Under review, "SIAM J. Scientific Comuting"

QMR-BASED PROJECTION TECHNIQUES FOR THE SOLUTION OF NON-HERMITIAN SYSTEMS WITH MULTIPLE RIGHT HAND SIDES *

MISHA KILMER†, ERIC MILLER† , AND CAREY RAPPAPORT†

Abstract. In this work we consider the simultaneous solution of large linear systems of the form $Ax^{(j)} = b^{(j)}, j = 1, \ldots, K$ where A is sparse and non-Hermitian. Our single-seed approach uses QMR to solve the seed system j and generate biorthogonal Krylov subspaces. Approximate solutions to the non-seed systems are simulanteously generated by minimizing their appropriately projected residuals. After the initial seed system has converged, the process is repeated by choosing a new seed from among the remaining non-converged systems and using the previously generated approximate solutions as initial guesses for the new seed and non-seed systems. We give theory supporting our observation in practice of super-convergence of (non-initial) seed systems as compared to the usual QMR process. The computational advantage of our method over using QMR to solve each system individually is illustrated on two examples. Finally, we propose a block QMR variant which combines the advantages of this approach and those of the block QMR with deflation scheme of Freund and Malhotra. The computational savings of our block method are shown in examples.

Key words. QMR, projection, Krylov subspace, iterative methods, block Krylov

AMS subject classifications. 65F10, 65N22

Running Title: QMR-based Projection Techniques

1. Introduction. In many applications one desires the solution of multiple linear systems of the form

$$Ax^{(j)} = b^{(j)}, j = 1, \ldots, K. \tag{1}$$

involving the same $N \times N$ coefficient matrix A but K different right hand sides $b^{(j)}$, all of which are available simultaneously. Such problems arise, for instance, in the numerical solution of frequency-domain electromagnetic wave scattering; here, the right hand sides might correspond to incident fields over the scatterer induced either by plane waves at various angles of incidence or by excitation sources at different locations.

Systems involving large, sparse matrices make good candidates for solution by iterative Krylov subspace methods since storage is kept to a minimum and matrix-vector products can be done efficiently. However, the naive approach of solving each of the K linear systems independently using a Krylov subspace method does not take advantage of the fact that the $b^{(j)}$'s, and hence the $x^{(j)}$'s, may be *closely related* due to the underlying physical nature of the problem. By closely related, we mean that the solution to the jth system has large components in the initial directions of the k-dimensional ($k \ll N$) Krylov subspace generated from one of the other systems. Projection-type techniques for both the Hermitian and non-Hermitian cases, discussed in more detail below, that specifically exploit such shared information have been proposed (e.g., see [4, 21, 19] and the references therein).

Another alternative is to use a block Krylov subspace algorithm, to solve the systems simultaneously [12, 18, 3, 7]. Essentially these methods seek solutions in

*This work supported by the Army Research Office Demining MURI under Grant DAAG55-97-1-0013

†Electrical and Computer Engineering Department, Northeastern University, Boston, MA 02115, email: mkilmer@ece.neu.edu, elmiller@cdsp.neu.edu, rappaport@neu.edu

block Krylov subspaces, or some deflated version thereof, generated by the matrix A and the $N \times K$ matrix $R = B - AX_0$; here the columns of B are the $b^{(j)}$ and the columns of X_0 are the initial estimates for each of the systems. However, this approach can be slightly more expensive in terms of storage than projection techniques because the length of the recurrences to update the iterates depends on the number of right hand sides, or, in the case of deflation [7], the number right hand sides corresponding to the deflated Krylov sequences. Also, if a deflation technique is used, a deflation tolerance must be specified in advance, and we have found in experiments that the performance and convergence of the systems depends in a somewhat unpredictable manner on this value. Therefore, we first pursue the idea of a single-seed projection type of technique and then present a block variant of our algorithm that actually exploits the best properties of the block QMR algorithm while preserving the basic properties of our sequential projection type of technique.

Specifically, the idea of a projection technique is to first select one of the systems as "seed" and solve it by an iterative Krylov subspace method. As the relevant subspaces are generated, the approximations to the other systems are simultaneously updated by projecting the residual onto a particular subspace and by either enforcing a Galerkin-type condition [10, 20] or by minimizing the projected residual [19]. Such methods are sometimes referred to as Lanczos-Galerkin approaches [16].

Smith [20] and Joly [10] both consider a projection approach based on BiCG for non-symmetric A. In [10] a similar approach for CGS is also given. However, the BiCG-projection approach can exhibit the potentially erratic convergence behavior observed when applying BiCG to a single linear system (see the results in [19]). Simoncini and Gallopoulos [19] also present an approach to solving (1) when A is nonsymmetric. They use an Arnoldi process to generate a orthogonal basis for some Krylov subspace of dimension, say l, for the residual of the seed system, then project the non-seed residuals onto that space. To improve the approximating among the systems they employ a hybrid method which exploits the generated GMRES residual polynomial.

Our algorithm is similar to the project-minimize approach but has certain advantages over the algorithm in [19]; namely, we do not need to store the basis vectors, we do not need to predetermine a subspace dimension, and the approximate solutions and residuals are cheaply computed and available at every stage of the algorithm because they are updated with short-term recurrences. As noted above, the success of our single-seed method over the approach of solving each system separately depends on the closeness among the right hand sides. Therefore, we also propose an extension of our algorithm, based on the BL-QMR algorithm presented in [7], which is more efficient when the right hand sides are not all close.

This paper is organized as follows. In §2, we give the necessary background on the QMR approach. We give an outline of our approach in §3 and in §4 we give theory supporting our choice of algorithm. A block variant of our QMR-projection algorithm is reported in §5. Section 6 gives numerical results and §7 reports conclusions and future work.

2. The QMR Algorithm. The quasi-minimal residual (QMR) algorithm was introduced by Freund and Nachtigal in [8] for the purpose of solving non-Hermitian linear systems with single right hand sides. The original algorithm was based on three-term recurrences. In [9], the authors propose a mathematically equivalent algorithm which employed coupled two term recurrences. Since the latter variant has been found to be more numerically stable for solving linear systems, in numerical experiments we

use this implementation. However, to simplify the notation in this section and in §3, and to be consistent with the notation in section §5, we will follow the notation in [8]. Further, for simplicity, we consider only the version without lookahead, but note that our algorithm could be adapted to account for lookahead.

In the remainder of the paper, the notation $\|\cdot\|$ always refers to the Euclidean norm $\|\cdot\|_2$. The superscript T is used to denote the transpose operation and superscript $*$ is used to denote the conjugate transpose operation.

A Krylov subspace of dimension k generated by a matrix G and a vector q is defined according to

$$K_k(G,q) = \text{span}\{q, Gq, G^2q, \ldots, G^{k-1}q\}.$$

The QMR algorithm is a Krylov-subspace based iterative method which can be used to solve non-Hermitian linear systems of the form

$$Ax = b, \quad A \in \mathcal{C}^{N\times N}.$$

At the kth iteration, the current solution estimate has the form

$$x_k = x_0 + V_k z_k, \tag{2}$$

where x_0 denotes the initial guess and

$$V_k = [v_1, v_2, \ldots, v_k]$$

is an $N \times k$ matrix whose columns are basis vectors for $K_k(A, v_1)$ with $v_1 = r_0/\|r_0\|$ and $r_0 = b - Ax_0$. The length k vector z_k is the vector of expansion coefficients, and will be chosen as the solution to a particular minimization problem, as discussed below. Those basis vectors are generated via the nonsymmetric Lanczos process (see [17]), and are constructed to be biorthogonal to vectors $w_i, i = 1, \ldots k$ which form a basis for $K_k(A^T, w_1)$[1]. The columns of the $N \times k$ matrix W_k are the w_i.

From biorthogonality it follows that

$$W_{k+1}^T V_{k+1} = D_{k+1}, \;\; D_{k+1} = \text{diag}(\delta_1, \ldots, \delta_{k+1}). \tag{3}$$

Also as a result of the Lanczos algorithm we obtain the relation

$$AV_k = V_{k+1}\bar{T}_k \tag{4}$$

where $\bar{T}_k$ is a $(k+1) \times k$ tridiagonal matrix. Using (2), (3), and (4), and setting $\beta = \|r_0\|$, we have that at the kth iteration the residual, $r_k = b - Ax_k$, is given by [17]

$$\begin{aligned} b - Ax_k &= b - A(x_0 + V_k z_k) \\ &= r_0 - AV_k z_k \\ &= \beta v_1 - V_{k+1}\bar{T}_k z_k \\ &= V_{k+1}(\beta e_1 - \bar{T}_k z_k), \end{aligned} \tag{5}$$

where e_1 denotes the first unit vector. Since the columns of V_{k+1} are not orthonormal, we have

$$\|r_k\| \le \|V_{k+1}\|\|\beta e_1 - \bar{T}_k z_k\|.$$

[1] Here we always take $w_1 \equiv v_1$, but note that other choices are possible. A version of the algorithm is also possible using $K_k(A^*, w)$ for the second Krylov subspace.

