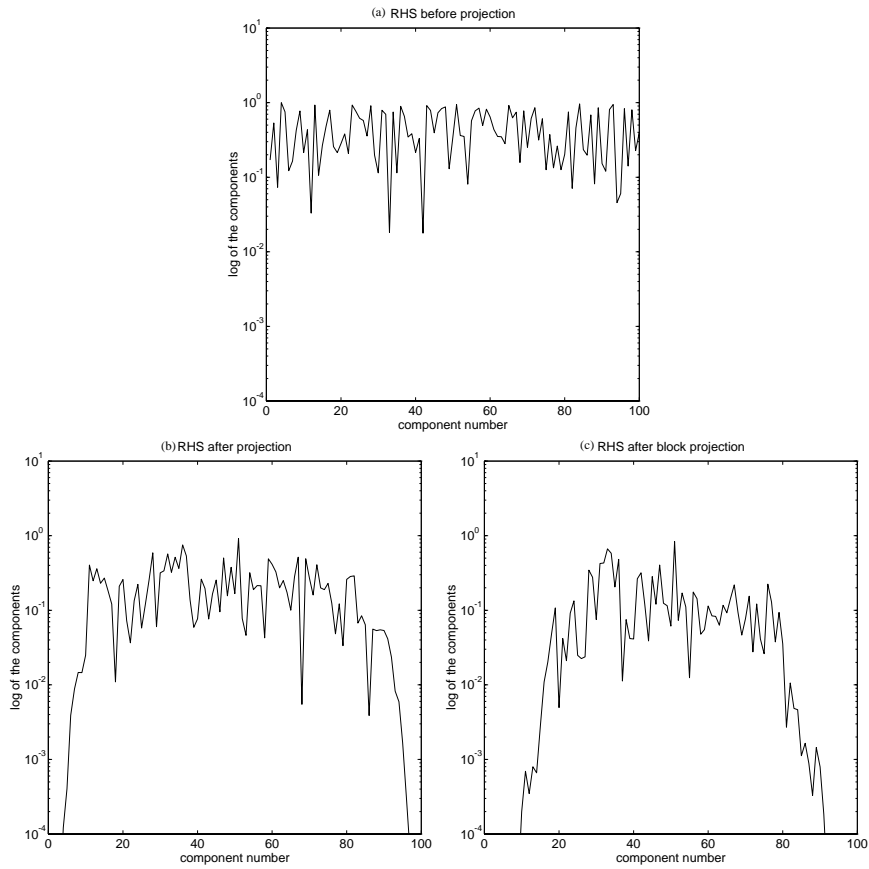


FIG. 2. Size distribution of the components of (a) the original vector  $b_4$ , (b)  $b_4$  after the projection, and (c)  $b_4$  after the block projection.

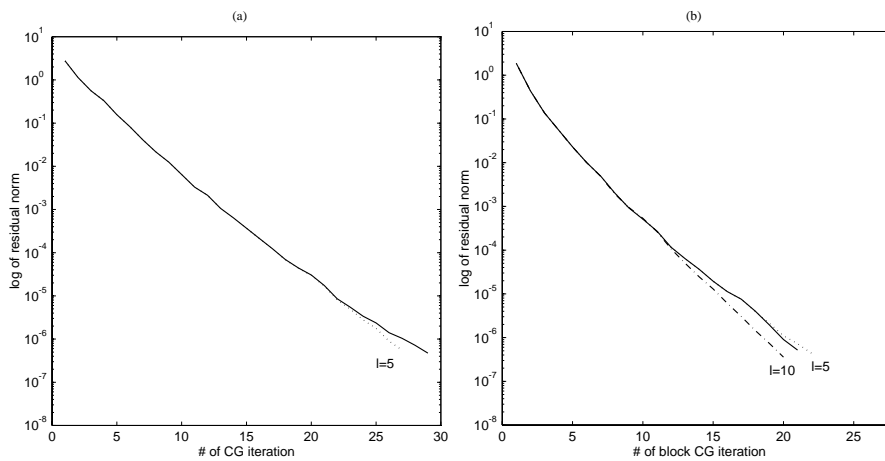


FIG. 3. The convergence behavior of the CG/block CG process with different initial guess. (a) The single seed method, and (b) the block seed method. Solid lines denote seed/block seed method. Dotted lines denote the case  $l = 5$ . Dashed dotted lines denote the case  $l = 10$ .

