Electronic Transactions on Numerical Analysis.
Volume 41, pp. 478-496, 2014.

ISSN 1068-9613.

ETNA
Kent State University
http://etna.math.kent.edu

A DEFLATED BLOCK FLEXIBLE GMRES-DR METHOD FOR LINEAR SYSTEMS WITH MULTIPLE RIGHT-HAND SIDES*

JING MENG†, PEI-YONG ZHU†, HOU-BIAO LI†, AND XIAN-MING GU†

Abstract. This study is mainly focused on the iterative solution of multiple linear systems with several right-hand sides. To solve such systems efficiently, we first present a flexible version of block GMRES with deflation of eigenvalues according to [R. B. Morgan, Restarted block-GMRES with deflation of eigenvalues, Appl. Numer. Math., 54 (2005), pp. 222–236] and then apply a modified block Arnoldi vector deflation technique to accelerate the convergence of this new flexible version. Incorporating this deflation technique, the new algorithm can address the possible linear dependence at each iteration during the block Arnoldi procedure and reduce computational expense. Moreover, by analyzing its main mathematical properties, we show that the vector deflation procedure arises from the non-increasing behavior of the singular values of the block residual. In addition, the new approach also inherits the property of deflating small eigenvalues to mitigate convergence slowdown. Finally, the effectiveness of the proposed method is illustrated by some numerical experiments.

Key words. deflated BFGMRES-DR, block Krylov subspace, modified block Arnoldi vector deflation, harmonic Ritz vectors, deflated block flexible Arnoldi procedure, multiple right-hand sides

AMS subject classifications. 65F10,65F50

1. Introduction.

Consider the linear system with p right-hand sides

$$AX = B, \tag{1.1}$$

where $A \in \mathbb{C}^{n\times n}$ is a large nonsingular matrix, $B \in \mathbb{C}^{n\times p}$ has full rank, and $X \in \mathbb{C}^{n\times p}$, where $(p \ll n)$. Linear systems with multiple right-hand sides arise in many applications, e.g., electromagnetic scattering [34], model reduction in circuit simulation [13], Quantum Chromo Dynamics QCD [2, 3, 4, 31], and dynamics of structures [8], etc.

At present, there has been substantial interest in developing block Krylov solvers for the solution of the problem (1.1). This is due to the fact that block Krylov subspaces can enlarge the search space, which makes all Krylov subspaces associated with each right-hand side contained. Moreover, the ability of using level-3 BLAS operations also makes block solvers much more competitive than non-block methods from a computational point of view. Methods based on the Lanczos process have been developed to solve (1.1) such as block CG [28], block QMR [14], block BiCGstab [11], block LSQR [18], and a recently proposed block IDR(s) method [10]. In addition, methods based on the Arnoldi process [1, 30], e.g., block restarted GMRES (BGMRES(m)) and its variants [6, 7, 9, 16, 19, 22, 25, 29, 33], were also proposed. Nevertheless, due to restarting, the convergence of a block method based on the Arnoldi process may stagnate and becomes slow. To retain the convergence rate, block GMRES with deflation of eigenvalues (BGMRES-DR) [25] was recently proposed by Morgan. This method exploits a deflation technique to remove (or deflate) small eigenvalues at each restart to improve convergence. For convenience, this technique is referred to as 'eigenvalue deflation' throughout this paper.

The BGMRES-DR method [25] is a well-established block Krylov subspace method for linear systems with multiple right-hand sides. In this paper, we extend the BGMRES-DR method to the case of variable preconditioning, which allows different preconditioning

*Received December, 27, 2012. Accepted July, 21, 2014. Published online on December 19, 2014. Recommended by James Baglama. Supported by the National Natural Science Foundation of China (11026085, 11101071, 10671134, 51175443).

†School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, P. R. China (mengmeng-erni@163.com, zpy6940@sina.com, lihoubiao0189@163.com, guxianming@live.cn).

ETNA
Kent State University
http://etna.math.kent.edu

(possibly nonlinear) operators at each step of the algorithm. In particular, an inexact solution of the preconditioned systems is considered as well as the use of an (inner) iterative method as a preconditioner. The resulting method is called block flexible GMRES with deflation of eigenvalues (BFGMRES-DR).

When block Krylov solvers are used in practice, it is common to come across a possible linear dependence of some columns of the block residuals. An attractive idea should be a combination of BFGMRES-DR with a skill to address such a dependence during the block iterative procedure. A simple technique is to delete linearly or almost linearly dependent vectors from the subspace explicitly. It is also called deflation [16]. To distinguish from eigenvalue deflation, we will refer to this technique as 'vector deflation'. Nevertheless, vector deflation may also lead to a loss of information that slows down the convergence [19]. To remedy this situation, Robbé and Sadkane kept the almost linearly dependent vectors and reintroduced them in the next iterations if necessary during the block Arnoldi procedure; for more details, see [29]. We call this technique modified block Arnoldi vector deflation throughout this paper.

Modified block Arnoldi vector deflation technique has shown great potential to improve the convergence and reduce computational costs for block Krylov subspace solvers [6, 16, 29] in many cases without dramatically increasing the memory requirements. Therefore, if we can combine BFGMRES-DR with this modified vector deflation technique, we will have an effective method which not only allows eigenvalue deflation and variable preconditioning but also addresses the possible linear dependence in the block Krylov subspace. This new approach is referred to as deflated BFGMRES-DR (DBFGMRES-DR).

The main contributions of this paper can be summarized as follows. First we derive the DBFGMRES-DR method by exploiting modified block Arnoldi vector deflation technology. Second, we analyze its main mathematical properties and then show that the deflation procedure is mainly based on a non-increasing behavior of the singular values of the block residual.

The structure of the paper is as follows. In Section 2, we recall some fundamental properties of block Krylov subspaces from [16, 17]. A flexible version of the BGMRES-DR algorithm is presented in Section 3. We describe in detail the DBFGMRES-DR method by exploiting modified block Arnoldi vector deflation technology in Section 4. In Section 5, we demonstrate the effectiveness of the proposed method. Finally, conclusions are summarized in Section 6.

2. Block Krylov subspace.

In this section, we first introduce some notation and definitions used in the remainder of this paper and then recall some fundamental properties of block Krylov subspaces from [16, 17].

2.1. Notations and definitions.

Throughout this paper, $\|.\|_2$ and $\|.\|_F$ denote the Euclidean norm and Frobenius norm, respectively. We use $.^H$ to refer to the conjugate transpose operation of a vector or matrix, the identity matrix of order k is designated as $I_k \in \mathbb{C}^{k\times k}$, and $0_{i\times j}$ is defined as the zero rectangular matrix with i rows and j columns. $\mathcal{R}(\cdot)$ and tol denote the range of the matrix and the convergence threshold, respectively. If $C \in \mathbb{C}^{k\times l}$ is a rectangular matrix ($k > l$), we denote by $C = U\Sigma W^H$ the singular value decomposition (SVD) of C, where $U \in \mathbb{C}^{k\times l}$, $W \in \mathbb{C}^{l\times l}$ are unitary, and $\Sigma = \mathrm{diag}(\sigma_1(C), \sigma_2(C), \ldots, \sigma_l(C)) \in \mathbb{C}^{l\times l}$ is diagonal with $\sigma_1(C) \geq \sigma_2(C) \geq \ldots \geq \sigma_l(C)$. In addition, MATLAB notation is used, for example, $U(1:i, 1:j)$ denotes the submatrix of the first i rows and the first j columns of U, $U(:, j)$ refers to its j-th column, and $U(i, j)$ corresponds to the $U_{i,j}$ entry of the matrix U.

ETNA
Kent State University
http://etna.math.kent.edu

DEFINITION 2.1 (Harmonic Ritz pair [15]). *Consider a subspace $\mathcal{U}$ of $\mathbb{C}^n$. Given $B \in \mathbb{C}^{n\times n}$, $\theta \in \mathbb{C}$, and $y \in \mathcal{U}$. Then (θ, y) is a harmonic Ritz pair of B with respect to $\mathcal{U}$ if and only if*

$$By - \theta y \perp B\mathcal{U},$$

or, equivalently for the canonical scalar product,

$$\forall w \in range(B\mathcal{U}), \quad w^H(By - \theta y) = 0.$$

We call y a harmonic Ritz vector associated with the harmonic Ritz value θ.

2.2. Block Krylov subspace. Let $X_0 \in \mathbb{C}^{n\times p}$ be the initial block guess and $R_0 = B - AX_0$ be the corresponding initial block residual. The block Krylov subspace generated by A from R_0 is defined as follows

$$\mathcal{K}_m(A, R_0) = \text{ span}\{R_0, AR_0, A^2R_0, \ldots, A^{m-1}R_0\}.$$

As mentioned in [16], the definition of 'block span' does not mean a linear combination of the block matrices, i.e., $\sum_{i=0}^{m-1} A^i R_0 \alpha_i$ for some scalars α_i's $\in \mathbb{C}$. Instead, it is a linear combination of all the $m \times p$ columns in $\{R_0, AR_0, A^2R_0, \ldots, A^{m-1}R_0\}$. To clarify this point, we give the following definition

$$\mathcal{K}_m(A, R_0) = \left\{ \sum_{i=0}^{m-1} A^i R_0 \gamma_i, \ \forall \gamma_i \in \mathbb{C}^{p\times p}, \ 0 \le i \le m-1 \right\} \subset \mathbb{C}^{n\times p}.$$

Then the approximate solution $X_m \in \mathbb{C}^{n\times p}$ generated by a block iterative method satisfies

$$X_m - X_0 \in \left\{ \sum_{i=0}^{m-1} A^i R_0 \gamma_i, \ \forall \gamma_i \in \mathbb{C}^{p\times p}, \ 0 \le i \le m-1 \right\} \subset \mathbb{C}^{n\times p}.$$

Note that each column of X_m satisfies

$$\begin{aligned} X_m(:,\ell) - X_0(:,\ell) &\in \left\{ \sum_{i=0}^{m-1} \sum_{j=1}^{p} A^i R_0(:,j)\gamma_i(j,\ell), \ \gamma_i(j,\ell) \in \mathbb{C}, \ \forall 1 \le \ell \le p \right\} \\ &\in \sum_{j=1}^{p} \mathcal{K}_m(A, R_0(:,j)), \end{aligned}$$

where $\mathcal{K}_m(A, R_0(:,j)) = \text{ span}\{R_0(:,j), AR_0(:,j), \ldots, A^{m-1}R_0(:,j)\}$. Unlike the standard Krylov solvers, the search space of block Krylov methods for each right-hand side is much larger, i.e., approximate solutions $X_m(:,\ell)$ are sought in the subspace $\sum_{j=1}^{p} \mathcal{K}_m(A, R_0(:,j))$ rather than in $\mathcal{K}_m(A, R_0(:,j))$, which hopefully leads to a reduction in terms of iteration count. This is the main reason for using block solvers.