The QMR algorithm determines z_k by minimizing the norm of the quasi-residual term; that is,

$$z_k = \arg\min_z \|\beta e_1 - \bar{T}_k z\|.$$

Since $\bar{T}_k$ is tridiagonal $(k+1)\times k$, this problem is readily solved using QR factorization where the orthonormal matrix Q is determined via Givens rotations [8].

We make the following alternate observation. From (5), we have

$$r_k = V_{k+1}(\beta e_1 - \bar{T}_k z_k),$$

and therefore

$$D_{k+1}^{-1} W_{k+1}^T r_k = \underbrace{(D_{k+1}^{-1} W_{k+1}^T V_{k+1})}_{I_{k+1}}(\beta e_1 - \bar{T}_k z_k),$$

where I_{k+1} denotes the identity matrix of dimension $k+1$ which follows from (3). Finally we observe

$$\|D_{k+1}^{-1} W_{k+1}^T r_k\| = \|\beta e_1 - \bar{T}_k z_k\|.$$

Thus, the z_k which defines the kth QMR iterate can also be thought of as the one that minimizes the norm of the residual projected onto a smaller dimensional subspace. We will make use of this alternate definition of the QMR iterates in subsequent sections.

3. The QMR-Projection Algorithm. In this section we describe a single seed QMR-projection algorithm for solving linear systems of the form (1). Our algorithm proceeds as follows. First, we select one system, say system j, to serve as "seed" and apply QMR (without lookahead) to the seed system. In the following, we use $r_0^{j,l}$ to denote the initial residual to system l, where l denotes the number of any of the non-converged systems given the starting guess $x_0^{j,l}$. We use $r_k^{j,l}$ to denote the residual of system l after k iterations. Since different choices of seed lead to different Krylov subspaces and hence different iterates, the superscript j is used to denote that this is the residual at the kth iteration for system l when system j was used as seed. By the beginning of the kth iterate, QMR has generated biorthogonal bases for two k-dimensional Krylov subspaces, $K_k(A, r_0^{j,j})$ and $K_k(A^T, r_0^{j,j})$. We denote the respective bases by the vectors $v_{j,i}$ and $w_{j,i}$, $i = 1, \ldots, k$: the subscript j is used to indicate that this particular set was generated using system j as seed. These vectors are the columns of the $N \times k$ matrices $V_{j,k}$ and $W_{j,k}$, respectively. The corresponding $k+1 \times k$ tridiagonal matrix is denoted as $\bar{T}_{j,k}$ (compare to (4)). By the end of the kth iterate, QMR has also generated the unnormalized versions of the vectors $v_{j,k+1}$ and $w_{j,k+1}$ for use in the $k+1$st iteration.

Let us comment on the values of $x_0^{j,l}$. If we suppose that j was the seed system and converged after m steps and that the index of the next seed has index j^*, then we set $x_0^{j^*,l} = x_m^{j,l}$ for all indecies l such that system l has not already converged.

From the previous section, we have that the kth iterate corresponding to the seed system is given by

$$x_k^{j,j} = x_0^{j,j} + V_{j,k} z_k^{j,j},$$

where

$$z_k^{j,j} = \arg\min_z \|\beta e_1 - \bar{T}_{j,k} z\|.$$

Now we also want the kth iterate of the (non-converged) non-seed system, say system l, to lie in $x_0^{j,l} + K_k(A, r_k^{j,l})$. In other words, we want

$$x_k^{j,l} = x_0^{j,l} + V_{j,k} z_k^{j,l}, \ l \neq j. \tag{6}$$

Next we must decide how we will define $z_k^{j,l}$. Note that

$$\begin{aligned} r_k^{j,l} &= b^{(l)} - A x_k^{j,l} \\ &= b^{(l)} - A(x_0^{j,l} + V_{j,k} z_k^{j,l}) \\ &= r_0^{j,l} - A V_{j,k} z_k^{j,l} \\ &= r_0^{j,l} - V_{j,k+1} \bar{T}_{j,k} z_k^{j,l}. \end{aligned} \tag{7}$$

Therefore, using biorthogonality we have

$$D_{j,k+1}^{-1} W_{j,k+1}^T r_k^{j,l} = D_{j,k+1}^{-1} W_{j,k+1}^T r_0^{j,l} - \bar{T}_{j,k} z_k^{j,l},$$

so that

$$\|D_{j,k+1}^{-1} W_{j,k+1}^T r_k^{j,l}\| = \|D_{j,k+1}^{-1} W_{j,k+1}^T r_0^{j,l} - \bar{T}_{j,k} z_k^{j,l}\|.$$

Finally, we use the above equality to determine $z_k^{j,l}$:

$$z_k^{j,l} = \arg\min_z \|D_{j,k+1}^{-1} W_{j,k+1}^T r_0^{j,l} - \bar{T}_{j,k} z\|. \tag{8}$$

In the following section we will describe how to efficiently solve equation (8) and give short term recurrence updates for $x_k^{j,l}$ and $r_k^{j,l}$.

3.1. Computational Issues. Let us address how to efficiently compute the iterates and residuals of the non-seed systems. As above, we will use the index l to denote an arbitrary non-seed system and j to denote the seed system. We will define $g_{k+1}^{j,l} = D_{j,k+1}^{-1} W_{j,k+1}^T r_0^{j,l}$, where the $k+1$st component of $g_{k+1}^{j,l}$ is the scalar $\gamma_{k+1}^{j,l} = \frac{1}{\delta_{j,k+1}} w_{j,k+1}^T r_0^{j,l}$.

Let the QR decomposition of $\bar{T}_{j,k}$ be

$$\bar{T}_{j,k} = Q_{j,k}^* \begin{bmatrix} R_{j,k} \\ 0 \end{bmatrix}$$

(note that $R_{j,k}$ is $k \times k$ upper triangular with upper bandwidth 2) where, using the notation of [8],

$$Q_{j,k} = G_{j,k} \begin{bmatrix} G_{j,k-1} & 0 \\ 0 & 1 \end{bmatrix} \cdots \begin{bmatrix} G_{j,1} & 0 \\ 0 & I_{k-1} \end{bmatrix},$$

and $G_{j,i}$ has the form

$$G_{j,i} = \begin{bmatrix} I_{i-1} & 0 & 0 \\ 0 & c_{j,i} & s_{j,i} \\ 0 & -\bar{s}_{j,i} & c_{j,i} \end{bmatrix}, \text{ where } c_{j,i} \in \mathcal{R}, s_{j,i} \in \mathcal{C}, c_{j,i}^2 + |s_{j,i}|^2 = 1.$$

Note that the QR decomposition of $\bar{T}_{j,k}$ can be easily and quickly obtained from the QR decomposition of $\bar{T}_{j,k-1}$. We can use this QR factorization of $\bar{T}_{j,k}$ to solve (8):

$$z_k^{j,l} = \arg\min_z \|Q_{j,k} g_k^{j,l} - \begin{bmatrix} R_{j,k} \\ 0 \end{bmatrix} z\|.$$

Defining

$$\begin{bmatrix} t_k^{j,l} \\ \tau_{k+1}^{j,l} \end{bmatrix} = Q_{j,k} g_{k+1}^{j,l},$$

we obtain

$$z_k^{j,l} = R_{j,k}^{-1} t_k^{j,l}. \tag{9}$$

Note that this implies

$$\|D_{j,k+1}^{-1} W_{j,k+1}^T r_0^{j,l} - \bar{T}_{j,k} z_k^{j,l}\| = |\tau_{k+1}^{j,l}|. \tag{10}$$

Next, using $g_{k+1}^{j,l} = [(g_k^{j,l})^T, \gamma_{k+1}^{j,l}]^T$ we observe that

$$\begin{bmatrix} t_k^{j,l} \\ \tau_{k+1}^{j,l} \end{bmatrix} = G_{j,k} \begin{bmatrix} Q_{j,k-1} g_k^{j,l} \\ \gamma_{k+1}^{j,l} \end{bmatrix} = G_{j,k} \begin{bmatrix} t_{k-1}^{j,l} \\ \tau_k^{j,l} \\ \gamma_{k+1}^{j,l} \end{bmatrix}.$$

Hence, by definition of $G_{j,k}$, $t_k^{j,l}$ differs from $t_{k-1}^{j,l}$ only in its last entry, which we shall denote by $y_k^{j,l}$. Further, $\tau_{k+1}^{j,l}$ and $y_k^{j,l}$ can be updated from $\gamma_{k+1}^{j,l}$ and $\tau_k^{j,l}$:

$$\begin{bmatrix} y_k^{j,l} \\ \tau_{k+1}^{j,l} \end{bmatrix} = \begin{bmatrix} c_{j,k} & s_{j,k} \\ -\bar{s}_{j,k} & c_{j,k} \end{bmatrix} \begin{bmatrix} \tau_k^{j,l} \\ \gamma_{k+1}^{j,l} \end{bmatrix}. \tag{11}$$

As in Equation 4.8 of [8], we define $P_{j,k} = [p_{j,1}, p_{j,2}, \ldots, p_{j,k}] \equiv V_{k,j} R_{j,k}^{-1}$. Since $R_{j,k}$ is upper triangular with bandwidth 2, there is a short term recurrence relation for the $p_{j,k}$ [8]. Using (6) and (9), we have that the kth iterate of the lth system is given by

$$x_k^{j,l} = x_{k-1}^{j,l} + y_k^{j,l} p_{j,k}. \tag{12}$$

From this, we derive an iterative update for the $r_k^{j,l}$ that does not require any additional matrix-vector products per iteration as follows.