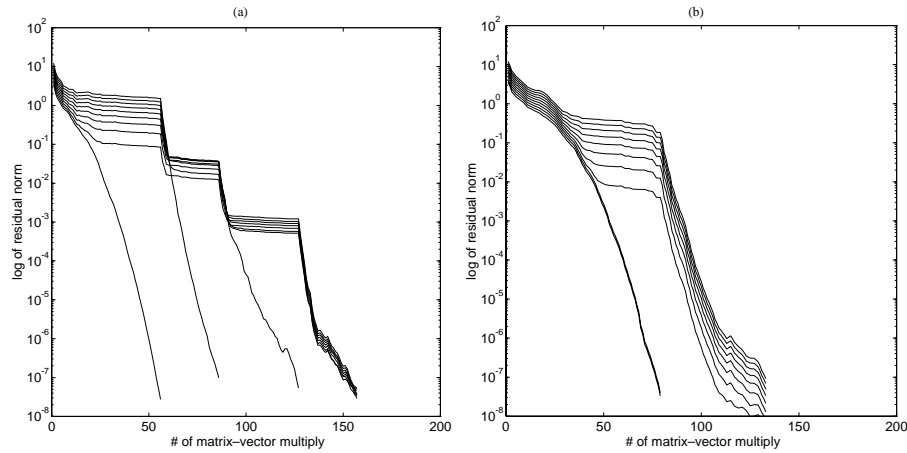


FIG. 4. The convergence behavior of all the systems (10 right-hand sides) when solving  $Ax(t) = b(t)$ , where  $A = \text{diag}(1:100)$  and  $b(t) = a_0 + ta_1 + t^2a_2 + t^3a_3$  by (a) the single seed method, and (b) the block seed method.

Theorems 3.5 and 5.5 tell us that all the systems should be almost solved in three restarts for the single seed method and in one restart for the block seed method. The convergence behaviors of all the systems are shown in Figs. 4(a) and (b). From the plot, each steepest declining line denotes the convergence of a seed and also for the nonseed in the last restart. Figure 4(a) shows that four seeds (corresponding to three restarts) are used to solve all the systems. Figure 4(b) shows that, in the block seed case, only two seeds (corresponding to one restart) are used to solve all the systems. This result exactly matches the prediction given by Theorems 3.5 and 5.5. Note that we plot the residual norm against the cost (the number of matrix–vector multiply) in place of the iteration number as in Example 1 so that we may compare the effectiveness of these methods by the cost. In this case, the block seed method is slightly better.

*Example 3.* In this example, we apply the single seed method and the block seed method to solve an acoustic-scattering problem. The object is assumed to be circular and the wave number is arbitrarily chosen as five. The Helmholtz equation or the reduced wave equation is solved by the double-layer potential, using the trapezoidal rule to discretize the integral. We show the efficiency of both methods and illustrate the effect of  $\Delta t$ , the change of the incident angle  $t$ , on them. More precisely, our test problem is

$$Ax(t) = b(t),$$

where  $A = \tilde{A}\tilde{A}^T$  and  $\tilde{A}$  is a particular nonsingular complex matrix of order  $128 \times 128$  arising from the acoustic-scattering problem.  $b(t)$  is the right-hand side which depends on the incident angle  $t$  and corresponds to the boundary data that accompany the circle geometry. The formula for the  $i$ th component of the right-hand side is

$$b_i(t) = -\cos(5 \cos(t - 2(i - 1)\pi/128)) - \sqrt{-1} \sin(5 \cos(t - 2(i - 1)\pi/128)).$$

We solve the problem in two cases. The first case corresponds to  $\Delta t = 1^\circ$  and the second case corresponds to  $\Delta t = 10^\circ$ . The number of right-hand sides is still 10.