Similarly to the standard Krylov subspace, a generalization of the block grade for the block Krylov space was discussed in [17].

3. Block flexible GMRES-DR. In this section, we propose a flexible version of BGMRES-DR (BFGMRES-DR) that combines the numerical features of BGMRES-DR and the flexibility property of FGMRES-DR [15].

We first recall the block flexible Arnoldi process [7] applied to the matrix A and starting with the $n \times p$ orthonormal matrix V_1. Let us denote by M_j the preconditioning operator used

ETNA
Kent State University
http://etna.math.kent.edu

at the j-th iteration. The block flexible Arnoldi algorithm [7] recursively constructs the block matrices $V_1, V_2, \ldots, V_m$ such that $\{V_1, V_2, \ldots, V_m\}$ is an orthonormal basis of the subspace $\{V_1, AZ_1, \ldots, AZ_{m-1}\}$, where $Z_j = M_j^{-1}V_j$ represents the preconditioning operation at iteration j ($1 \leq j \leq m$). At the end of the m-th iteration, a typical relation (block flexible Arnoldi relation) is obtained, that is,

$$A\mathcal{Z}_m = \mathcal{V}_{m+1}\underline{\mathcal{H}}_m, \tag{3.1}$$

where $\mathcal{Z}_m = [M_1^{-1}V_1, \ldots, M_m^{-1}V_m] \in \mathbb{C}^{n\times mp}$, $\mathcal{V}_{m+1} = [V_1, V_2, \ldots, V_{m+1}] \in \mathbb{C}^{n\times(m+1)p}$, and $\underline{\mathcal{H}}_m \in \mathbb{C}^{(m+1)p\times mp}$ has the following form

$$\underline{\mathcal{H}}_m = \begin{bmatrix} \mathcal{H}_m \\ H_{m+1,m}E_m^H \end{bmatrix}.$$

Note that

$$\mathcal{H}_m = \begin{bmatrix} H_{1,1} & H_{1,2} & \ldots & \ldots & H_{1,m} \\ H_{2,1} & H_{2,2} & & & \vdots \\ & H_{3,2} & \ddots & & \vdots \\ & & \ddots & \ddots & \vdots \\ & & & H_{m-1,m} & H_{m,m} \end{bmatrix} \in \mathbb{C}^{mp\times mp}$$

is supposed to be nonsingular with the $p\times p$ smaller matrix $H_{i,j}$ and $E_m = [0_{(m-1)p\times p},\ I_p]^H$. For simplicity of discussion, each pass through the block Arnoldi iteration between restarts is referred to as one "cycle".

Suppose that the block flexible Arnoldi relation (3.1) holds. In the following, we will deflate eigenvalues of smallest magnitude over the subspace $\mathcal{R}(A\mathcal{Z}_m)$ to improve convergence. This technique is similar to the one used in the BGMRES-DR method [25] (or the FGMRES-DR method [15]), which retains an approximate invariant subspace between cycles. In particular, it focuses on removing (or deflating) the eigenvalues of smallest magnitude by recycling an approximate invariant subspace associated with those eigenvalues. This approximate invariant subspace is constructed by harmonic Ritz vectors of A computed at the end of the previous cycle. Unlike the case of a fixed preconditioner, the eigenvalue deflation procedure for the flexible setting relies on harmonic Ritz vectors of $A\mathcal{Z}_m^H\mathcal{V}_m$ with respect to $\mathcal{R}(\mathcal{V}_m)$. The following theorem presents the harmonic Ritz formulation used in the BFGMRES-DR method.

THEOREM 3.1. *Block flexible GMRES-DR relies on the computation of k harmonic Ritz vectors $\tilde{Y}_k = \mathcal{V}_m G_k$ with $\mathcal{V}_m \in \mathbb{C}^{n\times mp}$ and $G_k = [g_1, \ldots, g_k] \in \mathbb{C}^{mp\times k}$, where each harmonic Ritz pair (θ_i, g_i) satisfies*

$$(\mathcal{H}_m + H_{m+1,m}^H H_{m+1,m}\mathcal{H}_m^{-H}E_mE_m^H)g_i = \theta_i g_i,\ 1 \leq i \leq k. \tag{3.2}$$

$\tilde{Y}_k$ corresponds to harmonic Ritz vectors of $A\mathcal{Z}_m\mathcal{V}_m^H$ with respect to $\mathcal{R}(\mathcal{V}_m)$, and the harmonic residual vectors $A\mathcal{Z}_m\mathcal{V}_m^H\mathcal{V}_m g_i - \theta_i\mathcal{V}_m g_i \in \mathcal{R}(\mathcal{V}_{m+1})$ are orthogonal to an mp-dimensional subspace spanned by the columns of $A\mathcal{Z}_m$.

Proof. The proof follows straightforwardly from [15, Proposition 1]. ∎

Let R_0 be the block residual from the previous cycle or, equivalently, the initial block residual for the new cycle, which can be stated as $R_0 = \mathcal{V}_{m+1}\hat{R}_m$, where $\hat{R}_m$ denotes the representation of the block residual in the $\mathcal{V}_{m+1}$ basis. Observe that the block residual R_0 resides in the subspace $\mathcal{R}(\mathcal{V}_{m+1})$ and is also orthogonal to the subspace $\mathcal{R}(A\mathcal{Z}_m)$. So the harmonic

ETNA
Kent State University
http://etna.math.kent.edu

residuals and the block residual are in the same p-dimensional space $\mathcal{R}(A\mathcal{Z}_m)^{\perp}\cap\mathcal{R}(\mathcal{V}_{m+1})$, and the harmonic residuals are all linear combinations of the columns of the block residual. We next characterize the relationship between the harmonic residuals and the block residual by the following formulae

$$A\mathcal{Z}_m G_k = \mathcal{V}_{m+1}\begin{bmatrix}\begin{bmatrix} G_k \\ 0_{p\times k}\end{bmatrix} & \hat{R}_m\end{bmatrix}\begin{bmatrix}\operatorname{diag}(\theta_1,\ldots,\theta_k)\\ a_{p\times k}\end{bmatrix},$$

where each column of the matrix a satisfies $A\mathcal{Z}_m g_i - \theta_i \mathcal{V}_m g_i = \sum_{j=1}^{p} R_0(:,j)a(j,i)$, $1\le i\le k$. Since BFGMRES-DR is a natural extension of the FGMRES-DR method [15], a block flexible Arnoldi-like relation can be derived analogous to that in [15, Section 3.1.2]. Therefore, we compute the reduced QR factorization of $G_k = P_k\Gamma_k$ and then orthonormalize the block matrix $\hat{R}_m$ against the columns of $\begin{bmatrix} P_k \\ 0_{p\times k}\end{bmatrix}$ to obtain the orthonormal matrix $[p_{k+1}, p_{k+2},\ldots,p_{k+p}]\in\mathbb{C}^{(m+1)\times p}$ satisfying

$$A\mathcal{Z}_m P_k = \mathcal{V}_{m+1}P_{k+1}\Gamma_{k+1}\begin{bmatrix}\operatorname{diag}(\lambda_1,\ldots,\lambda_k)\\ a_{p\times k}\end{bmatrix}\Gamma_k^{-1} \tag{3.3}$$

with $P_{k+1} = \begin{bmatrix}\begin{bmatrix} P_k \\ 0_{p\times k}\end{bmatrix} & p_{k+1},\ p_{k+2},\ \ldots,\ p_{k+p}\end{bmatrix}$ and $\Gamma_{k+1} = \begin{bmatrix}\begin{bmatrix} \Gamma_k \\ 0_{p\times k}\end{bmatrix} & u_{p\times k}\end{bmatrix}$.

Thus, by (3.1) and $P_{k+1}^H P_{k+1} = I_{k+p}$, we obtain

$$H_k^{new} = \Gamma_{k+1}\begin{bmatrix}\operatorname{diag}(\lambda_1,\ldots,\lambda_k)\\ a_{1\times k}\end{bmatrix}\Gamma_k^{-1} = P_{k+1}^H\underline{\mathcal{H}}_m P_k, \tag{3.4}$$

where H_k^{new} is a $(k+p)\times k$ rectangular matrix. Let $Z_k^{new} = \mathcal{Z}_m P_k$ and $V_{k+p}^{new} = \mathcal{V}_{m+1}P_{k+1}$. Combining the conditions (3.3) and (3.4) yields

$$\begin{aligned} AV_k^{new} &= V_{k+p}^{new}H_k^{new},\\ V_{k+p}^{new\,H}V_{k+p}^{new} &= I_{k+p},\\ \mathcal{R}([Y_k, R_0]) &= \mathcal{R}(V_{k+p}^{new}).\end{aligned} \tag{3.5}$$

We can consequently see that the block Arnoldi-like recurrence formulae (3.5) can be recovered without involving any matrix-vector product with A when restarting with some harmonic Ritz vectors.

