SIAM J. SCI. COMPUT.
Vol. 34, No. 2, pp. A714–A736

FLEXIBLE VARIANTS OF BLOCK RESTARTED GMRES METHODS WITH APPLICATION TO GEOPHYSICS*

HENRI CALANDRA†, SERGE GRATTON‡, JULIEN LANGOU§, XAVIER PINEL¶, AND XAVIER VASSEUR‖

Abstract. In a wide number of applications in computational science and engineering the solution of large linear systems of equations with several right-hand sides given at once is required. Direct methods based on Gaussian elimination are known to be especially appealing in that setting. Nevertheless, when the dimension of the problem is very large, preconditioned block Krylov space solvers are often considered as the method of choice. The purpose of this paper is thus to present iterative methods based on block restarted GMRES that allow variable preconditioning for the solution of linear systems with multiple right-hand sides. The use of flexible methods is especially of interest when approximate possibly iterative solvers are considered in the preconditioning phase. First we introduce a new variant of block flexible restarted GMRES that includes a strategy for detecting when a linear combination of the systems has approximately converged. This explicit block size reduction is often called deflation. We analyze the main properties of this flexible method based on deflation and notably prove that the Frobenius norm of the block residual is always nonincreasing. We also present a flexible variant based on both deflation and truncation to especially be used in case of limited memory. Finally we illustrate the numerical behavior of these flexible block methods for large industrial simulations arising in geophysics, where indefinite linear systems of size up to 1 billion unknowns with multiple right-hand sides have been successfully solved in a parallel distributed memory environment.

Key words. block Krylov space method, block size reduction, deflation, flexible preconditioning, multiple right-hand sides

AMS subject classifications. 65F10, 65N22, 15A06

DOI. 10.1137/10082364X

1. Introduction.

We consider block Krylov space methods for the solution of linear systems of equations with p right-hand sides given at once,

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

where $A \in \mathbb{C}^{n\times n}$ is supposed to be a nonsingular matrix of large dimension, $B \in \mathbb{C}^{n\times p}$ is full rank, and $X \in \mathbb{C}^{n\times p}$. We denote $X_0 \in \mathbb{C}^{n\times p}$ the initial block iterate and $R_0 = B - AX_0$ the initial block residual. In the case of no preconditioning, as stated in [20, 19], a block Krylov space method for solving the p systems is an iterative

*Submitted to the journal's Methods and Algorithms for Scientific Computing section February 8, 2011; accepted for publication (in revised form) December 12, 2011; published electronically March 22, 2012. This work was granted access to the HPC resources of IDRIS under allocation 2010065068 made by GENCI.

http://www.siam.org/journals/sisc/34-2/82364.html

†TOTAL, Centre Scientifique et Technique Jean Féger, avenue de Larribau F-64000 Pau, France (Henri.Calandra@total.com).

‡INPT-IRIT, University of Toulouse, and ENSEEIHT, 2 Rue Camichel, BP 7122, F-31071 Toulouse Cedex 7, France (serge.gratton@enseeiht.fr).

§University of Colorado, Denver, CO 80202 (julien.langou@ucdenver.edu). This author's work was supported by National Science Foundation grant NSF CCF 1054864.

¶CERFACS, 42 Avenue Gaspard Coriolis, F-31057 Toulouse Cedex 1, France (xavier.pinel@cerfacs.fr).