LEMMA 3.1. *The residual at the kth iteration corresponding to the lth system is given by*

$$r_k^{j,l} = r_{k-1}^{j,l} - y_k^{j,l} f_{j,k} \quad \textit{where} \quad f_{j,k} \equiv A p_{j,k},$$

and can be computed in $O(N)$ flops.

Proof: First, we note that the residual update formula follows directly from (12) and the definition $r_k^{j,l} = b^{(l)} - A x_k^{j,l}$. Now denote the entries in $R_{j,k}$ by

$$R_{j,k} = \begin{bmatrix} \zeta_{j,1} & \varepsilon_{j,2} & \theta_{j,3} & 0 & \ldots & 0 \\ 0 & \zeta_{j,2} & \varepsilon_{j,3} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \zeta_{j,3} & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \theta_{j,k} \\ \vdots & & & \ddots & \ddots & \varepsilon_{j,k} \\ 0 & \ldots & \ldots & \ldots & 0 & \zeta_{j,k} \end{bmatrix}.$$

From $P_{j,k} = V_{j,k} R_{j,k}^{-1}$, it follows (see Section 4 of [8]) that

$$p_{j,k} = \frac{1}{\zeta_{j,k}}(v_{j,k} - \varepsilon_{j,k} p_{j,k-1} - \theta_{j,k} p_{j,k-2}),$$

where it is understood that $p_{j,n} = 0$ for $n \leq 0$. Hence, $f_{j,k}$ is updated iteratively according to

$$f_{j,k} = \frac{1}{\zeta_{j,k}}(m_{j,k} - \varepsilon_{j,k} f_{j,k-1} - \theta_{j,k} f_{j,k-2}),$$

where $m_{j,k} \equiv A v_{j,k}$, $f_{j,n} = 0$ for $n \leq 0$. However, since $m_{j,k}$ has already been computed in the course of the iteration, it need not be recomputed. Therefore, to form $r_k^{j,l}$ only 3 vector sums and 4 scalar-vector products need to performed. Since the lengths of the vectors are N, the result follows. □

3.2. Seed Selection. Clearly, the success of our QMR-projection approach at reducing the total number of matrix-vector products needed to solve all the systems to the desired tolerance depends on which and in what order systems are selected as seed. We use the approach in [19]; namely, we choose the seed index j such that the norm of the residual of the corresponding system is larger than all the remaining non-converged systems. Developing more informed selection heuristics remains a subject for future research.

4. Convergence Theory. Suppose that QMR has been run once and that the initial seed system has converged after m steps. Our algorithm proceeds by choosing another seed and using as its initial guess that solution obtained via projection as the first system was solved. One of the main results of this section is that the rate of convergence of this second seed system behaves as if the extreme ends of the spectrum of A is cut off, provided that the previous Krylov subspace contains the extreme right eigenvectors well. The proof technique follows along the lines of the proof of Lemma 3.2 in [4]. The notation $\kappa_2(\cdot)$ denotes the 2-norm condition number of the argument.

In the following, we assume A is diagonalizable with eigendecomposition $A = Z\Lambda S$ where $S = Z^{-1}$. Here $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$ with $|\lambda_1| \geq |\lambda_2| \geq \ldots |\lambda_N| > 0$. We use s_j^* to denote the jth row of S; that is, s_j is a left eigenvector of A. Without loss of generality, we may assume that we have normalized so that $\|s_j\| = 1$. Under these definitions, we have the following.

THEOREM 4.1. *Consider two systems $Ax^{(1)} = b^{(1)}$ and $Ax^{(2)} = b^{(2)}$. Let $x_0^{2,2}$ be the starting vector for the 2nd system obtained via our projection approach after m steps of QMR has been run using the first system as seed; that is, $x_0^{2,2} = x_m^{1,2}$.*

Define $\mathcal{I} = \{1, \ldots, n\}$, for $n < N$ arbitrary but fixed and

$$Z_n = \mathrm{span}\{z_j : j \in \mathcal{I}\}.$$

Define $\bar{x}_0^{2,2}$ such that $x^{(2)} - \bar{x}_0^{2,2}$ is the projection of $x^{(2)} - x_0^{2,2}$ on $Z_n^{\perp}$. Let $\bar{x}_i^{2,2}$ be the ith iterate of the GMRES applied to system 2 with initial guess $\bar{x}_0^{2,2}$. Then for any i we have

$$\|b^{(2)} - Ax_i^{2,2}\| \leq \kappa_2(V_{2,i+1})(\|b^{(2)} - A\bar{x}_i^{2,2}\| + \delta)$$

where

$$\delta = \sum_{k \in \mathcal{I}} |\tilde{p}(\lambda_k)||z_j^* P_m^{\perp} r_0^{1,2}|$$

$\tilde{p}$ is a particular i-degree polynomial with constant term one, and $P_m^{\perp} \equiv I - AV_{1,m}(\bar{T}_{1,m}^{*}\bar{T}_{1,m})^{-1}\bar{T}_{1,m}D_{1,m+1}^{-1}W_{1,m+1}^{T}$ with $\mathcal{R}(P_m^{\perp}) = (AV_{1,m})^{\perp}$.

Proof: By definition we have $Z_n^{\perp} = \text{span}\{s_j : j \notin \mathcal{I}\}$. Hence

$$b^{(2)} - Ax_0^{2,2} = \sum_{k=1}^{N} \phi_k \lambda_k s_k \tag{13}$$

$$b^{(2)} - A\bar{x}_0^{2,2} = \sum_{k=n+1}^{N} \phi_k \lambda_k s_k \tag{14}$$

for some expansion coefficients ϕ_k. Now if $\bar{x}_i^{2,2}$ is the ith GMRES iterate with $\bar{x}_0^{2,2}$ as the initial guess, there exists a polynomial $\tilde{p}_i$ of degree less than or equal to i with $\tilde{p}_i(0) = 1$ such that

$$b^{(2)} - A\bar{x}_i^{2,2} = \sum_{k=n+1}^{N} \phi_k \lambda_k \tilde{p}_i(\lambda_k) s_k\,, \tag{15}$$

where $\tilde{p}_i$ satisfies

$$\tilde{p}_i = \arg\min_{p \in \bar{\Pi}^i} \|p(A)(b^{(2)} - A\bar{x}_0^{2,2})\|. \tag{16}$$

Here, $\bar{\Pi}^i$ denotes the set of all polynomials with degree less than or equal to i with constant term 1. From Theorem 7.1 in [17], we have a bound on the ith QMR residual in terms of the ith GMRES residual:

$$\begin{aligned} \|b^{(2)} - Ax_i^{2,2}\| &\le \kappa_2(V_{2,i+1})\|b^{(2)} - A\hat{x}_i^{2,2}\| \\ &= \kappa_2^2(V_{2,i+1}) \min_{p \in \bar{\Pi}^i} \|p(A)(b^{(2)} - Ax_0^{2,2})\| \end{aligned} \tag{17}$$

where $\hat{x}_i^{2,2}$ is the ith iterate of GMRES with starting guess $x_0^{2,2}$.

Now

$$\begin{aligned} \min_{p \in \bar{\Pi}^i} \|p(A)(b^{(2)} - Ax_0^{2,2})\| &\le \|\tilde{p}_i(A)(b^{(2)} - Ax_0^{2,2})\| \\ &= \|\sum_{k=1}^{N} \phi_k \lambda_k \tilde{p}_i(\lambda_k) s_k\|. \end{aligned} \tag{18}$$

Substituting this into (17) and using (15) gives

$$\begin{aligned} \|b^{(2)} - Ax_i^{2,2}\| &\le \kappa_2^2(V_{2,i+1})\|\sum_{i=1}^{N} \tilde{p}_i(\lambda_k)\phi_k\lambda_k s_k\| \\ &\le \kappa_2(V_{2,i+1})\left(\|\sum_{k=n+1}^{N} \tilde{p}_i(\lambda_k)\phi_k\lambda_k s_k\| + \|\sum_{k\in\mathcal{I}} \tilde{p}_i(\lambda_k)\phi_k\lambda_k s_k\|\right) \\ &= \kappa_2(V_{2,i+1})\left(\|b^{(2)} - A\bar{x}_i^{2,2}\| + \|\sum_{k\in\mathcal{I}} \tilde{p}_i(\lambda_k)\phi_k\lambda_k s_k\|\right) \\ &\le \kappa_2(V_{2,i+1})\left(\|b^{(2)} - A\bar{x}_i^{2,2}\| + \sum_{k\in\mathcal{I}} |\tilde{p}_i(\lambda_k)||\phi_k\lambda_k|\right). \end{aligned} \tag{19}$$