Assuming that k is divisible by p, then we carry out $m-\frac{k}{p}$ steps of the block flexible Arnoldi process with the starting block matrix $V_{k+p}^{new}(:,k+1:k+p)$ to eventually obtain

$$\begin{aligned} A\mathcal{Z}_m &= \mathcal{V}_{m+1}\underline{\mathcal{H}}_m,\\ {\mathcal{V}_{m+1}}^H\mathcal{V}_{m+1} &= I_{(m+1)p},\end{aligned}$$

where $\mathcal{Z}_m = [Z_1, Z_2,\ldots,Z_m]$ with $Z_i = Z_k^{new}(:,(i-1)p+1:ip)$, for $1\le i\le\frac{k}{p}$, and $Z_i = M_i^{-1}V_{i+1}$, for $\frac{k}{p}+1\le i\le m$, $\mathcal{V}_{m+1} = [V_1, V_2,\ldots,V_{m+1}]$ with $V_i = V_{k+p}^{new}(:,(i-1)p+1:ip)$, for $1\le i\le\frac{k}{p}+1$ and $\underline{\mathcal{H}}_m$ now is an upper block Hessenberg matrix, except for a full leading $k+p$ by k portion. At the end of the cycle the approximate solution $X_m = X_0 + \mathcal{Z}_m Y_m\in\mathbb{C}^{n\times p}$ is found by minimizing the residual norm $\|R_0 - A\mathcal{Z}_m Y\|_F$, where Y_m is the solution of the following least-squares problem of size $(m+1)p\times mp$,

$$Y_m = \underset{Y\in\mathbb{C}^m}{\operatorname{argmin}}\|R_0 - A\mathcal{Z}_m Y\|_F = \underset{Y\in\mathbb{C}^m}{\operatorname{argmin}}\left\|\hat{\Lambda}_0 - \underline{\mathcal{H}}_m Y\right\|_F,$$

ETNA
Kent State University
http://etna.math.kent.edu

Algorithm 1 BFGMRES-DR.

Input: $A \in \mathbb{C}^{n\times n}$, $B \in \mathbb{C}^{n\times p}$. Choose an initial guess X_0 and $tol > 0$.
Output: $X_m \in \mathbb{C}^{n\times p}$ with $X_m \approx A^{-1}B$.

1: Compute the block residual $R_0 = B - AX_0$. Compute the reduced QR decomposition of $R_0 = V_1 R$. Generate $\mathcal{V}_{m+1}$ and $\underline{\mathcal{H}}_m = \begin{bmatrix} \mathcal{H}_m \\ H_{m+1,m}E_m^H \end{bmatrix}$ with the block Arnoldi process. Set $\hat{\Lambda}_0 = [R^H, 0, \ldots, 0]^H$.
2: Solve $\min \left\| \hat{\Lambda}_0 - \underline{\mathcal{H}}_m Y \right\|$ for Y_m. Set $X_m = X_0 + \mathcal{V}_m Y_m$, $R_m = B - AX_m$. Check residual norms for convergence and proceed if not satisfied.
3: Compute the k smallest eigenpairs (θ_j, g_j) of $(\mathcal{H}_m + H_{m+1,m}^H H_{m+1,m}\mathcal{H}_m^{-H} E_m E_m^H)$.
4: Orthonormalize the vectors g_i by first separating them into real and imaginary parts if they are complex to form the columns of $P_k \in \mathbb{R}^{mp\times k}$. (It may be necessary to adjust k to include both the real and imaginary parts of complex eigenvectors.)
5: Extend the vectors $p_1, \ldots, p_k$ to length $mp+p$ with zero entries, then orthonormalize the columns of $\hat{R}_m = \hat{\Lambda}_0 - \underline{\mathcal{H}}_m Y_m$ against the columns of $\begin{bmatrix} P_k \\ 0 \end{bmatrix}$ to form $p_{k+1}, \ldots, p_{k+p}$. Set $P_{k+1} = \begin{bmatrix} P_k & p_{k+1} & \ldots & p_{k+p} \\ 0 & & & \end{bmatrix}$.
6: Set $Z_k^{new} = \mathcal{Z}_m P_k$, $V_{k+p}^{new} = \mathcal{V}_{m+1}P_{k+1}$ and $H_k^{new} = P_{k+1}^H \underline{\mathcal{H}}_m P_k$. Apply $m - \frac{k}{p}$ steps of the block flexible Arnoldi process to extend Z_k^{new}, V_{k+p}^{new} and H_k^{new} to $\mathcal{Z}_m$, $\mathcal{V}_{m+1}$ and $\underline{\mathcal{H}}_m$.
7: Let $\hat{\Lambda}_0 = \mathcal{V}_{m+1}^H R_m$ and $X_0 = X_m$. Go to step 2.

where $\hat{\Lambda}_0 = \mathcal{V}_{m+1}^H R_0$. Details of the BFGMRES-DR method are given in Algorithm 1.

As mentioned above, the BFGMRES-DR method exploits the eigenvalue deflation technique to mitigate convergence slowdown. In the next section, we will adopt another deflation technique to address the possible linear dependence occurring in the block Krylov space.

4. Deflated BFGMRES-DR. When block Krylov solvers are used in practice, the trouble comes from the possible linear dependence of some columns of the block residuals. Such a dependence implies that the matrix $[R_0, AZ_1, \ldots, AZ_{m-1}]$ is almost rank deficient. In such a situation, the block Arnoldi procedure does not continue as usual, and the computational cost will be expensive. To remedy this situation, Robbé and Sadkane [29] kept these almost linearly dependent vectors and reintroduced them in the next iteration if necessary rather than deleting them explicitly. To distinguish from eigenvalue deflation, we refer to this technique as modified block Arnoldi vector deflation throughout this paper, while the block flexible Arnoldi process with this deflation strategy is defined as deflated block flexible Arnoldi procedure [29]. In addition, they proposed two criteria either based on the numerical rank of the block Krylov basis (W-criterion) or on the numerical rank of the block residual (R-criterion), to detect dependence. The R-criterion is used to judiciously decompose the block residual into two parts. One part stores vectors ensuring convergence, the other keeps the deflated vectors. Recently, Calandra et al. [6] extended this concept by performing an additional decomposition at the beginning of each cycle. Inspired by these ideas in [6, 29], we next apply the modified block Arnoldi vector deflation strategy to the BFGMRES-DR method.

4.1. Deflated block flexible Arnoldi procedure. The j-th iteration of the deflated block flexible Arnoldi procedure [6] is briefly reviewed as follows. Assume that an orthonormal matrix $K \in \mathbb{C}^{n\times p}$ contains all the p Krylov directions after iteration $j-1$. However, all the

ETNA
Kent State University
http://etna.math.kent.edu

p Krylov directions in the subspaces of $\mathcal{R}(K)$ may not be needed for ensuring convergence along the iterative procedure. In order to reduce unnecessary computational cost, the R-criterion is considered to judiciously decompose $\mathcal{R}(K)$ into two parts:

$$\mathcal{R}(K) = \mathcal{R}(V_j) \oplus \mathcal{R}(P_j), \quad \text{with } [V_j, P_j]^H [V_j, P_j] = I_p,$$

where $V_j \in \mathbb{C}^{n\times k_j}$, $P_j \in \mathbb{C}^{n\times d_j}$ with $k_j + d_j = p$. Then k_j Krylov directions ensuring convergence (stored in V_j) will be effectively considered at iteration j and multiplied by A, while d_j directions (stored in P_j) do not participate in the matrix-vector product process. Langou [19] showed that deleting the deflated space $\mathcal{R}(P_j)$ is not recommended since it may lead to a loss of information that slows down the convergence. So the deflated part is not deleted but left aside at the same iteration and then reintroduced in the next iteration if necessary. In addition, AV_j is orthogonal to all the previous Krylov directions $V_i (1 \le i \le j)$ and P_j. The algorithmic details of the deflated block flexible Arnoldi procedure are described in Algorithm 2.

Algorithm 2 Deflated block flexible Arnoldi procedure [6]: computation of $\hat{V}_{j+1}$ with $[\mathcal{V}_j, P_j]^H [\mathcal{V}_j, P_j] = I_{s_{j-1}+p}$.

Input: $[\mathcal{V}_j, P_j] \in \mathbb{C}^{n\times(s_{j-1}+p)}$ with $\mathcal{V}_j = [V_1, V_2, \ldots, V_j]$, $V_j \in \mathbb{C}^{n\times k_i}$ such that $V_j^H V_j = I_{k_j}$, $P_j \in \mathbb{C}^{n\times d_j}$, and $k_j + d_j = p$.

Output: $\hat{V}_{j+1} \in \mathbb{C}^{n\times k_j}$ orthonormal columns, $\begin{bmatrix} H_j \\ H_{j+1,j} \end{bmatrix} \in \mathbb{C}^{(s_j+p)\times k_j}$ with $AZ_j = [\mathcal{V}_j, P_j, \hat{V}_{j+1}] \begin{bmatrix} H_j \\ H_{j+1,j} \end{bmatrix}$.

1: Define $s_{j-1} = \sum_{l=1}^{j-1} k_l$ with $s_0 = 0$.
2: Compute $Z_j = M_j^{-1} V_j$.
3: Compute $W = AZ_j$.
4: **for** $i = 1, \ldots, j$ **do**
5: $\quad H_{i,j} = V_i^H W$
6: $\quad W = W - V_i H_{i,j}$
7: **end for**
8: $H_p = P_j^H W$
9: $W = W - P_j H_p$
10: $H_j \in \mathbb{C}^{(s_{j-1}+p)\times k_j}$ as $H_j = \begin{bmatrix} H_{1,j} \\ \vdots \\ H_{j,j} \\ H_p \end{bmatrix}$
11: Compute the reduced QR decomposition of W as $W = QR$, $Q \in \mathbb{C}^{n\times k_j}$ and $R \in \mathbb{C}^{k_j \times k_j}$.
12: Set $\hat{V}_{j+1} = Q$, $H_{j+1,j} = R$.
13: Define s_j as $s_j = s_{j-1} + k_j$.
14: Define $\mathcal{V}_j \in \mathbb{C}^{n\times s_j}$ as $\mathcal{V}_j = [V_1, V_2, \ldots, V_j]$ and $\hat{\mathcal{V}}_{j+1} \in \mathbb{C}^{n\times(s_j+p)}$ as $\hat{\mathcal{V}}_{j+1} = [\mathcal{V}_j, P_j, \hat{V}_{j+1}]$ such that $AZ_j = [\mathcal{V}_j, P_j, \hat{V}_{j+1}] \begin{bmatrix} H_j \\ H_{j+1,j} \end{bmatrix}$.