‖CERFACS and HiePACS project joint INRIA-CERFACS Laboratory, 42 Avenue Gaspard Coriolis, F-31057 Toulouse Cedex 1, France (vasseur@cerfacs.fr).

method that generates approximations $X_m \in \mathbb{C}^{n\times p}$ with $m \in \mathbb{N}$ such that

$$X_m - X_0 \in \mathcal{K}_m(A, R_0),$$

where the block Krylov space $K_m(A, R_0)$ is defined as

$$\mathcal{K}_m(A, R_0) = \left\{ \sum_{k=0}^{m-1} A^k R_0 \gamma_k \ \forall \ \gamma_k \in \mathbb{C}^{p\times p} \ \ \text{with } k \mid 0 \leq k \leq m-1 \right\} \subset \mathbb{C}^{n\times p}.$$

When the right-hand sides are available simultaneously, block Krylov methods are appealing at least for two reasons. First, they enable the systematic use of operations on a block of vectors instead of on a single vector. Depending on the structure of A, this may considerably reduce the number of memory accesses ([6], [27, section 3.7.2.3]). Second, by construction, the block Krylov space $K_m(A, R_0)$ contains all Krylov subspaces generated by each initial residual $K_m(A, R_0(:, i))$ for i such that $1 \leq i \leq p$ and all possible linear combinations of the vectors contained in these subspaces. Thus, contrary to the single right-hand-side case ($p = 1$), the solution of each linear system is sought in a potentially richer space leading hopefully to a reduction in terms of iteration count. We refer the reader to [20] for a recent overview on block Krylov subspace methods and note that most of the standard Krylov subspace methods have a block counterpart (see, e.g., block GMRES [52], block BiCGStab [18], and block QMR [16]).

When solving very large systems of linear equations resulting, e.g., from the discretization of partial differential equations in three dimensions, the use of preconditioning techniques based on a possibly nonlinear, iteration-dependent operator is often considered. This is the case when adaptive preconditioners using information obtained from previous iterations [4, 15] are used or when inexact solutions of the preconditioning system using, e.g., adaptive cycling strategy in multigrid [33] or approximate interior solvers in domain decomposition methods [48, section 4.3] are considered. Several authors have proposed Krylov subspace methods that allow variable preconditioning for the case of a linear system with a single right-hand side; see [3, 34, 39, 47, 50] among others.

To the best of our knowledge we note, however, that these developments have rarely addressed the case of linear systems with multiple right-hand sides; an exception is made [14], where a flexible variant of block restarted GMRES is described. To allow variable preconditioning also for the solution of multiple right-hand-side problems it seems natural to combine algorithms related to both flexible Krylov subspace methods and block Krylov space methods. In this paper we propose to derive flexible variants of block restarted GMRES and simultaneously pay special attention to the computational cost and memory requirements of the derived methods. Although potentially appealing as discussed before, block GMRES based algorithms are known to be computationally expensive due to the cost of orthogonalization [20]. Thus a primary concern when deriving those variants is to remove useless information for the convergence as soon as possible during the iterative procedure. This supposes to include strategies for detecting when a linear combination of the p systems has approximately converged. This explicit block size reduction is later called deflation, as discussed in [20]. The first strategy to remove useless information from a block Krylov subspace is called initial deflation. It consists of detecting linear dependency in the block right-hand-side B or in the initial block residual R_0 ([20, section 12] and [27, section 3.7.2]). This requires us to compute numerical ranks using rank-revealing QR-factorizations [10] or singular value decompositions (SVDs) [17] according to a certain deflation tolerance [23]. The linear dependency in the block residual can also be detected at each iteration of the block Krylov method. This has been notably implemented both in the hermitian [32, 38] and nonhermitian cases [1, 5, 12, 16, 30, 35] for block Lanczos methods. It has then been extended to GMRES, full orthogonalization method [37], and GCR [28], respectively, for block Arnoldi methods. A cheap variant in terms of memory for block GCR with deflation is also proposed in [44]; this method builds the block solution using only one column of its block residual (the one of maximal Euclidean norm). When a restarted method is used, deflation can also be performed at each initial computation of the block residual [20, section 14]. This strategy spares some rank-revealing QR-factorizations or SVDs and can sometimes be as efficient as methods based on deflation at each iteration.