Using the definition of $x_0^{2,2}$ as $x_m^{1,2}$, it is easy to show

$$b^{(2)} - Ax_0^{2,2} = (I - AV_{1,m}(\bar{T}_{1,m}^*\bar{T}_{1,m})^{-1}\bar{T}_{1,m}^* D_{1,m+1}^{-1} W_{1,m+1}^T)r_0^{1,2} = P_m^\perp r_0^{1,2}.$$

It is easy to see that $P_m^\perp$ is a projector since $(P_m^\perp)^2 = P_m^\perp$. Additionally, it is clear that $\mathcal{N}(P_m^\perp)$ is spanned by the columns of $AV_{1,m}$. Therefore the range $\mathcal{R}(P_m^\perp)$ is $(AV_{1,m})^\perp$. From (13) and the above equation we have $P_m^\perp r_0^{1,2} = \sum_{k=1}^N \phi_k \lambda_k s_k$. Using $z_j^* s_k = \delta_{jk}$ we obtain

$$|z_j^* P_m^\perp r_0^{1,2}| = |\phi_j \lambda_j|. \tag{20}$$

Substituting (20) into (19), we obtain the desired result. □

Now let us discuss why we expect δ to be small. Suppose that $c_i, i = m+1, \ldots, N$ are a basis for $(AV_{1,m})^\perp$. Then $P_m^\perp r_0^{1,2}$ can be written as $\sum_{i=m+1}^N \xi_i c_i$ for some numbers ξ_i where $c_i^* AV_{1,m} = 0$.

Next, let $T_{1,m}U_{1,m} = U_{1,m}\Sigma_{1,m}$ be the eigendecomposition of $T_{1,m}$, where $T_{1,m}$ denotes the tridiagonal $m \times m$ leading submatrix of $\bar{T}_{1,m}$. The condition $c_i^* AV_{1,m} = 0$ implies $c_i^* AV_{1,m}U_{1,m} = 0$ as well. Now since QMR is built on top of the unsymmetric Lanczos process, in exact arithmetic if m is large enough we expect the Ritz vectors given by the columns of $V_{1,m}U_{1,m}$ to contain good approximations to the n extremal right eigenvectors (the z_k such that $k \in \mathcal{I}$) of A [1, 5, 6]. Thus, the c_i must be nearly orthogonal to Z_n. In other words, the c_i lie mostly in $Z_n^\perp$.

We have $|z_j^* P_m^\perp r_0^{1,2}| = |\sum_{i=m+1}^N \xi_i z_j^* c_i|$ for $j \in \mathcal{I}$. Since c_i is mostly contained in $Z_n^\perp$ and $z_j^* s_k = 0$ for $j \in \mathcal{I}$ and $k \notin \mathcal{I}$ then $z_j^* c_i$ must be small. It follows that δ is small if m is sufficiently large.

Using the definition of $\tilde{p}_i$ in (16), we have in analogy with the standard GMRES upper bound for diagonalizable matrices [17], the following corollary:

COROLLARY 4.2. *Let S_n denote the $N \times (N-n)$ matrix with columns s_j for $j \notin \mathcal{I}$. Then*

$$\begin{aligned}\|b^{(2)} - Ax_i^{2,2}\|_2 &\le \kappa_2(V_{2,i+1}) \left(\min_{p \in \Pi^i} \max_{\substack{\lambda_k \\ k=n+1,\ldots,N}} |p(\lambda)| \, \|S_n\| \, (\sum_{k=n+1}^N |\phi_k \lambda_k|^2)^{1/2} + \delta \right) \\ &= \kappa_2(V_{2,i+1})(\min_{p \in \Pi^i} \max_{\substack{\lambda_k \\ k=n+1,\ldots,N}} |p(\lambda)| \, \|S_n\| \, \|PZ^* \bar{r}_0\| + \delta)\end{aligned}$$

with δ defined as in Theorem 4.1 and $P = [e_{n+1}, \ldots, e_N]^T$.

Proof: The proof follows from the result of the theorem by first writing (15) as $S_n \tilde{p}(\hat{\Lambda}) \sum_{k=n+1}^N \phi_k \lambda_k e_k$, where $\hat{\Lambda}$ is $\mathrm{diag}(\lambda_{n+1}, \ldots, \lambda_N)$, and then taking norms. To get the second equality, use the identity $e_k = PZ^* s_k$. □

Thus if m is large enough, then second seed system converges as if the extremal end of the spectrum of A has been cut off.

Next, we bound the residual norms of the non-seed systems.

THEOREM 4.3. *Let j denote the index of the seed system and l denote the index of a (non-converged) non-seed system. Then*

$$\begin{aligned}\|r_k^{j,l}\| \le \sqrt{k+1} &\left(|\gamma_1^{j,l}| \left| \prod_{i=0}^{k-1} s_{j,k-i} \right| + \sum_{i=0}^{k-1} |\gamma_{k-i+1}^{j,l}| |c_{j,k-i}| \left| \prod_{m=0}^{i-1} s_{j,k-m} \right| \right) \\ &+ \sqrt{N-k-1} \|h_k^{j,l}\|\end{aligned} \tag{21}$$

where $h_k^{j,l} = [\gamma_{k+2}^{j,l}, \ldots, \gamma_N^{j,l}]^T$.

Proof: Using (7) we have

$$r_k^{j,l} = r_0^{j,l} - V_{j,k+1}\bar{T}_{j,k} z_k^{j,l} = V_{j,k+1}(g_{k+1}^{j,l} - \bar{T}_{j,k} z_k^{j,l}) + V_{j,k+1}^{\perp} h_k^{j,l}, \tag{22}$$

where $V_{j,k+1}^{\perp} = [v_{j,k+2}, \ldots, v_{j,N}]$. Taking norms, we get

$$\|r_k^{j,l}\| \leq \|V_{j,k+1}\| \|g_{k+1}^{j,l} - \bar{T}_{j,k} z_k^{j,l}\| + \|V_{j,k+1}^{\perp}\| \|h_k^{j,l}\|. \tag{23}$$

Since $V_{j,k+1}$ and $V_{j,k+1}^{\perp}$ have $k+1$ and $N-k-1$ columns of Euclidean norm 1, respectively, we have

$$\|V_{j,k+1}\| \leq \sqrt{k+1}, \quad \|V_{j,k+1}\| \leq \sqrt{N-k-1}. \tag{24}$$

From (10) and (11), noting that by definition $\tau_1^{jl} = \gamma_1^{j,l}$, it follows that

$$\|g_{k+1}^{j,l} - \bar{T}_{j,k} z_k^{j,l}\| \leq |\gamma_1^{j,l}| \left| \prod_{i=0}^{k-1} s_{j,k-i} \right| + \sum_{i=0}^{k-1} |\gamma_{k-i+1}^{j,l}| |c_{j,k-i}| \left| \prod_{m=0}^{i-1} s_{j,k-m} \right|. \tag{25}$$

Finally applying (25) and (24) to (23), we obtain the desired result. □

In order for us to interpret the above result, let us consider from [8, Proposition 4.1a], the upper bound on the residual norm of the seed system at iteration k:

$$\|r_k^{j,j}\| \leq \sqrt{k+1} \|r_0^{j,j}\| \left| \prod_{i=0}^{k-1} s_{j,k-i} \right|, \tag{26}$$

where $\|r_0^{j,j}\| \left| \prod_{i=0}^{k-1} s_{j,k-i} \right| = |\tau_{k+1}^{j,l}|$ is the norm of the quasi-residual of the seed system which is being minimized at every iteration. Therefore, if the norm of the quasi-residual is quickly decreasing with k then $|\prod_{i=0}^{k-1} s_{j,k-i}|$ must quickly be decreasing with k. But this implies that the first term in parenthesis of (21) is rapidly being reduced by the same amount.

Note that since $r_0^{j,j} = V_{j,k+1} \|r_0^{j,j}\| e_1$, we have

$$D_{j,k+1}^{-1} W_{j,k+1}^T r_0^{j,j} = \|r_0^{j,j}\| e_1.$$

Using notation consistent with that for the non-seed systems, $\|r_0^{j,j}\| = |\gamma_1^{j,j}|$ and $|\gamma_k^{j,j}| = 0, k > 1$. Hence, if the systems are closely related, in the sense described in the introduction, we expect (and found in practice) $|\gamma_k^{j,l}|$ to be small for $l \neq j$ as k increases; this implies the last term in (21) is small. The middle term in (21) is also small by virtue of the fact that the largest $|\gamma_{k-i+1}^{j,l}|$ values (only for $i \approx k$, assuming closeness) are multiplied by terms $\ll 1$.

5. Block Variant. The goal of this section is to develop a multiple seed variant of the projection based algorithm introduced in §3. Since our multiple seed algorithm will be based on the block QMR algorithm (BL-QMR) of [7], we first need to present some properties of their algorithm together with some notation.