The deflated orthogonalization procedure leads to the following relation, for $1 \le j \le m$,

$$AZ_j = \hat{\mathcal{V}}_{j+1} \begin{bmatrix} H_j \\ H_{j+1,j} \end{bmatrix}. \tag{4.1}$$

ETNA
Kent State University
http://etna.math.kent.edu

Assume that the deflated block flexible Arnoldi relation holds at the beginning of the j-th iteration

$$A\mathcal{Z}_{j-1} = [\mathcal{V}_j, P_j]\underline{\mathcal{H}}_{j-1} \tag{4.2}$$

with $[\mathcal{V}_j, P_j]^H[\mathcal{V}_j, P_j] = I_{(s_{j-1}+p)}$ and $\underline{\mathcal{H}}_{j-1} \in \mathbb{C}^{(s_{j-1}+p)\times s_{j-1}}$. We can rewrite (4.1) together with (4.2) as a deflated block flexible Arnoldi-like relation,

$$[A\mathcal{Z}_{j-1}, AZ_j] = [\mathcal{V}_j, P_j, \hat{V}_{j+1}] \left[\begin{array}{c|c} \underline{\mathcal{H}}_{j-1} & H_j \\ \hline 0_{k_j\times s_{j-1}} & H_{j+1,j} \end{array}\right], \tag{4.3}$$

$$A\mathcal{Z}_j = \hat{\mathcal{V}}_{j+1}\hat{\underline{\mathcal{H}}}_j.$$

Next, the R-criterion [29] is considered to decompose the subspace $\mathcal{R}([P_j, \hat{V}_{j+1}])$ into two parts $[V_{j+1}, P_{j+1}]$. Suppose that the subspace decomposition can be stated as

$$[V_{j+1}, P_{j+1}] = [P_j, \hat{V}_{j+1}]F_{j+1}, \tag{4.4}$$

where $F_{j+1} \in \mathbb{C}^{p\times p}$ is a unitary matrix. We can rewrite (4.3) together with (4.4) as a deflated block flexible Arnoldi relation

$$A\mathcal{Z}_j = [\mathcal{V}_j, P_j, \hat{V}_{j+1}]\mathcal{F}_{j+1}\mathcal{F}_{j+1}^H\hat{\underline{\mathcal{H}}}_j = [\mathcal{V}_{j+1}, P_{j+1}]\underline{\mathcal{H}}_j,$$

where $\mathcal{F}_{j+1} = \begin{bmatrix} I_{s_j} & 0_{s_j\times p} \\ 0_{p\times s_j} & F_{j+1} \end{bmatrix}$. So, the deflated block flexible Arnoldi procedure can continue as usual. The way to choose the matrix F_{j+1} will be considered in Section 4.2.

In the following, we present a framework for the new method named deflated BFGMRES-DR (DBFGMRES-DR) based on the deflated block flexible Arnoldi relation, which allows eigenvalue deflation, vector deflation, and variable preconditioning simultaneously. Since the eigenvalue deflation technique is not performed at the first cycle, the new method carries out m steps of the deflated block flexible Arnoldi process starting with the initial block residual. After the first cycle, we carry out step (3.2) to compute the k harmonic Ritz vectors and then construct a new block flexible Arnoldi-like recurrence (3.5). Note that the harmonic Ritz vectors added to the next subspace are independent. Therefore, the algorithm only runs $m - \frac{k}{p}$ steps of the deflated block flexible Arnoldi process with the starting block matrix $V_{k+p}^{new}(:, k+1:p)$. Consequently, it only requires a total of $s_m - k$ (or s_m for the first cycle) matrix-vector products with A to construct a new orthogonal basis for the deflated block Krylov subspace of dimension of $s_m + d_m$ ($\leq mp$), including k harmonic Ritz vectors. The details of the DBFGMRES-DR approach are described in Algorithm 3.

REMARK 4.1. If we replace M_j with I_n, i.e., without using a preconditioner, the above algorithm reduces to the deflated BGMRES-DR method (referred to as DBGMRES-DR), which relies on the deflated block Arnoldi relation. The numerical behavior concerning DBGMRES-DR is presented in Section 5.

4.2. Details of the modified block Arnoldi vector deflation. In this section, we briefly introduce Robbé and Sadkane's R-criterion [29] that is used to detect dependencies and identify the linearly independent vectors ensuring convergence in the deflated block flexible Arnoldi procedure.

Let $X_j \in \mathbb{C}^{n\times p}$ be an approximation generated by DFBGMRES-DR at iteration j. The corresponding residual is given by

$$\begin{aligned} R_j &= B - AX_j = R_0 - A\mathcal{Z}_jY_j = R_0 - \mathcal{V}_{j+1}\underline{\mathcal{H}}_jY_j = R_0 - [\mathcal{V}_j, P_j, \hat{V}_{j+1}]\hat{\underline{\mathcal{H}}}_jY_j \\ &= [\mathcal{V}_j, P_j, \hat{V}_{j+1}](\hat{\Lambda}_j - \hat{\underline{\mathcal{H}}}_jY_j) = [\mathcal{V}_j, P_j, \hat{V}_{j+1}]\hat{R}_j, \end{aligned}$$

ETNA
Kent State University
http://etna.math.kent.edu

Algorithm 3 Deflated BFGMRES-DR algorithm.

Input: $A \in \mathbb{C}^{n\times n}$, $B \in \mathbb{C}^{n\times p}$. Choose an initial guess X_0 and $tol > 0$.
Output: $X_m \in \mathbb{C}^{n\times p}$ with $X_m \approx A^{-1}B$.

1: Computer $R_0 = B - AX_0$.
2: Compute the QR decomposition of R_0 as $R_0 = \hat{\mathcal{V}}_1\hat{\Lambda}_0$ such that $p_0 = rank(\hat{\mathcal{V}}_1)$ with $\hat{\mathcal{V}}_1 \in \mathbb{C}^{n\times p_0}$ and $\hat{\Lambda}_0 \in \mathbb{C}^{p_0\times p}$; Set $s_0 = 0$, $j = 1$.
3: **for** $cycle = 1, \ldots$ **do**
4: Use Algorithm 4 to determine deflation unitary matrix $\mathcal{F}_j$ and k_j, d_j such that $k_j + d_j = p$. Set $s_1 = k_1$ if $j = 1$ or $s_j = k$ if $j = \frac{k}{p} + 1$.
5: Define $[\mathcal{V}_j, P_j] = \hat{\mathcal{V}}_j\mathcal{F}_j$, with $\mathcal{V}_j \in \mathbb{C}^{n\times s_j}(P_j \in \mathbb{C}^{n\times d_j})$ as the first s_j (last d_j) columns of $\hat{V}_j\mathcal{F}_j$.
6: Define $V_j = \mathcal{V}_j$, $\mathcal{H}_j = \mathcal{F}_j^H\underline{\hat{\mathcal{H}}}_j$ and $\Lambda_j = \mathcal{F}_j^H\hat{\Lambda}_{j-1}$ with $\Lambda_j \in \mathbb{C}^{(s_{j-1}+p)\times p}$.
7: **while** $j \le m$, **do**
8: Apply Algorithm 2 to get $\mathcal{V}_j \in \mathbb{C}^{n\times s_j}$, $\hat{\mathcal{V}}_{j+1} \in \mathbb{C}^{n\times(s_j+p)}$, $\underline{\hat{\mathcal{H}}}_j \in \mathbb{C}^{(s_j+p)\times s_j}$ such that $A\mathcal{Z}_j = \hat{\mathcal{V}}_{j+1}\underline{\hat{\mathcal{H}}}_j$ with $\hat{\mathcal{V}}_{j+1} = [V_1, V_2, \ldots, V_j, P_j, \hat{V}_j]$.
9: Set $\hat{\Lambda}_j \in \mathbb{C}^{(s_j+p)\times p}$ as $\hat{\Lambda}_j = \begin{bmatrix}\Lambda_j \\ 0_{k_j\times p}\end{bmatrix}$.
10: Compute $Y_j = \arg\min_{Y\in C^{s_j\times p}} \left\|\hat{\Lambda}_j - \underline{\hat{\mathcal{H}}}_j Y\right\|_F$.
11: Compute $\hat{R}_j = \hat{\Lambda}_j - \underline{\hat{\mathcal{H}}}_j Y_j$.
12: Set $X_j = X_0 + \mathcal{V}_j Y_j$, $R_j = B - AX_j$. Check residual norms for convergence, and proceed if not satisfied.
13: Determine deflation unitary matrix $\mathcal{F}_{j+1} \in \mathbb{C}^{(s_j+p)\times(s_j+p)}$ and k_{j+1}, d_{j+1} such that $k_{j+1} + d_{j+1} = p$.
14: Set $s_{j+1} = s_j + k_{j+1}$.
15: Define $[\mathcal{V}_{j+1}, P_{j+1}] = \hat{\mathcal{V}}_{j+1}\mathcal{F}_{j+1}$, with $V_{j+1} \in \mathbb{C}^{n\times k_{j+1}}$ and $\mathcal{V}_{j+1} \in \mathbb{C}^{n\times s_{j+1}}(P_j \in \mathbb{C}^{n\times d_{j+1}})$ as the first s_{j+1} (last d_{j+1}) columns of $\hat{V}_{j+1}\mathcal{F}_{j+1}$.
16: Define $\underline{\mathcal{H}}_j = \mathcal{F}_{j+1}^H\underline{\hat{\mathcal{H}}}_j$ and $\Lambda_{j+1} = \mathcal{F}_{j+1}^H\hat{\Lambda}_j$ with $\underline{\mathcal{H}}_j \in \mathbb{C}^{(s_j+p)\times s_j}$ and $\Lambda_{j+1} \in \mathbb{C}^{(s_j+p)\times p}$.
17: **end while**
18: Let $H_m \in \mathbb{C}^{p\times s_m}$ be the submatrix of $\underline{\mathcal{H}}_m$ with rows from $s_m + 1$ to $s_m + p$. Let $F = \mathcal{H}_m^{-H}H_m^H$ with $\mathcal{H}_m = \underline{\mathcal{H}}_m(1 : s_m, :)$.
19: Compute the k smallest eigenpairs (θ_i, g_i) of $\mathcal{H}_m + FH_m$.
20: Orthonormalize the vectors g_is to form the columns of $P_k \in \mathbb{C}^{s_m\times k}$.
21: Extend the columns of P_k to length $s_m + p$ with zero entries, then orthonormalize the columns of $\hat{R}_{s_m+p}$ against the columns of $\begin{bmatrix}P_k \\ 0_{(s_m+p)\times k}\end{bmatrix}$ to form $p_{k+1}, p_{k+2}, \ldots, p_{k+p}$, set $P_{k+p} = \begin{bmatrix}P_k & p_{k+1} & \cdots & p_{k+p} \\ 0_{(s_m+p)\times k}\end{bmatrix}$.
22: Set $\mathcal{Z}_k^{new} = \mathcal{Z}_m P_k$ and $\mathcal{V}_{k+p}^{new} = \mathcal{V}_m P_{k+p}$.
23: Set $j = \frac{k}{p} + 1$, $\underline{\hat{\mathcal{H}}}_j = P_{k+p}^H\underline{\mathcal{H}}_m P_k$ and $\hat{V}_j = \mathcal{V}_{k+p}^{new}(:, k+1 : k+p)$.
24: Let $\hat{\Lambda}_{j-1} = [\hat{R}_m^H P_{k+p}, 0]^H$ and $X_0 = X_m$.
25: **end for**