The contribution of this paper will thus be twofold. First we will derive flexible variants of block GMRES that include deflation at the restart, and second, we will detail the convergence properties of those methods. In particular we will show that for some norms including the Frobenius norm, the norm of the block residual is nonincreasing along the iterations and will show the relevance of the approach on a challenging application. This paper is organized as follows. In section 2 we introduce the block flexible GMRES(m) method as a natural combination of block GMRES(m) and flexible GMRES(m). Then in section 3 we propose two variants of block flexible GMRES(m) based on deflation and analyze their main convergence properties. Furthermore, we demonstrate the effectiveness of the proposed algorithms for a challenging application in geophysics in section 4. Finally we draw some conclusions in section 5.

2. A flexible variant of block restarted GMRES.

2.1. Notation. Throughout this paper we denote $\|.\|_2$ the Euclidean norm, $\|.\|_F$ the Frobenius norm, $I_k \in \mathbb{C}^{k\times k}$ the identity matrix of dimension k, and $0_{i\times j} \in \mathbb{C}^{i\times j}$ the zero rectangular matrix with i rows and j columns. Superscript H denotes the transpose conjugate operation. Given a vector $d \in \mathbb{C}^k$ with components d_i, $D = \mathrm{diag}(d_1, \ldots, d_k)$ is the diagonal matrix $D \in \mathbb{C}^{k\times k}$ such that $D_{ii} = d_i$. If $C \in \mathbb{C}^{k\times l}$ we denote the singular values of C by $\sigma_1(C) \geq \cdots \geq \sigma_{\min(k,l)}(C) \geq 0$. Finally $e_m \in \mathbb{C}^n$ denotes the mth canonical vector of $\mathbb{C}^n$. Regarding the algorithmic part (Algorithms 1–4), we adopt notation similar to that of MATLAB in the presentation. For instance, $U(i,j)$ denotes the U_{ij} entry of matrix U, $U(1:m, 1:j)$ refers to the submatrix made of the first m rows and first j columns of U and $U(:,j)$ corresponds to its jth column.

2.2. Block flexible GMRES. In this section we present a block GMRES algorithm that allows variable preconditioning referred to as BFGMRES(m), where m denotes the maximum projection dimension (also called restart parameter). As briefly described in [14], it is derived as a natural combination of two existing algorithms: block GMRES (BGMRES) [52] and flexible GMRES(m) [39]. BGMRES was presented for the first time by Vital [52]. Since then numerous variants have been proposed; see [13, 25, 26, 29, 40, 41, 42] and also [21, 31] for versions exploiting spectral information to improve the convergence rate. Next we will introduce a flexible variant relying on a block version of the Arnoldi method. Throughout the paper, the orthogonalization scheme chosen is block modified Gram–Schmidt, although it is clear that one can change it at will with similar convergence effects as for the GMRES algorithm in floating point arithmetic.

2.2.1. Algorithm of block flexible GMRES. First we present in Algorithm 1 the block orthogonalization procedure used in the flexible setting, where M_j^{-1} denotes the preconditioning operator at step j ($1 \leq j \leq m$).

ALGORITHM 1. Flexible block Arnoldi with block modified Gram–Schmidt: Computation of $\mathcal{V}_{j+1}$, $\mathcal{Z}_j$ and $\bar{\mathcal{H}}_j$ for $1 \leq j \leq m$ with $V_1 \in \mathbb{C}^{n\times p}$ such that $V_1^H V_1 = I_p$.

1: **for** $j = 1, \ldots, m$ **do**
2: Choose preconditioning operator M_j^{-1}
3: $Z_j = M_j^{-1} V_j$
4: $S = A Z_j$
5: **for** $i = 1, \ldots, j$ **do**
6: $H_{i,j} = V_i^H S$
7: $S = S - V_i H_{i,j}$
8: **end for**
9: Compute the QR decomposition of S as $S = QR$ with $Q \in \mathbb{C}^{n\times p}$ and $R \in \mathbb{C}^{p\times p}$
10: Set $V_{j+1} = Q$, $H_{j+1,j} = R$ and $H_{i,j} = 0_{p\times p}$ for $i > j+1$
11: Define $\mathcal{Z}_j = [Z_1, \ldots, Z_j]$, $\mathcal{V}_{j+1} = [V_1, \ldots, V_{j+1}]$, $\bar{\mathcal{H}}_j = (H_{k,l})_{1\leq k\leq j+1, 1\leq l\leq j}$
12: **end for**