The result of the first case is shown in Figs. 5(a) and (b). In our previous analysis, we learn that the initial guess given by the block seed method is better than that of

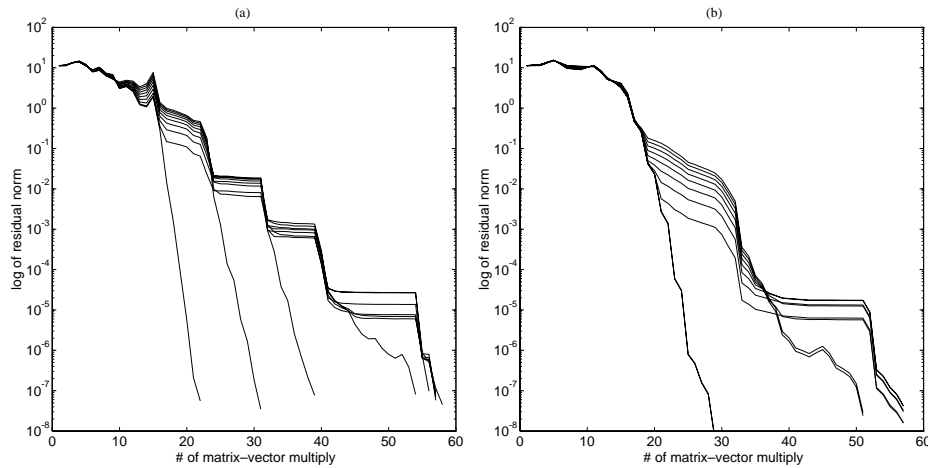


FIG. 5. The convergence behavior of all the systems when solving  $Ax(t) = b(t)$ , where  $b(t)$  corresponds to the boundary data of a radar scattering problem and  $\Delta t = 1^\circ$  by (a) the single seed method, and (b) the block seed method.

the single seed method when  $\Delta t$  small. Figures 5(a) and (b) show that the residual of the initial guess given by the single seed method is  $O(10^{-1})$  while that given by the block seed method is  $O(10^{-2})$ . However, since the right-hand sides are close when  $\Delta t$  is small, the performance of the block seed method is not especially good since some matrices are near singular. In fact, both methods solve the problem using nearly the same cost.

In Fig. 5(a), we can easily see that the single seed method only takes six CG runs to solve all the systems. Actually, the first four CG runs nearly solve all the systems. The last two are just minor corrections of the unsolved systems. In fact, the first four singular values of  $B$  are 35.24, 6.18, 0.37, and 0.02 and the remaining ones are only  $O(10^{-4})$  or less. So, this experiment verifies the discussion following Theorem 3.7 that the single (and the block) seed method will automatically exploit the rank deficiency of  $B$  and save a lot of CG solves.

For the second case the result is shown in Figs. 6(a) and (b). Since  $\Delta t$  is large, we can expect that the initial guesses given by the single seed method are not good. Also, because  $\Delta t$  is large,  $b^{(j)}$ 's are not close to each other and  $B$  has full rank. These unfavorable conditions weaken the performance of the single seed method. But for the block seed method, it uses two instead of one right-hand sides to generate a subspace in each restart, so the subspace is larger, which is good for the convergence of the seed systems as well as for reducing the residual of the nonseed systems by projection. So Figs. 6(a) and (b) clearly show that the block seed method performs much better in this case. This example illustrates that the single seed method depends heavily on the closeness of the right-hand sides while the block seed method depends less.

*Example 4.* In this example, we show the performance of the block seed method with block size  $s = 3, 4$ . We use the same matrices  $A$  and  $B$  as in example 3 and choose  $\Delta t = 10^\circ$ . Figures 7(a) and (b) show the result of  $s = 3$  and 4, respectively. When compared to Figs. 6(a) and (b), we see that the convergence rate of the seeds for  $s = 3, 4$  is better than for  $s = 1, 2$ . Actually, we can see a trend of increasing convergence rate with increasing  $s$ . But the problem for large block size is that near singularity of some matrices may produce unstable results. In fact, we can see from Figs. 7(a) and (b) that some systems did not converge. From our experience,  $s = 2$