ETNA
Kent State University
http://etna.math.kent.edu

where $\hat{R}_j = \hat{\Lambda}_j - \underline{\hat{\mathcal{H}}}_j Y_j$ is the j-th block quasi-residual. Consider the SVD of the block quasi-residual $\hat{R}_j = U\Sigma W^H$, where $U \in \mathbb{C}^{(s_j+p)\times p}$, $W \in \mathbb{C}^{p\times p}$ have orthonormal columns and $\Sigma \in \mathbb{C}^{p\times p}$ is diagonal. By [7, 29], we can determine a subset of singular values of $\hat{R}_j$ satisfying the following condition:

$$\sigma_\ell(\hat{R}_j) > \varepsilon_d tol, \quad \forall\, \ell \text{ such that } 1 \le \ell \le p_d \ (p_d < p),$$

where ε_d is a real positive parameter smaller than one. This allows us to decompose the matrix Σ as follows,

$$\Sigma = \begin{bmatrix} \Sigma_+ & 0_{p_d\times(p-p_d)} \\ 0_{(p-p_d)\times p_d} & \Sigma_- \end{bmatrix},$$

with $\Sigma_+ = \Sigma(1:p_d, 1:p_d)$ and $\Sigma_- = \Sigma(p_d+1:p, p_d+1:p)$. Thus, the block residual can be written as

$$R_j = [\mathcal{V}_j, P_j, \hat{V}_{j+1}](U_+\Sigma_+ W_+^H + U_-\Sigma_- W_-^H).$$

Since $[\mathcal{V}_j, P_j, \hat{V}_{j+1}]$ is an orthogonal matrix, it is straightforward to obtain

$$\left\| [\mathcal{V}_j, P_j, \hat{V}_{j+1}] U_+\Sigma_+ W_+^H \right\|_2 > \varepsilon_d tol,$$

while

$$\left\| [\mathcal{V}_j, P_j, \hat{V}_{j+1}] U_-\Sigma_- W_-^H \right\|_2 \le \varepsilon_d tol.$$

We set $k_{j+1} = p_d$ and $d_{j+1} = p - p_d$. If $p_d < p$, then there exist d_{j+1} linearly or almost linearly dependent vectors in the generated block Krylov space. In this case, the subspace decomposition will be performed to select the linearly independent vectors (stored in V_{j+1}) and leave aside the linearly dependent vectors (stored in P_{j+1}).

From a practical point of view, the search space $\mathcal{R}(V_{j+1})$ spanned by the linearly independent vectors should satisfy $\mathcal{R}(V_{j+1}) \subset \mathcal{R}([\mathcal{V}_j, P_j, \hat{V}_{j+1}]U_+)$. Moreover, the orthonormal relation $V_{j+1} \perp \mathcal{V}_j$ is required. This can be expressed as

$$\begin{aligned} \mathcal{R}(V_{j+1}) &= \mathcal{R}((I - \mathcal{V}_j\mathcal{V}_j^H)[\mathcal{V}_j, P_j, \hat{V}_{j+1}]U_+) = \mathcal{R}([0_{n\times s_j}, P_j, \hat{V}_{j+1}]\hat{R}_j W_+) \\ &= \mathcal{R}([0_{n\times s_j}, P_j, \hat{V}_{j+1}] \begin{bmatrix} \hat{R}_{s_j} \\ \hat{R}_p \end{bmatrix} W_+) = \mathcal{R}([P_j, \hat{V}_{j+1}]\hat{R}_p W_+). \end{aligned} \tag{4.5}$$

Similarly,

$$\mathcal{R}(P_{j+1}) = \mathcal{R}([P_j, \hat{V}_{j+1}]\hat{R}_p W_-). \tag{4.6}$$

We can rewrite (4.5) together with (4.6) as the following relation

$$\mathcal{R}([V_{j+1}, P_{j+1}]) = \mathcal{R}([P_j, \hat{V}_{j+1}]\hat{R}_p W) = \mathcal{R}([P_j, \hat{V}_{j+1}]F_{j+1}T), \tag{4.7}$$

where F_{j+1} is a $p \times p$ unitary factor of the QR factorization of $\hat{R}_p W$. In view of (4.7), we can keep the first k_{j+1} columns of $[P_j, \hat{V}_{j+1}]F_{j+1}$ as V_{j+1}, while the next d_{j+1} columns are stored in P_{j+1}. For more details, see Algorithm 4.

Note that determining F_{j+1} is rather inexpensive in terms of computational operations since $p \ll n$. Moreover, we conclude that the singular values of the block residual generated by the DBFGMRES-DR are monotonically decreasing.

ETNA
Kent State University
http://etna.math.kent.edu

Algorithm 4 Determination of k_{j+1}, d_{j+1} and of $\mathcal{F}_{j+1}$ [6].

1: Choose a deflation threshold ε_d.
2: Compute the SVD of $\hat{R}_j$ as $\hat{R}_j = U\Sigma W^H$. with $U \in \mathbb{C}^{(s_j+p)\times p}$, $\Sigma \in \mathbb{C}^{p\times p}$ and $W \in \mathbb{C}^{p\times p}$.
3: Select p_d singular values of $\hat{\mathcal{R}}_j$ such that $\sigma_l(\hat{\mathcal{R}}_j) > \varepsilon_d tol$ for all l such that $1 \le l \le p_d$.
4: Set $k_{j+1} = p_d$ and $d_{j+1} = p - k_{j+1}$.
5: Define $\hat{\mathcal{R}}_p \in \mathbb{C}^{p\times p}$ as $\hat{\mathcal{R}}_p = \hat{\mathcal{R}}_j(s_j + 1 : s_j + p, 1 : p)$.
6: Compute the QR factorization: $\hat{\mathcal{R}}_p W$ as $\hat{\mathcal{R}}_p W = F_{j+1}T$ with $F_{j+1} \in C^{p\times p}$ and $F_{j+1}^H F_{j+1} = I_p$.
7: Define $\mathcal{F}_{j+1} \in \mathbb{C}^{(s_j+p)\times(s_j+p)}$ as $\begin{bmatrix} I_{s_j} & 0_{s_j\times p} \\ 0_{p\times s_j} & F_{j+1} \end{bmatrix}$.

THEOREM 4.2. *Let $R_{j,\ell}$ be the block residual at the j-th iteration of the ℓ-th cycle of DFBGMRES-DR. Then the singular values $\sigma_i(R_{j,\ell})$ satisfy the inequality*

$$\sigma_i(R_{j,\ell}) \le \sigma_i(R_{j-1,\ell}), \quad 1 \le i \le p.$$

Proof. The relation $R_{j,\ell} = (I - \mathcal{V}_j\mathcal{V}_j^H)R_{j-1,\ell}$ holds. Therefore the proof follows straightforwardly from [6, Proposition 3]. ∎

Since k_{j+1} is directly determined by the singular values of $R_{j,\ell}$, i.e., $k_{j+1} = p_d$. From Theorem 4.2, we deduce that the sequence of each cycle k_j, $j \ge 1$, is progressively decreasing, which may yield a significant reduction in terms of matrix-vector products and then, hopefully, lead to a reduction in terms of computational operations.

REMARK 4.3. In addition, it is worth mentioning that one difficulty arises when k_j reaches the value 0, and the block solver with vector deflation technique has not satisfied the stopping criterion. In this case, the deflated block flexible Arnoldi procedure cannot continue since $k_j = 0$. Therefore, we investigate a combination of DBFGMRES-DR and BGMRESD(m) as Calandra et al. did in [6]. When k_j reaches the value 0, the DBFGMRES-DR approach is replaced with BFGMRES(m) at the next restart in order to achieve a convergence criterion. We refer to this combination as Combined($m, 0$). Experiments show that Combined($m, 0$) works well and faster convergence behavior is retained; see Section 5.

4.3. Computational cost for the modified block Arnoldi vector deflation. Compared with BFGMRES-DR, additional operations concerning the computation of $[V_{j+1}, P_{j+1}]$, F_{j+1}, Λ_j, and $\underline{\mathcal{H}}_j$ are needed in DBFGMRES-DR. We summarize the additional computational cost for every iteration of DBFGMRES-DR in Table 4.1.

TABLE 4.1
Computational cost for the modified block Arnoldi vector deflation.

Operation	Cost
$\mathcal{F}_{j+1}^H \underline{\hat{\mathcal{H}}}_j$	$2p^3$
$F_{j+1}^H \hat{\Lambda}_j$	$2p^3$
$[P_j, \hat{V}_{j+1}]F_{j+1}, (j \ne 0)$	$2np^2$
Computation of F_{j+1}	$4s_jp^2 + 14p^3$

Table 4.1 shows that the computation of $[V_{j+1}, P_{j+1}]$ is the most expensive one in practice, while the computation of F_{j+1}, Λ_j and $\mathcal{H}_j$ is rather inexpensive since these matrices do not depend on n, requiring $\sum_{j=1}^{m}(4s_jp^2 + 18p^3) + 14p^3$ (or $\sum_{j=\frac{k}{p}+1}^{m}(4s_jp^2+18p^3)+14p^3$)

ETNA
Kent State University
http://etna.math.kent.edu

operations. Moreover, the cost for each cycle is monotonically decreasing as the method converges since the sequence k_j $(j \geq 1)$ is non-increasing. In addition, if p^3 is larger than the problem size n, we can naturally split the right-hand sides into small subblocks and solve the successive subsystems block by block.