5.1. BL-QMR Background. The BL-QMR algorithm of Freund and Malhotra attempts to solve (1) in the following way. First, given K vectors r_i and p vectors l_i, they define

$$R = [r_1, \ldots, r_K], \quad L = [l_1, \ldots, l_p].$$

The block Krylov sequences generated by R, A and L, A^T are

$$\left\{R, AR, A^2R, \ldots, A^{j-1}R, \ldots\right\} \text{ and } , \left\{L, A^TL, \ldots, (A^T)^{j-1}L, \ldots\right\}. \tag{27}$$

However, if $A^{j-1}r_i$ (likewise $(A^T)^{j-1}l_i$) is linearly or nearly linearly dependent on the previous vectors, so are all $A^k r_i$ (likewise $(A^T)^k l_i$) for $k \geq j$. Thus, Freund and Malhotra propose scanning the vectors in the two sequences in (27) from left to right and deleting those which are linearly or nearly linearly dependent on previous ones. In the process they obtain deflated Krylov sequences whose vectors are linearly independent. Following [7], we refer to the n-dimensional subspaces generated by these deflated sequences as $K_n^{\mathrm{dl}}(R, A)$ and $K_n^{\mathrm{dl}}(L, A^T)$. Note that in the presence of no deflation, $K_n^{\mathrm{dl}}(R, A)$ and $K_n^{\mathrm{dl}}(L, A^T)$ are spanned by the first n columns of (27) with $n \leq jK$ or $n \leq jp$, respectively.

Within BL-QMR is a Lanczos-type of algorithm which incorporates the deflation as mentioned above in order to generate biorthogonal bases for $K_n^{\mathrm{dl}}(R, A)$ and $K_n^{\mathrm{dl}}(L, A^T)$: that is, two sequences of right and left Lanczos vectors

$$v_1, \ldots, v_n \text{ and } w_1, \ldots, w_n, \quad n = 1, 2, \ldots$$

such that

$$\begin{aligned} \mathrm{span}\{v_1, \ldots, v_n\} &= K_n^{\mathrm{dl}}(R, A) \\ \mathrm{span}\{w_1, \ldots, w_n\} &= K_n^{\mathrm{dl}}(L, A^T) \\ w_j^T v_k &= \begin{cases} 0, & \text{if} j \neq k \\ \delta_j \neq 0, & \text{if} j = k \end{cases}. \end{aligned}$$

Putting the v_i into the columns of an $N \times n$ matrix V_n and the w_i into the columns of an $N \times n$ matrix W_n, we have by biorthogonality

$$W_n^T V_n = D_n \equiv \mathrm{diag}(\delta_1, \ldots, \delta_n), \; n = 1, 2, \ldots.$$

Also, the matrix equation relating the $v's$ is

$$AV_\mu = V_n T_\mu + \hat{V}_\mu^{\mathrm{dl}}, \mu \geq 1$$

where $\mu = n - m_{\mathrm{cr}}$ and m_{cr} is defined by the fact that $K - m_{\mathrm{cr}}$ is the total number of deflations performed in the v sequence up to iteration n in the Lanczos algorithm. Further, T_μ is $n \times \mu$ with lower bandwidth $K + 1$ and upper bandwidth $p + 1$. Also, $\hat{V}_\mu^{\mathrm{dl}} = V_\mu^{\mathrm{dl}} + E_\mu$ where V_μ^{dl} is $N \times \mu$ but has only $K - m_{\mathrm{cr}}$ nonzero columns corresponding to vectors that were deflated and E_μ has nonzero entry in row i column $p + j, j = 1, \ldots$ only if a deflation of the ith w occurs for $i > K$. We note that if *deftol* is the deflation tolerance then $\|V_\mu^{\mathrm{dl}}\| \leq$ *deftol* $\sqrt{K - m_{\mathrm{cr}}}$. For further details, the reader is referred to [7].

Now let us assume $R = [r_0^{(1)}, r_0^{(2)}, \ldots, r_0^{(K)}]$; that is, the matrix R contains the initial residuals of each of the K systems we would like to solve. Thus, the $v's$

correspond to the initial residuals. The way the deflation strategy in [7] works is that if a v is deflated, the corresponding linear system is also set aside; then upon convergence of the remaining systems, the solution to the deflated system is updated using the solutions of the other systems. Thus, in what follows we consider only the updates to the non-deflated linear systems, and we denote with a subscript cr submatrices of the originals with $m_{\rm cr}$ columns that correspond to these systems

Recall that when QMR is applied to a single linear system, the μth iterate is an appropriate linear combination of the Lanczos vectors, plus the initial guess. Similarly, the block QMR iterate is defined as

$$X_{\mu,\rm cr} = X_{0,\rm cr} + V_\mu Z, \;\; Z \in \mathcal{C}^{\mu \times m_{\rm cr}}.$$

As with QMR, then, we need to find the matrix Z which determines the appropriate linear combination. Following [7], the residual block $R_{\mu,\rm cr}$ related to $X_{\mu,\rm cr}$ satisfies

$$\begin{aligned} R_{\mu,\rm cr} &= B_{\rm cr} - AX_{\mu,\rm cr} \\ &= R_{0,\rm cr} - AV_\mu Z \\ &= R_{0,\rm cr} - V_n T_\mu Z - \hat{V}_{\mu,\rm dl} Z \\ &= V_n \left(\begin{bmatrix} \beta_{\rm cr} \\ 0 \end{bmatrix} - T_\mu Z \right) - \hat{V}_{\mu,\rm dl} Z, \end{aligned}$$

where $\beta_{\rm cr}$ is $m_1 \times m_{\rm cr}$ defined by taking the appropriate columns of β, with

$$V_{m_1}\beta + V_{0,\rm dl} = R$$

and m_1 is the number of columns of R (recall R has K columns) which are not deflated as the first K Lanczos vectors are created ($m_1 \leq K$).

Because the columns of V_n are not unitary and $\hat{V}_{\mu,\rm dl}$ has non-zero columns, one cannot find Z such that $\|R_{\mu,\rm cr}\|$ is minimal. Rather, we seek $Z = Z_\mu$ such that

$$Z_\mu = \arg \min_{Z \in \mathcal{C}^{\mu \times m_{\rm cr}}} \| \begin{bmatrix} \beta_{\rm cr} \\ 0 \end{bmatrix} - T_\mu Z \|.$$

Since T_μ is banded, the standard approach based on the QR factorization of T_μ is used to implicitly determine Z_μ and ultimately determine short term recurrences for $X_{\mu,\rm cr}$. Following [7] we have

$$T_\mu = (Q^{(\mu)})^* \begin{bmatrix} R^{(\mu)} \\ 0 \end{bmatrix}$$

for a unitary $n \times n$ matrix $Q^{(\mu)}$ and a non-singular, $\mu \times \mu$, upper triangular matrix $R^{(\mu)}$. Thus,

$$Z^{(\mu)} = (R^{(\mu)})^{-1} t_\mu \text{ where } \begin{bmatrix} t_\mu \\ \tau_\mu \end{bmatrix} = Q^{(\mu)} \begin{bmatrix} \beta_{\rm cr} \\ 0 \end{bmatrix}.$$

Finally,

$$X_{\mu,\rm cr} = X_{0,\rm cr} + V^{(\mu)} (R^{(\mu)})^{-1} t_\mu \tag{28}$$

$$= X_{\mu-1,\rm cr} + p_\mu y_\mu^T, \tag{29}$$

where p_μ and y_μ^T are given by equations (5.13) and (5.10) of [7]:

$$p_\mu = (v_\mu - \sum_{i=j^*}^{\mu-1} p_i\theta_i)/\theta_\mu \tag{30}$$

$$\begin{bmatrix} y_\mu^T \\ \tau_\mu \end{bmatrix} = Q_\mu \begin{bmatrix} \tau_{\mu-1} \\ 0 \end{bmatrix}. \tag{31}$$

The θ_i are scalars corresponding to the last column of $R^{(\mu)}$ and Q_μ (not to be confused with $Q^{(\mu)}$) is a particular matrix of Givens rotations described in (5.2) of [7].

5.2. Block QMR-Projection Method. In a manner similar to §3, we describe a block QMR-projection approach to solving (1) that combines the advantageous properties of both algorithms.

Suppose that we select a subset of size $m < K$ linear systems to serve as "seed" from among the original K. Let $\mathcal{I}_{m_1}$ be the index set $i_1, \ldots, i_m$ of the chosen systems. We use $\mathcal{I}_{m_1}^c$ to denote the indecies from 1 to K which are not in $\mathcal{I}_{m_1}$. We put the $b^{(j)}$ with $j \in \mathcal{I}_{m_1}$ into the m columns of the matrix $B^{(1)}$. We put the remaining $J = K - m$ right hand sides (corresponding to non-seed systems indexed by $\mathcal{I}_{m_1}^c$) into the columns of the matrix $B^{(2)}$. We define $X_0^{(1)}$ as the matrix $[x_0^{(i_1)}, \ldots, x_0^{(i_m)}]$ of initial guesses for the m seed systems, and $X_0^{(2)}$ as the matrix of initial guesses for the non-seed systems. The corresponding initial block residuals are $R_0^{(1)} = B^{(1)} - AX_0^{(1)}$ and $R_0^{(2)} = B^{(2)} - AX_0^{(2)}$.