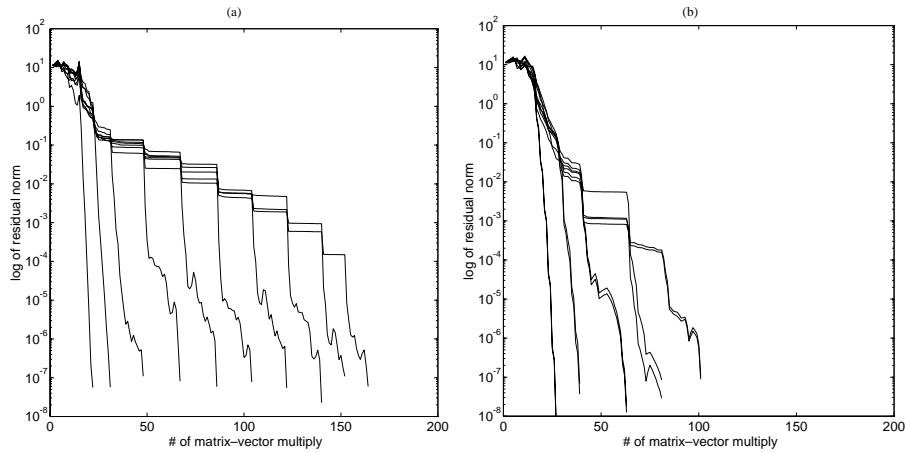


FIG. 6. The convergence behavior of all the systems (10 right-hand sides) when solving  $Ax(t) = b(t)$ , where  $b(t)$  corresponds to the boundary data of a radar scattering problem and  $\Delta t = 10^\circ$  by (a) the single seed method, and (b) the block seed method.

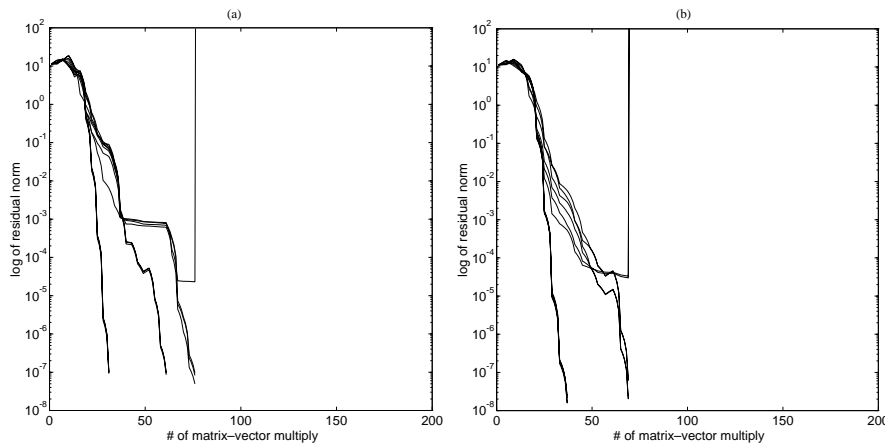


FIG. 7. The convergence behavior of all the systems when solving  $Ax(t) = b(t)$ , where  $b(t)$  corresponds to the boundary data of a radar-scattering problem and  $\Delta t = 10^\circ$  by the block seed method with (a)  $s = 3$ , and (b)  $s = 4$ .

is an optimal choice. Although it still has the chance of near singularity, it is rarely the case when compared with the other choices of  $s$ .

**7. Conclusion.** We have claimed that the single seed method possesses two merits that make it an effective method to solve multiple right-hand sides with smooth dependence on  $t$ . One is the superconvergence of the seed system and the other is that it only takes a few number of restarts to solve all the systems. These two properties are essentially derived from the intrinsic property of the Galerkin projection. The single seed method automatically makes use of this property and improves the efficiency of solving multiple right-hand sides. We have proved these two properties analytically and also estimated the initial error of the projected solution. We have also performed numerical experiments to justify the theoretical analysis and to show that the single seed method is effective to solve this problem when the right-hand sides are close to each other.

On the other hand, we have also noted the weakness of the single seed method when the right-hand sides are not close to each other. We therefore proposed a block generalization of the single seed method. The basic idea of this block seed method is to exploit a larger subspace by the block seed systems. We have proved that the block seed method possesses the same properties of the single seed method and found that the bounds are better than those for the single seed method. We have estimated the initial errors as well. However, we have noted that the block seed method inherits both the advantages and disadvantages of the block CG method. Actually, we have shown numerically that the performance of the block seed method is very good when the right-hand sides are not close to each other but not that good in the opposite case. In fact, in that case, the subspace generated by the block seed systems is no better than the subspace generated by a single seed. Nevertheless, the block seed method is shown to be an attractive alternative to the single seed method when the right-hand sides are not close to each other.