Even though DBFGMRES-DR requires more computational cost due to the modified block Arnoldi vector deflation, this can be balanced with the faster convergence speed as shown in Section 5.

5. Numerical experiments. In this section, we present some numerical experiments to illustrate the potential of the new algorithm, with or without preconditioning, for the solution of the linear system (1.1).

In the following subsections, we mainly evaluate and compare the performance of the new method against the GMRES-DR approach [26] and other popular block iterative algorithms for solving linear systems with multiple right-hand sides. The first block solver is BGMRES-DR [25] with no deflation strategy. The second method is BGMRES with SVD based deflation (BGMRESD(m)) [7]. The third approach is modified block GMRES with deflation at each iteration (BGMRES-S(m)) [6]. Meanwhile, we also investigate the numerical behavior of Combined($m, 0$).

Here, m and k denote the number of iterations for each restart and the number of harmonic Ritz vectors, respectively. If the harmonic Ritz vectors are added to the subspace, the algorithm only runs $m - \lceil \frac{k}{p} \rceil$[1] steps of the deflated block flexible Arnoldi process to limit the dimension of the space. To limited storage or significant orthogonalization cost, the maximum of the dimension of the subspace is set to 100. In all of our runs we use $X_0 = 0_{n\times p}$ as our initial guess. The block right-hand side B has p columns generated randomly from a normal distribution. The deflation threshold is considered as $\varepsilon_d = 0.1$. We make comparisons in three aspects: the number of matrix-vector products, the runtime in seconds (referred to as CPU), and final true relative residual F-norm defined as $\frac{\|B-AX_k\|_F}{\|R_0\|_F}$ (referred to as res.norm). As stopping criterion we used either the condition $\frac{\|B-AX_k\|_F}{\|B\|_F} \leq 10^{-6}$ for all the solvers or that the matrix-vector products exceed the maximal matrix-vector product number (referred to as $MAXIT$). We consider $MAXIT = 2n$ except for the last two cases where we choose $MAXIT = 2000$. All the numerical experiments were performed in MATLAB 2011b on a PC-Pentium(R), CPU 2.00 GHz, 8.00 GB of RAM.

EXAMPLE 5.1. The purpose of this example is to illustrate the numerical behavior of BGMRES-DR, DBGMRES-DR, and the GMRES-DR approach when applied to the solution of the p linear systems in sequence. Following [25], the test matrix is a tridiagonal matrix with entries 0.1, 0.2, 0.3, 0.4, 0.5, 6, 7, ..., 1000 on the main diagonal, sub-diagonal entries all 1, and super-diagonal entries all 1. The right-hand sides are chosen to be random vectors with $p = 5$ or 10. For fair comparison, GMRES-DR will build a Krylov subspace of dimension $m \times p$. The numerical computations are carried out with $m = 10$, $k = 10$. The convergence curves are plotted in Figure 5.1.

From Table 5.1 and Figure 5.1, we can see that the block solvers require fewer matrix-vector products than GMRES-DR applied to the sequence of linear systems. In addition, when the number of right-hand sides increases, the average number of matrix-vector products for DBGMRES-DR is non-increasing. The above results show that block iterations have the potential to speed up the convergence, compared with the standard Krylov method.

EXAMPLE 5.2. In the second part, we test three matrices from Matrix Market [5]. These cases are summarized in Table 5.2, which shows the names of the matrices, the size, the

[1] $\lceil x \rceil$ rounds the elements of x to the nearest integers greater than or equal to x.

ETNA
Kent State University
http://etna.math.kent.edu

TABLE 5.1
Results of tridiagonal matrix.

	$p = 5, m = 10$		$p = 10, m = 10$	
	mvps	CPU	mvps	CPU
BGMRES-DR(m)	665	0.420	990	0.499
DBGMRES-DR(m)	517	0.385	777	0.467
GMRES-DR(mp)	1050	1.569	1900	3.398

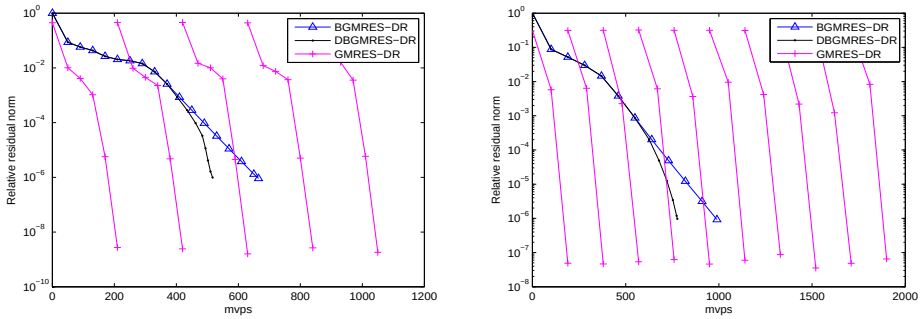

FIG. 5.1. *The convergence curves of different iterative methods on tridiagonal matrix. Left: p = 5. Right: p = 10.*

density of nonzero elements, and the type. We evaluate the performance of the five block solvers on the first two test matrices without preconditioning, while for the last one (`saylr4`) we use ILU preconditioning [20]. As block right-hand side we choose $B = \texttt{rand}(\texttt{n}, \texttt{p})$ (RHSs1) or $B =$ (random rank-six matrix) $+ 10^{-4} \times \texttt{rand}(\texttt{n}, \texttt{p})$ with $p = 10$ (RHSs2). The parameters are set to $m = 10$ and $k = 10$.

Figures 5.2, 5.3, and 5.4 show the convergence histories for all block solvers for RHSs1 (left) and RHSs2 (right). The corresponding matrix-vector products, CPU, and residual norms are reported in Table 5.3. Block methods allowing eigenvalue and vector deflation, i.e., DBGMRES-DR and Combined($m, 0$), are found to be efficient. They enjoy a significantly faster decrease in the number of matrix-vector products than other block solvers. Moreover, their convergence curves clearly highlight the interest of performing modified block Arnoldi vector deflation, compared with the BGMRES-DR algorithm. In addition, we also remark that the convergence curves of DBGMRES-DR and Combined($m, 0$) are different at the end for RHSs2 since DBGMRES-DR does not satisfy the stopping criterion due to $k_j = 0$. Even though Combined($m, 0$) requires slightly more runtime than DBGMRES-DR, it is able to satisfy the stopping criterion. Therefore, Combined($m, 0$) can be considered a slight modification of the DBGMRES-DR algorithm. On the whole, the two deflated solvers are more competitive for tough problems with small eigenvalues.

In addition, we also show the behaviour of k_j while performing modified block Arnoldi vector deflation. Figures 5.5, 5.6, and 5.7 depict the evolution of k_j, which is considered as the effective Krylov directions at iteration j, for two deflated block space solvers (DBGMRES-DR and BGMRES-S(m)). It is observed that both of them have a non-increasing behavior for k_j. However, the evolution of k_j for DBGMRES-DR enjoys a significant decrease due to eigenvalue deflation that exploits some spectral information, which could lead to less computing cost than for BGMRES-S(m).

ETNA
Kent State University
http://etna.math.kent.edu

TABLE 5.2
Summary of cases are used to study DBGMRES-DR.

	Problem	n	nnz	Density	Type	Application area
1	`sherman4`	1104	3784	0.0031	real unsymmetric	Oil reservoir modeling
2	tt ocean	4629	32,063	0.0015	real unsymmetric	Ocean circulation problem
3	`saylr4`	3564	22,316	0.0018	real unsymmetric	Oil reservoir modeling

TABLE 5.3
Performance comparisons for different block iterative methods.

problem	method	RHSs1			RHSs2		
		mvps	CPU	res.norm	mvps	CPU	res.norm
`sherman4`	BFGMRES-DR	820	3.090e-01	9.607e-07	520	2.083e-01	8.101e-07
	BFGMRES-S(m)	2288	1.328e+00	9.997e-07	1361	9.429e-01	3.208e-07
	BFGMRESD(m)	2170	6.790e-01	9.974e-07	1140	3.568e-01	5.339e-07
	DBFGMRES-DR	595	2.607e-01	9.877e-07	322	1.643e-01	1.827e-06
	Combined($m, 0$)	595	2.650e-01	9.877e-07	420	2.102e-01	1.447e-07
`ocean`	BFGMRES-DR	14,270	1.108e+01	9.878e-07	15,550	1.169e+01	9.885e-07
	BFGMRES-S(m)	18,516	1.742e+01	4.419e-01	18,517	2.218e+01	1.425e-02
	BFGMRESD(m)	18,517	1.342e+01	4.401e-01	18,517	1.510e+01	1.259e-02
	DBFGMRES-DR	7758	7.208e+00	9.990e-07	5287	6.117e+00	1.455e-06
	Combined($m, 0$)	7758	7.192e+00	9.990e-07	5487	6.224e+00	9.419e-07
`saylr4`	BFGMRES-DR	270	2.532e-01	8.443e-07	280	2.561e-01	8.211e-07
	BFGMRES-S(m)	744	1.311e+00	9.995e-07	1077	2.990e+00	9.378e-09
	BFGMRESD(m)	610	5.225e-01	7.794e-07	600	5.391e-01	9.074e-07
	DBFGMRES-DR	189	1.981e-01	8.792e-07	143	1.711e-01	1.398e-06
	Combined($m, 0$)	189	1.922e-01	8.792e-07	243	2.373e-01	2.587e-08