The flexible block Arnoldi method leads to the following relation (later called block flexible Arnoldi relation) for $1 \leq j \leq m$:

$$A\,[Z_1, \ldots, Z_j] = [V_1, V_2, \ldots, V_{j+1}] \begin{bmatrix} H_{1,1} & H_{1,2} & \ldots & H_{1,j} \\ H_{2,1} & H_{2,2} & \ldots & H_{2,j} \\ 0_{p\times p} & H_{3,2} & \ldots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0_{p\times p} & 0_{p\times p} & 0_{p\times p} & H_{j+1,j} \end{bmatrix}.$$

Equivalently with notation introduced in Algorithm 1, line 11, the orthogonalization procedure produces matrices $\mathcal{Z}_j \in \mathbb{C}^{n\times jp}$, $\mathcal{V}_{j+1} \in \mathbb{C}^{n\times (j+1)p}$, and $\bar{\mathcal{H}}_j \in \mathbb{C}^{(j+1)p\times jp}$, which satisfy

$$A\mathcal{Z}_j = \mathcal{V}_{j+1}\,\bar{\mathcal{H}}_j. \tag{2.1}$$

It should be noted that $\bar{\mathcal{H}}_j$ is no longer a Hessenberg matrix but a block Hessenberg matrix. More precisely, its block subdiagonal is made of upper triangular blocks of size $p \times p$. BFGMRES(m) (given in Algorithm 2) uses the flexible block version of the Arnoldi method with modified block Gram–Schmidt presented in Algorithm 1. In Algorithm 2 we denote by $\mathcal{B}_j \in \mathbb{C}^{(j+1)p\times p}$ the representation of the block residual $R_0 = B - AX_0$ in the $\mathcal{V}_{j+1}$ basis ($R_0 = \mathcal{V}_{j+1}\mathcal{B}_j$) and by $Y_j \in \mathbb{C}^{jp\times p}$ the solution of the following minimization problem:

$$\mathcal{P}_r : Y_j = \underset{Y\in\mathbb{C}^{jp\times p}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F. \tag{2.2}$$

The goal of this paper is to guarantee residual bounds at convergence for variants of block flexible GMRES methods. We start by extending a convergence property of block GMRES shown in [52] to the case of block flexible GMRES. We first show in Proposition 1 that the block flexible GMRES method minimizes the Euclidean norm of the residual of each linear system. This important property justifies the choice of the stopping criterion based on the Euclidean norm (Algorithm 2, line 10), as discussed later in section 2.2.2.

PROPOSITION 1. *In block flexible GMRES (BFGMRES(m), Algorithm* 2*) solving the reduced minimization problem* $\mathcal{P}_r$ (2.2) *amounts to minimizing the Frobenius norm*