**Acknowledgments.** We would like to thank G. Golub and D. Boley for their helpful comments on Theorem 3.5. The first author would also like to acknowledge the support and hospitality of the Computer Science Department of The Chinese University of Hong Kong; this work was initiated during a visit there.

## REFERENCES

- [1] C. FARHAT AND F. X. ROUX, *Implicit Parallel Processing in Structural Mechanics*, Tech. report CU-CSSC-93-26, Center for Aerospace Structures, University of Colorado, Boulder, CO, 1993.
- [2] P. F. FISHER, *Projection Techniques for Iterative Solution of  $Ax = b$  with Successive Right-Hand Sides*, Tech. report 93-90, ICASE, Hampton, VA, 1993.
- [3] G. H. GOLUB AND R. UNDERWOOD, *The block Lanczos method for computing eigenvalues*, in *Mathematical Software III*, J. R. Rice, ed., New York, Academic Press, 1977, pp. 361–377.
- [4] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, 1989.
- [5] P. JOLY, *Résolution de systèmes linéaires avec plusieurs seconds membres par la méthode du gradient conjugué*, Tech. report R-91012, Publications du Laboratoire d'Analyse Numérique, Université Pierre et Marie Curise, Paris, 1991.
- [6] A. A. NIKISHIN AND A. Y. YEREMIN, *Variable Block CG Algorithms for Solving Large Sparse Symmetric Positive Definite Linear Systems on Parallel Computers, I: General Iterative Scheme*, Tech. report EM-RR 1/92, Elegant Mathematics, Inc. (USA), Seattle, WA, 1992.
- [7] D. P. O'LEARY, *The block conjugate gradient algorithm and related methods*, *Linear Algebra Appl.*, 29 (1980), pp. 292–322.
- [8] M. PAPADRAKAKIS AND S. SMEROU, *A new implementation of the Lanczos method in linear problems*, *Internat. J. Numer. Methods Engrg.*, 29 (1990), pp. 141–159.
- [9] B. N. PARLETT, *A new look at the Lanczos algorithm for solving symmetric systems of linear equations*, *Linear Algebra Appl.*, 29 (1980), pp. 323–346.
- [10] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice–Hall, Englewood Cliffs, NJ, 1980.
- [11] Y. SAAD, *On the rates of convergence of the Lanczos and the block-Lanczos methods*, *SIAM J. Numer. Anal.*, 17 (1980), pp. 687–706.
- [12] Y. SAAD, *On the lanczos method for solving symmetric linear systems with several right-hand sides*, *Math. Comp.*, 48 (1987), pp. 651–662.
- [13] Y. SAAD, *Numerical Methods for Large Eigenvalue Problems*, Manchester University Press, Manchester, UK, 1992.
- [14] G. F. SIMMONS, *Introduction to Topology and Modern Analysis*, McGraw–Hill, New York, 1963.
- [15] V. SIMONCINI AND E. GALLOPOULOS, *A memory-conserving hybrid method for solving linear systems with multiple right-hand sides*, in *Proc. Copper Mountain Conf. Iterative Methods*, Copper Mountain, CO, April 1992, Tech. report 1203, Center for Supercomputing Research and Development, University of Illinois at Urbana–Champaign, October, 1993.

- [16] V. SIMONCINI AND E. GALLOPOULOS, *An iterative method for nonsymmetric systems with multiple right-hand sides*, SIAM J. Sci. Comput., 16 (1995), pp. 917–933.
- [17] C. F. SMITH, *The Performance of Preconditioned Iterative Methods in Computational Electromagnetics*, Ph.D. thesis, Department of Electrical Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 1987.
- [18] C. F. SMITH, A. F. PETERSON, AND R. MITTRA, *A conjugate gradient algorithm for the treatment of multiple incident electromagnetic fields*, IEEE Trans. Antennas and Propagation, 37 (1989), pp. 1490–1493.
- [19] A. VAN DER SLUIS AND H. A. VAN DER VORST, *The rate of convergence of conjugate gradients*, Numer. Math., 48 (1986), pp. 543–560.
- [20] H. A. VAN DER VORST, *An iteration solution method for solving  $f(A)x = b$ , using Krylov subspace information obtained for the symmetric positive definite matrix  $a$* , J. Comput. Appl. Math., 18 (1987), pp. 249–263.