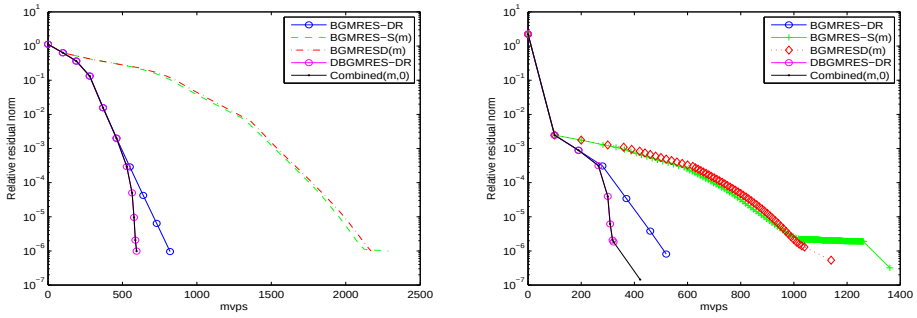

FIG. 5.2. *The convergence curves of different block iterative methods on matrix* `sherman4`. *Left: RHSs1. Right: RHSs2.*

EXAMPLE 5.3. This test case evaluates the performance of block solvers applied to two-dimensional Helmholtz problems with flexible preconditioning. We consider the problem on the domain $0 \le x \le 1, 0 \le y \le 1$ with the following Robin boundary conditions:

$$\begin{aligned} -\triangle u - \beta^2 u &= f, \\ u(0,y) &= g(y), \\ u(x,0) &= \rho(y), \\ u_x(1,y) &= pu(1,y) + a(y), \\ u_y(x,1) &= qu(x,1) + c(x), \end{aligned}$$

ETNA
Kent State University
http://etna.math.kent.edu

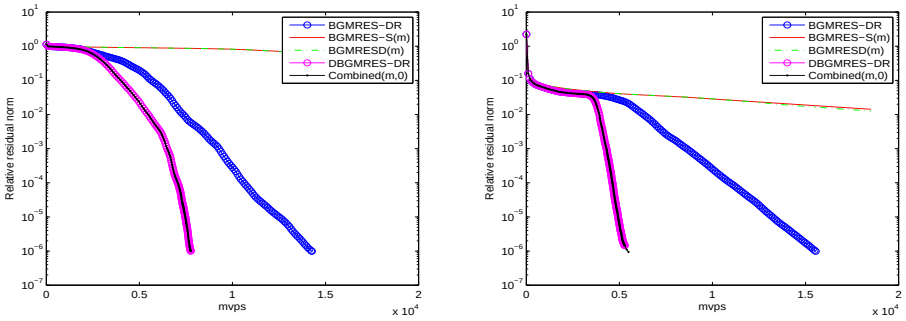

FIG. 5.3. *The convergence curves of different block iterative methods on matrix* `ocean`. *Left: RHSs1. Right: RHSs2.*

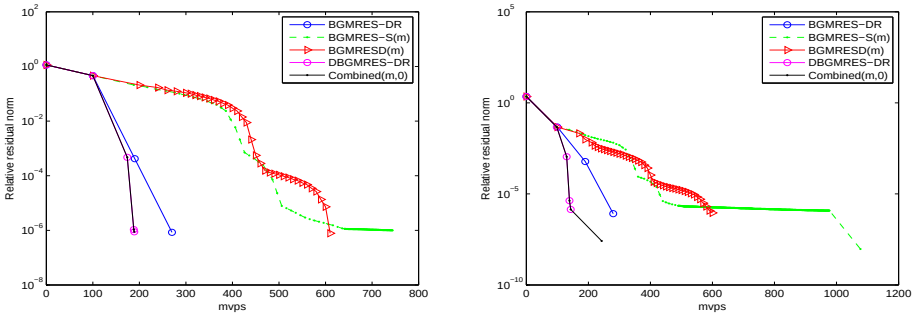

FIG. 5.4. *The convergence curves of different block iterative methods on matrix* `saylr4`. *Left: RHSs1. Right: RHSs2.*

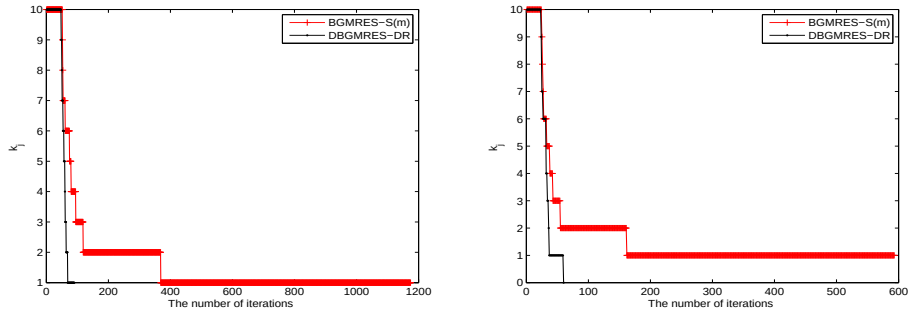

FIG. 5.5. *Evolution of* k_j *versus iterations different block iterative methods on matrix* `sherman4`. *Left: RHSs1. Right: RHSs2.*

where β, p, and q are constants. We specify the following conditions:

$$g(y) = 0,\ \rho(x) = 0,\ p = -3,\ q = 2,\ a(y) = 3\sin \pi y,\ \text{and}\ \ c(x) = -\pi \sin \frac{\pi x}{2}.$$

We use the classical five-points difference scheme to discretize the Helmholtz equation. We consider $2^7 \times 2^7$ and $2^8 \times 2^8$ grids for the discretization of the Helmholtz equation, which lead to nonsymmetric 16384×16384 and 65536×65536 matrices, respectively. Different

ETNA
Kent State University
http://etna.math.kent.edu

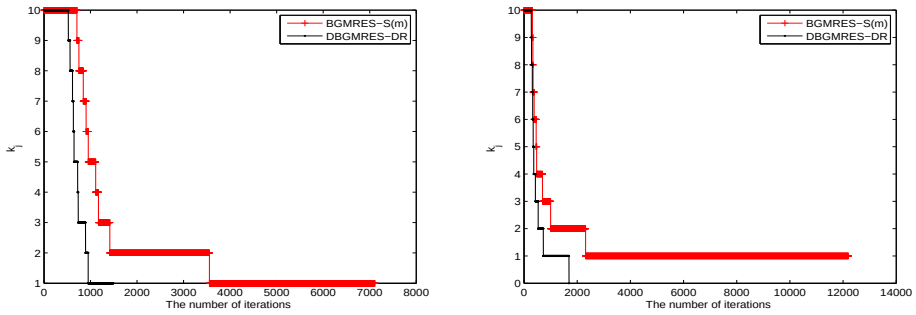

FIG. 5.6. *Evolution of k_j versus iterations different block iterative methods on matrix* `ocean`. *Left: RHSs1. Right: RHSs2.*

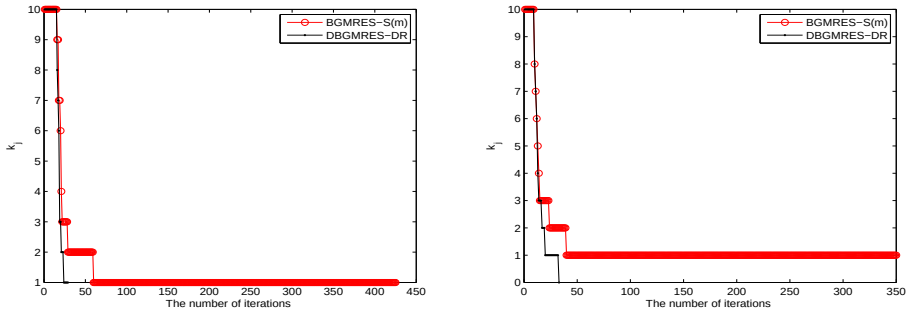

FIG. 5.7. *Evolution of k_j versus iterations different block iterative methods on matrix* `saylr4`. *Left: RHSs1. Right: RHSs2.*

grid resolutions are used to solve the problem with wavenumbers $\beta = \pi$. The flexible preconditioning is governed by a few steps of block full GMRES. Here, we consider 10 steps of block full GMRES. As block right-hand side we choose $B = rand(n, p)$ with $p = 10$ or $p = 20$ and consider $k = 10$, $m = 10$ or $m = 5$. The results are shown in the Table 5.4.

We can see from Table 5.4, Figures 5.8, and 5.9 that BGMRESD(m) and BGMRES-S(m) can not solve all the problems within the given steps with variable preconditioning. However, one should note that DBFGMRES-DR and Combined($m, 0$) enjoy a significant decrease in the number of matrix-vector products. Moreover, in terms of CPU time, the two deflated solvers are faster than the BFGMRES-DR method in most cases. In addition, since DBFGMRES-DR converges to the stopping criterion before k_j reaches the value 0, the convergence curves of DBFGMRES-DR are the same as those of Combined($m, 0$).

6. Conclusions and future work. We have derived a new DBFGMRES-DR method for linear systems with multiple right-hand sides. The new algorithm can address the possible linear dependence at each iteration during the block Arnoldi procedure and reduce computational expense. It is observed by experiments that DBFGMRES-DR significantly reduces the number of matrix-vector products. In addition, numerical examples also show that it enjoys faster convergence than some other block solvers on tough problems with small eigenvalues. In future work, we will combine the modified block Arnoldi vector deflation technique with the block GCROT(m, k) method [23]. The corresponding results [24] are also being considered.