ALGORITHM 2. BFGMRES(m).

```
Choose a convergence threshold tol, the size of the restart m and the maximum
   number of iterations itermax
Choose an initial guess X_0 ∈ C^{n×p}
Compute the initial block residual R_0 ∈ C^{n×p} as R_0 = B − AX_0
for iter = 1, ..., itermax do
   Compute the QR decomposition of R_0 as R_0 = QT with Q ∈ C^{n×p} and T ∈ C^{p×p}
   Set V_1 = Q and B_k = [ T ; 0_{kp×p} ], 1 ≤ k ≤ m.
   for j = 1, ..., m do
      Completion of V_{j+1}, Z_j and H̄_j: Apply Algorithm 1 from line 3 to 11
      with flexible preconditioning (Z_j = M_j^{-1} V_j, 1 ≤ j ≤ m) to obtain V_{j+1} ∈
      C^{n×(j+1)p}, Z_j ∈ C^{n×jp} and the matrix H̄_j ∈ C^{(j+1)p×jp} such that:

            A Z_j = V_{j+1} H̄_j   with   V_{j+1}^H V_{j+1} = I_{(j+1)p}.

      Solve the minimization problem Y_j = argmin_{Y ∈ C^{jp×p}} ||B_j − H̄_j Y||_F
      if ||B_j(:,l) − H̄_j Y_j(:,l)||_2 / ||B(:,l)||_2 ≤ tol, ∀ l | 1 ≤ l ≤ p then
         Compute X_j = X_0 + Z_j Y_j; stop
      end if
   end for
   Compute X_m = X_0 + Z_m Y_m and R_m = B − AX_m
   Set R_0 = R_m and X_0 = X_m
end for
```

of the block true residual $||B - AX||_F$ *over the space* $X_0 + \operatorname{range}(\mathcal{Z}_j Y)$ *at iteration* j $(1 \le j \le m)$ *of a given cycle, i.e.,*

$$\begin{aligned} \underset{Y \in \mathbb{C}^{jp \times p}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F &= \underset{Y \in \mathbb{C}^{jp \times p}}{\operatorname{argmin}} ||B - A(X_0 + \mathcal{Z}_j Y)||_F, \\ \min_{Y \in \mathbb{C}^{jp \times p}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F &= \min_{Y \in \mathbb{C}^{jp \times p}} ||B - A(X_0 + \mathcal{Z}_j Y)||_F. \end{aligned} \tag{2.3}$$

Furthermore, solving the reduced minimization problem $\mathcal{P}_r$ (2.2) *is also equivalent to minimizing the Euclidean norm of each linear system over the space* $X_0(:,l) + \operatorname{range}(\mathcal{Z}_j)$ $(1 \le l \le p)$ *at iteration* j $(1 \le j \le m)$.

Proof. Using successively the unitary invariance of the Frobenius norm, $R_0 = \mathcal{V}_{j+1}\mathcal{B}_j$, and the block flexible Arnoldi relation (2.1), we can formulate the minimization problem $\mathcal{P}_r$ (2.2) as

$$\underset{Y \in \mathbb{C}^{jp \times p}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F = \underset{Y \in \mathbb{C}^{jp \times p}}{\operatorname{argmin}} ||B - A(X_0 + \mathcal{Z}_j Y)||_F,$$

which ends the first part of the proof. This establishes the following relation between the block true residual $R_j = B - AX_j$ and $\mathcal{B}_j - \bar{\mathcal{H}}_j Y_j$, the Arnoldi residual (also called the block quasi-residual in [20])

$$||B - AX_j||_F = ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y_j||_F,$$

which will be useful later when defining appropriate stopping criterion for the block flexible GMRES(m) method. Finally, using essentially the same arguments now in the Euclidean norm we can rewrite the $||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F^2$ as

$$\begin{aligned} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F^2 &= \sum_{l=1}^{p} ||\mathcal{B}_j(:,l) - \bar{\mathcal{H}}_j Y(:,l)||_2^2, \\ &= \sum_{l=1}^{p} ||R_0(:,l) - A\mathcal{Z}_j Y(:,l)||_2^2, \\ &= \sum_{l=1}^{p} ||B(:,l) - A(X_0(:,l) + \mathcal{Z}_j Y(:,l))||_2^2. \end{aligned}$$

Therefore the initial minimization problem $\mathcal{P}_r$ (2.2) posed in the Frobenius norm is separable. Minimizing $||B - AX_j||_F$ can then be performed by solving p independent least-squares problems, one for each linear system. ∎