The idea is to set R (and L) defined in the previous section to $R_0^{(1)}$ and run BL-QMR to solve the seed systems while using a projection idea to update the non-seed systems. Once BL-QMR on the seed systems converges, the process is repeated by choosing a new subset, indexed by $\mathcal{I}_{m_2} \subset \mathcal{I}_{m_1}^c$, of the non-converged, non-seed systems. The systems indexed by $\mathcal{I}_{m_2}$ now serve as seed, where the columns of $X_0^{(1)}$ are now understood to be the estimated solutions, generated in the first round of projected BL-QMR, to the systems with indecies in $\mathcal{I}_{m_2}$. The remaining systems, indexed by $\mathcal{I}_{m_2}^c = \mathcal{I}_{m_1}^c \setminus \mathcal{I}_{m_2}$, are updated by projection. In the following, $X_\mu^{(1)}$ ($R_\mu^{(1)}$) denotes the μth block iterate (residual) of the current block seed while $X_\mu^{(2)}$ ($R_\mu^{(2)}$) denotes the μth block iterate (residual) of the current non-seed block. We shall further assume that m is the number of *current* seed systems and J is the number of *current* non-converged, non-seed systems. The numbers m and J can change at each round.

At iteration μ, we desire our non-seed systems lie in the current Krylov subspace:

$$X_\mu^{(2)} \in X_0^{(2)} + K_\mu^{\mathrm{dl}}(R_0^{(1)}, A).$$

Since the columns of V_μ span this subspace, this means we want

$$X_\mu^{(2)} = X_0^{(2)} + V_\mu Z_\mu^{(2)}$$

for some $\mu \times J$ matrix $Z_\mu^{(2)}$. Now we must decide how to define $Z_\mu^{(2)}$. We observe

$$\begin{aligned} R_\mu^{(2)} &= B^{(2)} - A(X_0^{(2)} + V_\mu Z_\mu^{(2)}) \\ &= R_0^{(2)} - AV_\mu Z_\mu^{(2)} \\ &= R_0^{(2)} - V_n T_\mu Z_\mu^{(2)} - \hat{V}_{\mathrm{dl}} Z_\mu^{(2)} \end{aligned}$$

Using biorthogonality we have

$$D_n^{-1} W_n^T R_\mu^{(2)} = D_n^{-1} W_n^T R_0^{(2)} - T_\mu Z_\mu^{(2)} - D_n^{-1} W_n^T \hat{V}_{\mathrm{dl}} Z_\mu^{(2)}.$$

Then

$$\|D_n^{-1} W_n^T R_\mu^{(2)}\| \leq \|D_n^{-1} W_n^T R_0^{(2)} - T_\mu Z_\mu^{(2)}\| + \|D_n^{-1} W_n^T \hat{V}_{\mathrm{dl}} Z_\mu^{(2)}\|.$$

Note that if no deflations have occurred, $\hat{V}_{\mathrm{dl}}$ is zero, so we have equality rather than inequality. Therefore, in analogy with the single seed algorithm of §3, we define

$$Z_\mu^{(2)} \equiv \min_{Z \in C^{\mu \times J}} \|D_n^{-1} W_n^T R_0^{(2)} - T_\mu Z\|.$$

Using the QR factorization $T_\mu = Q^{(\mu)} R^{(\mu)}$ as described in the previous section we have

$$Z_\mu^{(2)} = \arg\min_Z \|Q^{(\mu)} G_\mu - \begin{bmatrix} R^{(\mu)} \\ 0 \end{bmatrix} Z\|$$

where G_μ is the $n \times J$ matrix $G_\mu = D_n^{-1} W_n^T R_0^{(2)}$. If

$$\begin{bmatrix} t_\mu^{(2)} \\ \tau_{\mu+1}^{(2)} \end{bmatrix} = Q^{(\mu)} G_\mu \tag{32}$$

we obtain

$$Z_\mu^{(2)} = (R^{(\mu)})^{-1} t_\mu^{(2)}, \tag{33}$$

so that

$$\|D_n^{-1} W_n^T R_0^{(2)} - T_\mu Z_\mu^{(2)}\| = \|\tau_{\mu+1}^{(2)}\|. \tag{34}$$

Using

$$G_\mu = \begin{bmatrix} G_{\mu-1} \\ g_\mu^T \end{bmatrix},$$

together with (32) and the definition of $Q^{(\mu)}$ in 5.1 of [7], it is easy to show that

$$\begin{bmatrix} t_\mu^{(2)} \\ \tau_{\mu+1}^{(2)} \end{bmatrix} = \begin{bmatrix} I_{\mu-1} & 0 \\ 0 & Q_\mu \end{bmatrix} \begin{bmatrix} t_{\mu-1}^{(2)} \\ \tau_{\mu-1}^{(2)} \\ g_\mu^T \end{bmatrix} = \begin{bmatrix} t_{\mu-1}^{(2)} \\ Q_\mu \begin{bmatrix} \tau_{\mu-1}^{(2)} \\ g_\mu^T \end{bmatrix} \end{bmatrix}. \tag{35}$$

Thus, $t_\mu^{(2)}$ differs from $t_{\mu-1}^{(2)}$ only in its last row, which we call $(y_\mu^{(2)})^T$:

$$t_\mu^{(2)} = \begin{bmatrix} t_{\mu-1}^{(2)} \\ (y_\mu^{(2)})^T \end{bmatrix} \text{ where } (y_\mu^{(2)})^T \in \mathcal{C}^{1 \times J}.$$

From the above relation and (35) it follows that to obtain $(y_\mu^{(2)})^T$ one only needs to perform a product with Q_μ:

$$\begin{bmatrix} (y_\mu^{(2)})^T \\ \tau_\mu^{(2)} \end{bmatrix} = Q_\mu \begin{bmatrix} \tau_{\mu-1}^{(2)} \\ g_\mu^T \end{bmatrix},$$

which, since Q_μ by definition is a product of $m_{\rm cr}$ Givens rotations, is an easy task.

With p_i defined as in (30), it is now easy to show that the μth non-seed block iterate is

$$X_\mu^{(2)} = X_{\mu-1}^{(2)} + p_\mu (y_\mu^{(2)})^T.$$

Thus, we may readily show

$$R_\mu^{(2)} = R_{\mu-1}^{(2)} - A p_\mu (y_\mu^{(2)})^T. \tag{36}$$

However, using the definition of p_μ, we find an update formula for the block residual which does not actually require any additional matrix-vector products.

LEMMA 5.1. *$R_\mu^{(2)}$ can be updated from $R_{\mu-1}^{(2)}$ in at most $O(N(J+2m))$ additional floating point operations.*

Proof: By substituting (30) into (36), we obtain the following formula for updating $R_\mu^{(2)}$:

$$R_\mu^{(2)} = R_{\mu-1}^{(2)} - f_\mu (y_\mu^{(2)})^T, \quad \text{with} \quad f_i \equiv A p_i = \frac{1}{\theta_i}(A v_i - \sum_{k=j^*}^{i-1} \theta_k f_k). \tag{37}$$

Consider forming f_μ. Now the matrix-vector product $A v_\mu$ is computed in the course of the Lanczos process at iteration μ and need not be recomputed. Therefore, it is clear that to compute the length N vector f_μ requires at most $O(2mN)$ flops since $(\mu - j^*) \le 2m$ by definition (see Section 5 of [7]). Noting that the computation of the outer product $f_\mu (y_\mu^{(2)})^T$ requires $O(JN)$ operations, the proof is complete. □

We note that a similar update is valid for $R_\mu^{(1)}$:

$$R_\mu^{(1)} = R_{\mu-1}^{(1)} - f_\mu (y_\mu^{(1)})^T. \tag{38}$$

5.3. Issues in Practical Computation. Clearly, the performance of our multiple seed algorithm in terms of savings of matrix-vector products depends on which, and how many, systems are chosen to be seed. Deflation in BL-QMR solves the problem of removing redundancy if systems with starting residuals which are nearly linearly dependent are chosen as seed. Ideally, however, we would like to choose as seed systems some subset of the non-converged systems which are in some sense optimally independent in order to increase the chance that the solutions to the non-seed systems will lie nearly in the Krylov subspaces generated by the seed systems, thereby ensuring the effectiveness of the projection process.