ETNA
Kent State University
http://etna.math.kent.edu

TABLE 5.4
Results of Example Helmholtz.

p	m	method	n=16384			n=65536		
			mvps	CPU	res.norm	mvps	CPU	res.norm
10	10	BFGMRES-DR	330	5.556e+00	5.165e-07	710	4.795e+01	9.312e-07
		BFGMRES-S(m)	817	1.296e+01	9.693e-07	2001	1.182e+02	6.152e-02
		BFGMRESD(m)	1000	1.383e+01	8.704e-07	2000	1.053e+02	3.489e-02
		DBFGMRES-DR	200	3.369e+00	9.263-07	424	2.867e+01	9.736e-07
		Combined($m, 0$)	200	3.368e+00	9.263e-07	424	2.865e+01	9.736e-07
20	5	BFGMRES-DR	820	1.530e+01	7.096e-07	2020	1.579e+02	8.172e-01
		BFGMRES-S(m)	2002	4.261e+01	1.960e-03	2001	1.498e+02	7.363e-01
		BFGMRESD(m)	2000	3.378e+01	2.142e-03	2000	1.378e+02	7.546e-01
		DBFGMRES-DR	356	6.821e+00	8.580e-07	1076	9.133e+01	9.647e-07
		Combined($m, 0$)	356	6.822e+00	8.580e-07	1076	9.111e+01	9.647e-07

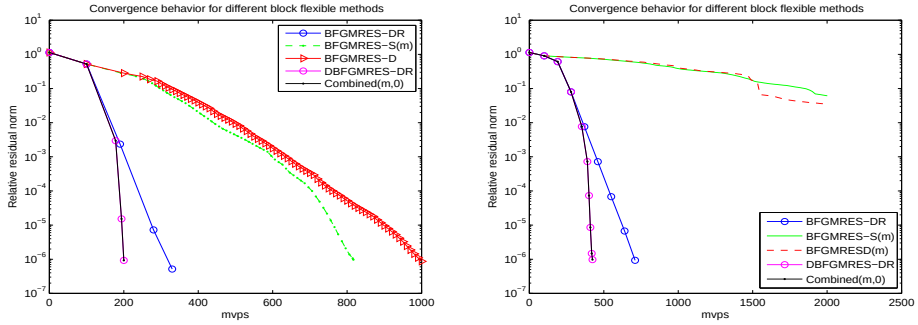

FIG. 5.8. *The convergence curves for different block flexible methods on Helmholtz problem with* $p = 10$. *Left: n = 16384. Right: n = 65536.*

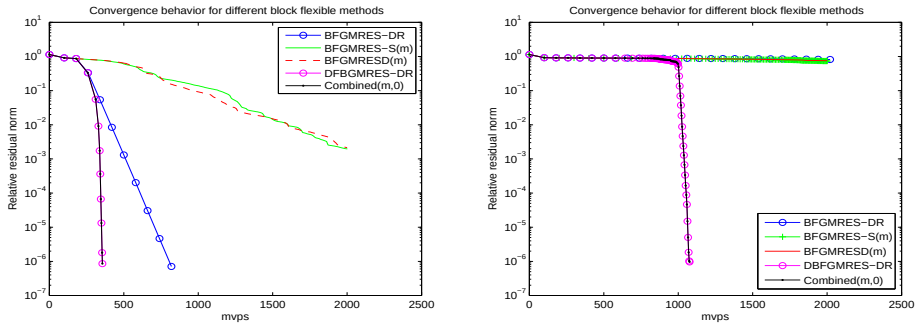

FIG. 5.9. *The convergence curves for different block flexible methods on Helmholtz problem with* $p = 20$. *Left: n = 16384. Right: n = 65536.*

Acknowledgments. The authors are grateful to the anonymous referees and the editors James Baglama and Carl Jagels for their valuable and constructive comments that greatly improved the quality of our paper.

ETNA
Kent State University
http://etna.math.kent.edu

REFERENCES

[1] J. BAGLAMA, *Augmented block Householder Arnoldi method*, Linear Algebra Appl., 429 (2008), pp. 2315–2334.
[2] J. C. R. BLOCH, T. BREU, A. FROMMER, S. HEYBROCK, K. SCHÄFER, AND T. WETTIG, *Short-recurrence Krylov subspace methods for the overlap Dirac operator at nonzero chemical potential*, Comput. Phys. Commun., 181 (2010), pp. 1378–1387.
[3] J. C. R. BLOCH AND S. HEYBROCK, *A nested Krylov subspace method to compute the sign function of large complex matrices*, Comput. Phys. Commun., 182 (2011), pp. 878–889.
[4] ———, *A nested Krylov subspace method for the overlap operator*, in Proceedings of the XXVII International Symposium on Lattice Field Theory, Bejing, C. Liu and Y. Zhu, eds., Proceedings of Science, Trieste, (LAT 2009) 025 (7 pages).
[5] R. BOISVERT, R. POZO, K. REMINGTON, B. MILLER, AND R. LIPMAN, *Matrix Market*, National Institute of Standards and Technology, 1996. Available at http://math.nist.gov/MatrixMarket/.
[6] H. CALANDRA, S. GRATTON, R. LAGO, X. VASSEUR, AND L. M. CARVALHO, *A modified block flexible GMRES method with deflation at each iteration for the solution of non-Hermitian linear systems with multiple right-hand sides*, SIAM J. Sci. Comput., 35 (2013), pp. S345–S367.
[7] H. CALANDRA, S. GRATTON, J. LANGOU, X. PINEL, AND X. VASSEUR, *Flexible vatiants of block restarted* GMRES *methods with application to geophysis*, SIAM J. Sci. Comput., 34 (2012), pp. A714–A736.
[8] R. W. CLOUGH AND J. PENZIEN, *Dynamics of Structures*, McGraw-Hill, New York, 1975.
[9] R. D. DA CUNHA AND D. BECKER, *Dynamic block* GMRES*: an iterative method for block linear systems*, Adv. Comput. Math., 27 (2007), pp. 423–448.
[10] L. DU, T. SOGABE, B. YU, Y. YAMAMOTO, AND S. L. ZHANG, *A block* IDR(s) *method of nonsymmetric linear systems with multiple right-hand side*, J. Comput. Appl. Math., 235 (2011), pp. 4095–4106.
[11] A. EL GUENNOUNI, K. JBILOU, AND H. SADOK, *A block version of* BICGSTAB *for linear systems with multiple right-hand sides*, Electron. Trans. Numer. Anal., 16 (2003), pp. 129–142. http://etna.mcs.kent.edu/vol.16.2003/pp129-142.dir
[12] ———, *The block Lanczos method for linear systems with multiple right-hand sides*, Appl. Numer. Math., 51 (2004), pp. 243–256.
[13] R. FREUND, *Krylov-subspace methods for reduced-order modeling in circuit simulation*, J. Comput. Appl. Math, 123 (2000), pp. 395–421.
[14] R. FREUND AND M. MALHOTRA, *A Block-*QMR *algorithm for non-hermitian linear systems with multiple right-hand sides*, Linear Algebra Appl., 254 (1997), pp. 119–157.
[15] L. GIRAUD, S. GRATTON, X. PINEL, AND X. VASSEUR, *Flexible GMRES with deflated restarting*, SIAM J. Sci. Comput., 32 (2010), pp. 1858–1878.
[16] M. H. GUTKNECHT, *Block Krylov space methods for linear systems with multiple right-hand sides: an introduction*, in Modern Mathematical Models, Methods and Algorithms for Real World Systems, A. H. Siddiqi, I. S. Duff, and O. Christensen, eds., Anshan, New Delhi, 2007, pp. 420–447.
[17] M. H. GUTKNECHT AND T. SCHMELZER, *The block grade of a block Krylov space*, Linear Algebra Appl., 430 (2009), pp. 174–185.
[18] S. KARIMI AND F. TOUTOUNIAN, *The block least squares method for solving nonsymmetric linear systems with multiple right-hand sides*, Appl. Math. Comput., 177 (2006), pp. 852–862.
[19] J. LANGOU, *Iterative Methods for Solving Linear Systems with Multiple Right-hand Sides*, Ph.D. Thesis, CERFACS, Toulouse, 2003.
[20] N. LI AND Y. SAAD, *Crout versions of ILU factorization with pivoting for sparse symmetric matrices*, Electron. Trans. Numer. Anal., 20 (2005), pp. 75–85. http://etna.mcs.kent.edu/vol.20.2005/pp75-85.dir
[21] N. LI, Y. SAAD, AND E. CHOW, *Crout versions of ILU for general sparse matrices*, SIAM J. Sci. Comput., 25 (2003), pp. 716–728.
[22] H.-L. LIU AND B.-J. ZHONG, *Simpler block* GMRES *for nonsymmetric systems with multiple right-hand sides*, Electron. Trans. Numer. Anal., 30 (2008), pp. 1–9. http://etna.mcs.kent.edu/vol.30.2008/pp1-9.dir
[23] J. MENG, P.-Y. ZHU, AND H.-B. LI, *A block GCROT(m,k) method for linear systems with multiple right-hand sides*, J. Comput. Appl. Math., 255 (2014), pp. 544–554.
[24] J. MENG, P.-Y. ZHU, H.-B. LI, AND Y.-F. JING, *A new deflated block GCROT(m,k) method for linear systems with multiple right-hand sides*, Preprint on arXiv, http://arxiv.org/abs/1405.3733.
[25] R. B. MORGAN, *Restarted block* GMRES *with deflation of eigenvalues*, Appl. Numer. Math., 54 (2005), pp. 222–236.
[26] ———, GMRES *with deflated restarting*, SIAM J. Sci. Comput., 24 (2002), pp. 20–37.
[27] A. NIKISHIN AND A. YEREMIN, *Variable block* CG *algorithms for solving large sparse symmetric positive definite linear systems on parallel computers I: general iterative scheme*, SIAM J. Matrix Anal. Appl.,

16 (1995), pp. 1135–1153.
[28] D. P. O' LEARY, , Linear Algebra Appl., 29 (1980), pp. 293–322.
[29] M. ROBBÉ AND M. SADKANE, *Exact and inexact breakdowns in the block* GMRES *method*, Linear Algebra Appl., 419 (2006), pp. 265–285.
[30] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, SIAM, Philadelphia, 2003.
[31] T. SAKURAI, H. TADANO, AND Y. KURAMASHI, *Application of block Krylov subspace algorithms to the Wilson-Dirac equation with multiple right-hand sides in lattice QCD*, Comput. Phys. Commun., 181 (2010), pp. 113–117.
[32] V. SIMONCINI, *A stabilized* QMR *version of block BICG*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 419–434.
[33] V. SIMONCINI AND E. GALLOPOULOS, *Convergence properties of block* GMRES *and matrix polynomials*, Linear Algebra Appl., 247 (1996), pp. 97–119.
[34] P. SOUDAIS, *Iterative solution methods of a 3-D scattering problem from arbitrary shaped multidielectric and multiconducting bodies*, IEEE Trans. on Antennas and Propagation, 42 (1994), pp. 954–959.
[35] B. VITAL, *Etude de Quelques Méthodes de Résolution de Problèmes Linéaires de Grande Taille sur Multiprocesseur*, Ph.D. Thesis, Universit de Rennes, 1990.