Remark 1. In Algorithm 2, line 14, we propose to compute the block true residual explicitly rather than using $R_m = \mathcal{V}_{m+1}(\mathcal{B}_m - \bar{\mathcal{H}}_m Y_m)$ as obtained in Proposition 1. Indeed if A is a sparse matrix with $nnz(A)$ nonzero entries, it is usually cheaper to compute explicitly $R_m = B - AX_m$ ($2nnz(A)p + np$ operations) than to evaluate $\mathcal{V}_{m+1}(\mathcal{B}_m - \bar{\mathcal{H}}_m Y_m)$ ($2n(m+1)p^2$ operations), where terms only proportional to the size of the problem n have been considered in the last estimate.

2.2.2. Detection of convergence.

The detection of convergence related to the p linear systems is performed at each iteration during a given cycle of BFGMRES(m), as shown in Algorithm 2.[1] We briefly motivate the choice of the stopping criterion.

COROLLARY 1. *In block flexible GMRES (BFGMRES(m), Algorithm 2) detecting the convergence on the block true residual is equivalent to detecting the convergence on the block quasi-residual in exact arithmetic:*

$$\frac{||B(:,l) - AX_j(:,l)||_2}{||B(:,l)||_2} \leq tol \;\forall\, l \mid 1 \leq l \leq p \Leftrightarrow \frac{||\mathcal{B}_j(:,l) - \bar{\mathcal{H}}_j Y_j(:,l)||_2}{||B(:,l)||_2} \leq tol \;\forall\, l \mid 1 \leq l \leq p.$$

Proof. This is a direct consequence of Proposition 1. ∎

Proposition 1 and Corollary 1 have guided the choice of the stopping criterion proposed in Algorithm 2, line 10. We note that the Frobenius norm could also be used to check the convergence since

$$\max_{1 \leq l \leq p} ||R_j(:,l)||_2^2 \leq ||R_j||_F^2 \leq p \max_{1 \leq l \leq p} ||R_j(:,l)||_2^2,$$

and hence the inequality $||R_j||_F \leq tol$ guarantees convergence on all systems. However, the convergence of each individual linear system (or a combination of them) may occur sooner. Variants that aim at reducing the global computational cost will be detailed in section 3.

3. Flexible variants of block GMRES(m) based on deflation and truncation.

3.1. Block flexible GMRES with deflation.

When solving multiple right-hand-side problems, linear dependence of the residuals of the p linear systems may occur. Such dependence has to be taken into account to reduce the block size along the iterations and yield effective block Krylov space methods as stressed in [20, section 8].

[1] A cycle of BFGMRES(m) corresponds to the computational operations sketched from line 5 to line 15 in Algorithm 2.

Determining a linearly independent subset of the columns of the block true residual is thus required. The dimension of this subset will correspond to the effective number of linear systems to be considered; this explicit reduction is called deflation. In practice approximate deflation depending on a deflation tolerance is usually preferred since exact deflation is rare. The main ideas related to deflation in block Krylov methods are presented in ([20, section 14], [30]) and are generalizations of initial deflation techniques proposed in [27].

Block GMRES with deflation has been detailed in [20, sections 12–14], where this explicit reduction is implemented at each restart with help of rank-revealing factorizations. Robbé and Sadkane [37] have recently proposed to introduce deflation during each iteration of block GMRES(m). The main idea consists of detecting linear dependency in the block residual at each iteration. Of course, this implies an additional computational cost but it has been found that this strategy can really improve convergence at the same memory cost as in BGMRES(m). However, since small restart parameters are sometimes considered in practice for memory issues, we propose a simpler algorithm implementing deflation solely at the restart of BFGMRES(m).