In our examples, we used the following heuristics to determine which and how many seed systems to use. First, we set *deftol*=1e-8, $B = [b^{(1)}, \ldots, b^{(K)}]$ and $X_0 = 0$ and used the initialization phase of BL-QMR to determine which of the K right hand sides would be kept and which would be deflated. Suppose $k_1 \le K$ of these were kept and that B_{k_1} is the matrix containing these as its columns. Then, as $k_1 \le K \ll N$, we computed a compact pivoted QR factorization of B_{k_1},

$$\tilde{B}_{k_1} \equiv B_{k_1} \Pi = QR \quad Q \in \mathcal{C}^{N \times k_1}, \; R \in \mathcal{C}^{k_1 \times k_1}$$

to determine which of the remaining were most independent. Here, Π is just a permutation matrix which serves to permute the columns of B_{k_1} such that the first few columns of $B_{k_1}\Pi$ are the most independent. In particular, if ρ denotes the diagonal entries of R and if $|\rho(1)|/|\rho(i)| > 10^5$ for any $1 \le i \le k_1$, then we discard the

corresponding column of $\tilde{B}_{k_1}$. The remaining m columns of $\tilde{B}_{k_1}\Pi$ serve as the seed systems. On the next round of projected BL-QMR, however, we simply decided on a new number of seeds to use ($m \leftarrow \lceil m/2 \rceil$), and took those with the largest m relative residuals to serve as seed. More efficient means of determining m for each round and for determining the m seeds need to be determined in the future; clearly if K is very large, our heuristic would be too expensive.

One additional problem that we encountered in practice in using either our single seed or our multiple seed algorithm was that loss of biorthogonality could affect the accuracy of the $\gamma_n^{jl} = (1/\delta_{j,n}) w_{j,n}^T r_0^{jl}$, or $g_n^T = (1/\delta_n) w_n^T R_0^{(2)}$. This loss of accuracy would thereby adversely affect the convergence of the computed solution. To avoid this difficulty for the block projection algorithm, we used the following fact. If no deflations were performed up to the μth iteration when solving the single seed system, we have

$$R_{\mu-1}^{(2)} = R_0^{(2)} - V_{n-1} T_{\mu-1} Z_{\mu-1}^{(2)},$$

and therefore

$$g_n^T = \frac{1}{\delta_n} w_n^T R_{\mu-1}^{(2)} = \frac{1}{\delta_n} w_n^T R_0^{(2)},$$

where it is understood that $R_j^{(2)} = R_0^{(2)}, j < 0$. Thus, at the beginning of iteration $\mu \geq 1$, we computed g_n^T based on the current residual estimate, then updated the residual estimate using Lemma 5.1. If deflations do occur, observe

$$g_n^T = \frac{1}{\delta_n} w_n^T R_{\mu-1}^{(2)} - \frac{1}{\delta_n} w_n^T \hat{V}_{\mathrm{dl}} Z_\mu^{(2)}.$$

In our examples, the second term was on the order of the deflation tolerance. This was because $\hat{V}_{\mathrm{dl}} = V_{\mathrm{dl}}$ since no w deflations occurred for indecies larger than J. Hence non-zero columns of V_{dl} were nearly linear combinations of the first m_1 v's for which $\frac{1}{\delta_n} w_n^T v = 0$. In this work we choose to ignore the second term rather than go to the extra expense of computing inner products with the non-zero columns of $\hat{V}_{\mathrm{dl}}$.

Likewise, for the single seed algorithm we use

$$\gamma_n^{jl} = \frac{1}{\delta_{j,n}} w_{j,n}^T r_0^{jl} = \frac{1}{\delta_{j,n}} w_{j,n}^T r_{n-2}^{jl}, \quad n \geq 2.$$

An investigation into the reason behind the success of these approaches in finite precision arithmetic will be the subject of future work. We note that a similar phenomenon was observed in [15] with respect to practical implementation of GMRES variants and an explanation for such behavior in finite precision arithmetic was given.

6. Numerical Results. In this section we given numerical results which indicate the potential effectiveness of our approach on electromagnetic scattering problems. All experiments were conducted in Matlab using IEEE double floating point arithmetic. We compare our results with results from applying the Matlab implementation of block QMR with deflation, algorithm BL-QMR in [7].

Mathematically, we would like to solve a two-dimensional Helmholtz-type equation for the scattered electric field $E(x, y)$:

$$(\Delta + k^2(x,y))E(x,y) = \chi_m(x,y)E_0(x,y) \quad \text{in } \Omega \tag{39}$$

with perfectly matched layer (PML) boundary conditions [2, 13]: the specific mathematical formulation we use is described in [11]. Here $k^2(x,y) = \omega^2\mu_0\epsilon(x,y)$ is the square of the wave number, with ω representing angular frequency and μ_0 a constant denoting the magnetic permeability. The function $\epsilon(x,y)$, called the electrical permittivity, is defined as the complex-valued quantity

$$\epsilon = \epsilon_0\epsilon_{rel} + i\frac{\sigma}{\omega},$$

for some real $\sigma \geq 0, \epsilon_{rel} \geq 1$ with $i = \sqrt{-1}$ and ϵ_0 a constant (the permittivity of free-space). The value σ is the conductivity of the material. The function χ_m describes the properties of the buried object and has support only over the region in which the object is located. $E_0(x,y)$ is the incident electric field, which is known.

We discretize the continuous problem in space using finite differences, which leads to a matrix equation involving the matrix A which is $N \times N$, sparse, complex, and structured, but neither symmetric nor Hermitian due to the boundary conditions. Because the matrix is highly indefinite, we need to use a preconditioner to speed convergence. In all examples, the preconditioner we use is the one described in [11] and we perform all preconditioning from the right.

For both our algorithm and the BL-QMR algorithm, we take the initial starting guesses $x_0^{(j)}$ to be zero. We stop running our algorithm when the norms of the residuals of all of the systems, relative to the norms of their respective right hand sides, are less than $tol = 10^{-7}$. For the BL-QMR algorithm, we update residuals via (38). For our block-seed algorithm we use (38) to update residuals of the seed block system and monitor convergence by taking norms of the columns of $R_\mu^{(1)}$. We update the residuals of the non-seed systems via (37) compute norms of columns only after the seed systems have converged in order to determine which systems should next serve as seed. Since the major computational expense per iteration is the matrix-vector product and application of the preconditioner, as our measure of success we consider the total number of matrix-vector products required for all the systems to converge.

6.1. Example 1. In this experiment we would like to find the scattered electric fields caused when plane waves at various angles impinge on a horizontal air-soil interface and scatter from a 7cm $\times$ 6cm object buried 5cm below the surface. Each angle corresponds to a different E_0 in (39), which in turn corresponds to a different right hand side $b^{(j)}$ in (1). Figure 1 gives a physical illustration of the problem.

In this example, we use a soil type (referred to as "Seabee" in the literature [14]) and conduct experiments at two different frequencies, $\omega/(2\pi) = 45$ MHz and 475 MHz. At 45MHz, Seabee has $\epsilon_{rel} = 35.65$ and $\sigma = .13$, while at 475MHz $\epsilon_{rel} = 21.31$ and $\sigma = .23$. We assumed that the buried object has $\epsilon_{rel} = 2.9$ and $\sigma = .001$ at both frequencies. For air, $\epsilon_{rel} = 1$, $\sigma = 0$. In discretizing the problem, we have sampled at a rate of 50 points per wavelength of soil at 45 MHz and 20 points per wavelength at 475 MHz. In both cases, the total number of unknowns (N) is $2(2^7 - 1)$.

We centered the buried object (refer to Figure 1) and considered the scattered field due to planewaves impinging on the surface at evenly spaced angles from -60 to 60 degrees from the normal. The second column in Tables 1 and 2 gives the total number of matrix-vector products needed if preconditioned QMR is applied to each system separately. Note that we define the matrix-vector product count as the number of multiplies by AM^{-1} or its transpose where M is the preconditioner. The third column gives the total number of matrix-vector products needed if our preconditioned QMR with projection algorithm is used. The forth and fifth columns

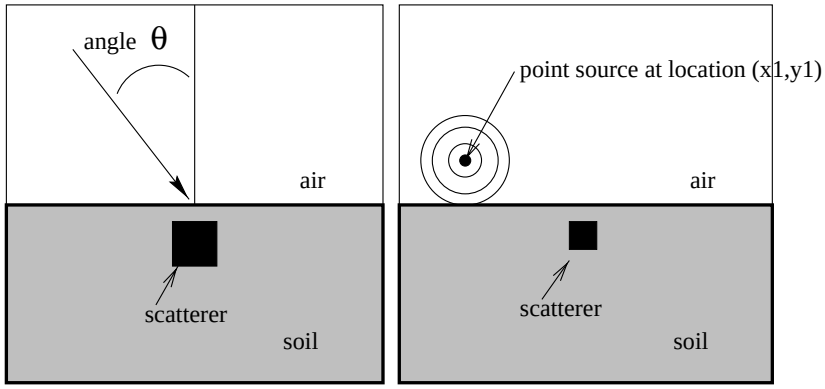

FIG. 1. *Physical configurations for Example 1 (left) and Example 2 (right).*

	No. Matrix-Vector Products, 45 MHz				
No. RHS	Seq.	Prj.	BL-QMR $1E-8$	BL-QMR $1E-9$	BL-QMR + Prj.
7	816	260	—	194	166
13	1554	288	—	198	166
25	2994	196	—	208	166

TABLE 1

Example 1, 45MHz: Number of matrix-vector products required for convergence by each of the methods (each system independently, single-seed QMR-projection, BL-QMR with $deftol = 10^{-8}$*, BL-QMR with* $deftol = 10^{-9}$*, and block-QMR-projection) for experiments involving different numbers of right hand sides. Dash indicates no convergence of the method in under 300 iterations.*

give the total number of matrix-vector products computed when solving the problem using BL-QMR with two different deflation tolerances. The final column shows the total number of matrix-vector products computed when our block-seed approach is used ($deftol = 10^{-9}$), where the initial and subsequent seed blocks are chosen using the heuristic outlined in §5.3. The dashes in the table indicate that the convergence tolerance was not met within *maxit=300* iterations (600 matrix-vector products), as the result of stagnation in the relative residual.

As Table 1 shows, for the 45MHz case, BL-QMR failed to converge after 300 iterations in all cases when the deflation tolerance was set to 10^{-8}. On the other hand, BL-QMR notably outperformed our single seed projection method for the case of 7 and 13 right hand sides when the deflation tolerance was 10^{-9}. Our single seed method only outperformed BL-QMR with a deflation tolerance of 10^{-9} when 25 right hand sides were used, and always substantially outperformed the naive approach of solving each system separately. The last column of the table shows that our block seed projection approach performs noticeably better than all the other methods.

At 475MHz, we expected our $x^{(j)}$'s not to be as close as in the previous case due to the underlying physics of the problem, and therefore, we did not expect as much savings with our single seed projection approach. Indeed, Table 2 shows that although the single seed approach gives significant savings in work over simply solving each system separately, the difference between the second and third columns is not as dramatic as in Table 1. Table 2 also shows that BL-QMR, with the deflation tolerance set at either 10^{-8} or 10^{-9}, outperforms our single seed projection approach. However, comparing the last column with the others, we find that our block seed projection

	No. Matrix-Vector Products, 475 MHz				
No. RHS	Seq.	Prj.	BL-QMR $1E-8$	BL-QMR $1E-9$	BL-QMR + Prj.
7	556	328	178	178	176
13	1036	322	220	254	196
25	1992	396	264	338	194

TABLE 2

Example 1, 475MHz: Number of matrix-vector products required for convergence by each of the methods for experiments involving different numbers of right hand sides.

	No. Matrix-Vector Products				
No. RHS	Seq.	Prj.	BL-QMR $1E-8$	BL-QMR $1E-9$	BL-QMR + Prj.
25	1482	588	—	326	236
35	2082	648	—	374	290

TABLE 3

Example 2: Number of matrix-vector products required for convergence by each of the methods for experiments involving different numbers of right hand sides. Dash indicates no convergence of the method in under 300 iterations.

approach can provide substantial savings over the other methods.

6.2. Example 2. For our second example, each of our K systems corresponds to solving for scattered electric field from a buried object when the source of the incident field is a point source, located at position x_i, y_i above the earth (see Figure 1). We consider the case when the frequency is 480MHz, and the soil has $\epsilon_{rel} = 6.5$ and $\sigma = .019$. As before, the buried object has $\epsilon_{rel} = 2.9$ and $\sigma = .001$. The buried object is 8cm by 8cm buried 5cm deep and centered left to right. The width of each cell in the discrete grid is 2cm, and the total number of unknowns (N) is $2(2^6 - 1)$. Our point sources are each located 2cm above the earth, and either vary in the horizontal direction, with 0 being in the middle, from -24cm to 24cm in 2cm increments (resulting in 25 systems) or -34cm to 34cm in 2cm increments (resulting in 35 systems). The numbers of matrix-vector products required by each of the different methods to solve these systems are given in Table 3.

Table 3 indicates that the method consistently requiring the fewest number of matrix-vector products is our multiple seed projection approach. The single seed approach requires more work than BL-QMR when the deflation tolerance was set to 10^{-9}; however, with the deflation tolerance at 10^{-8}, BL-QMR will not converge in under 300 iterations. We note that in similar experiments that we conducted, we found that the deflation tolerance sometimes had to be as small as 10^{-10} for BL-QMR to converge in under 300 iterations.

7. Conclusions and Future Work. We have introduced a new single-seed projection approach, based on QMR, for solving multiple linear systems with the same coefficient matrix but different right hand sides. Our approach, compared to solving each system separately by QMR, can significantly reduce the work (e.g. number of matrix-vector products) needed to solve all the systems to within a specified tolerance provided the right hand sides are close. We provided theory for the single-seed approach which suggests that under certain conditions, QMR on subsequent seed systems converges as if the extremal end of the spectrum has been cut off. We also gave an upper bound for the rate of convergence of the non-seed systems.

As our numerical results showed, the BL-QMR algorithm [7] could sometimes

outperform our single-seed QMR-projection method in terms of matrix-vector product savings, particularly when the right hand sides are not as close. Therefore, we developed a block-seed projection method, based on the BL-QMR algorithm, which combined the best aspects of both algorithms. Numerical results indicated that significant savings in work could be achieved using our block-seed projection approach as long as there was at least some shared information among the right hand sides. We will address the associated theoretical convergence issues for our block-seed approach in further work. The performance of our block-seed approach depended on our choices of successive seed blocks. Determining good seed selection strategies remains a subject for future research.

REFERENCES

[1] Z. Bai, *Error analysis of the lanczos algorithm for the nonsymmetric eigenvalue problem*, Math. Comp., 62 (1994), pp. 209–226.

[2] J. Berenger, *A perfectly matched layer for the absorption of electromagnetic waves*, J. Math. Phys., 114 (1994), pp. 185–200.

[3] W. E. Boyse and A. A. Seidel, *A block QMR method for computing multiple simultaneous solutions to complex symmetric systems*, SIAM J. Sci. Comput., 17 (1996), pp. 263–274.

[4] T. F. Chan and W. L. Wan, *Analysis of projection methods for solving linear systems with multiple right hand sides*, SIAM J. Sci. Comput., 18 (1997), pp. 1698–1721.

[5] J. Cullum, *Arnoldi versus nonsymmetric Lanczos algorithms for solving nonsymmetric eigenvalue problems*, Tech. Report CS-TR-95-123, UMIACS-TR-3576, Department of Computer Science and Institute for Advanced Studies, University of Maryland, College Park, 1995.

[6] D. Day, *An efficient implementation of the nonsymmetric Lanczos algorithm*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 566–589.

[7] R. Freund and M. Malhotra, *A block-QMR algorithm for non-Hermitian linear systems with multiple right hand sides*, Linear Algebra and Its Applications, 254 (1997), pp. 197–257.

[8] R. Freund and N. Nachtigal, *QMR: A quasi minimal residual method for non-hermitian linear system s*, Numerische Mathematik, 60 (1991), pp. 315–339.

[9] ———, *An implementation of the QMR method based on coupled two-term recurrences*, SIAM J. Sci. Comput., 15 (1994), p. 313.

[10] P. Joly, *Resolution de systems lineaires avec plusieurs seconds membres par la methode du gradient conjugue*, Tech. Report Tech. Report R-91012, Publications du Laboratoire d'Analyse Numerique, Universite Pierre et Marie Curie, Paris, March 1991. cited in [14].

[11] M. Kilmer, E. Miller, and C. Rappaport, *Preconditioners for structured matrix problems arising in subsurface object detection*, (1998). Submitted to *SIAM J. Sci. Comput.*

[12] D. P. O'Leary, *The block conjugate gradient algorithm and related methods*, Linear Algebra Appl., 29 (1980), pp. 293–322.

[13] C. Rappaport, *Interpreting and improving the pml absorbing boundary condition using anisotropic lossy mapping of space*, IEEE Trans. Magn., 32 (1996), pp. 968–974.

[14] E. M. Rosen and T. W. Altshuler, *Analysis of UXO and clutter signatures from the DARPA background clutter experiment*, in Proceedings of UXO Forum '98, May 1998.

[15] M. Rozložník and Z. Strakoš, *Variants of the residual minimizing Krylov space methods*, in Proceedings of the XI-th Summer School Software and Algorithms of Numerical Mathematics, I. Marek, ed., University of West Bohemia, Plzen, 1995, pp. 208–225.

[16] Y. Saad, *On the lanczos method for solving symmetric linear systems with several right hand sides*, Math. Comp., 48 (1987), pp. 651–662.

[17] ———, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, 1996.

[18] V. Simoncini, *A stabilized QMR version of block BiCG*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 419–434.

[19] 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.

[20] C. F. Smith, *The performance of preconditioned iterative methods in computational*, PhD thesis, Univ. of Illinois at Urbana-Champaign, 1987.

[21] C. F. Smith, A. F. Peterson, and R. Mittra, *A conjugate gradient algorithm for the treatment of multiple incident electromagnetic fields*, IEEE Transactions on Antennas and Propagation, 37 (1989), pp. 1490–1493.