3.1.1. Algorithm of block flexible GMRES with deflation. The block flexible restarted GMRES with deflation, later named BFGMRESD(m), is presented in Algorithm 3. Hereafter we outline how approximate deflation has been introduced and thus describe a given cycle of the method (lines 6 to 21 in Algorithm 3). The deflation procedure detects approximate linear dependency in the block true residual. For that purpose, given a QR-factorization of the scaled block true residual $R_0 D^{-1} = QT$, where $D \in \mathbb{C}^{p\times p}$ is defined as $D = \mathrm{diag}(d_1, \dots, d_p)$ with $d_l = ||B(:,l)||_2$ $(1 \le l \le p)$, an SVD of the upper triangular matrix $T \in \mathbb{C}^{p\times p}$ is performed, which leads to the following relation:

$$T = U\Sigma W^H, \tag{3.1}$$

where $U \in \mathbb{C}^{p\times p}$, $W \in \mathbb{C}^{p\times p}$ are unitary and $\Sigma \in \mathbb{C}^{p\times p}$ is diagonal. The use of diagonal scaling with matrix D enables convergence detection on the true block residual scaled by the norm of the right-hand sides, as explained later in section 3.1.2. We note that the related cost of the SVD of T ($O(p^3)$ operations) is negligible in practice since p, the number of right-hand sides, is supposed to be considerably less than n, the dimension of the problem. As explained in section 3.1, deflation consists of selecting relevant information from the decomposition (3.1). Indeed, we determine a subset of the singular values of T according to the following condition:

$$\sigma_l(T) > \varepsilon_d\ tol \quad \forall\ l \,\text{such that}\, 1 \le l \le p_d, \tag{3.2}$$

where ε_d is a real positive parameter less than one. This leads to following decomposition of the diagonal matrix Σ

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

with $\Sigma_+ \in \mathbb{C}^{p_d\times p_d}$ defined as $\Sigma_+ = \Sigma(1:p_d, 1:p_d)$ and $\Sigma_- \in \mathbb{C}^{(p-p_d)\times(p-p_d)}$ as $\Sigma_- = \Sigma(p_d+1:p, p_d+1:p)$. Due to the approximate deflation condition (3.2), we note that

$$||\Sigma_+||_2 > \varepsilon_d\ tol \quad \text{and} \quad ||\Sigma_-||_2 \le \varepsilon_d\ tol.$$

Furthermore the scaled block true residual $R_0\, D^{-1}$ can be written as

$$R_0\, D^{-1} = Q\ [U_+\ U_-] \begin{bmatrix} \Sigma_+ & 0_{p_d \times (p-p_d)} \\ 0_{(p-p_d)\times p_d} & \Sigma_- \end{bmatrix} [W_+\ W_-]^H,$$

$$R_0\, D^{-1} = Q\ U_+ \Sigma_+ W_+^H + Q\ U_- \Sigma_- W_-^H, \tag{3.3}$$

where we set $U_+ \in \mathbb{C}^{p \times p_d}$ as $U_+ = U(:, 1 : p_d)$ and $W_+ \in \mathbb{C}^{p\times p_d}$ as $W_+ = W(:, 1 : p_d)$. Similarly, we define $U_- \in \mathbb{C}^{p\times(p-p_d)}$ as $U_- = U(:, p_d + 1 : p)$ and $W_- \in \mathbb{C}^{p\times(p-p_d)}$ as $W_- = W(:, p_d + 1 : p)$. U_+, W_+ and Σ_+ denote the quantities effectively considered in a given cycle of Algorithm 3, while U_-, W_-, and Σ_- are put aside due to deflation. Indeed, since $W = [W_+, W_-]$ is unitary, it is straightforward to see from (3.3) that

$$||R_0\, D^{-1} W_-||_2 \le \varepsilon_d\ tol.$$

If deflation is active in this cycle ($p_d < p$), *only* p_d linear systems will be considered, which may yield a significant reduction in terms of operations. Given $V_1 = QU_+$ the flexible block Arnoldi method with block modified Gram–Schmidt (Algorithm 1) is applied to obtain $\mathcal{Z}_j \in \mathbb{C}^{n\times jp_d}$, $\mathcal{V}_{j+1} \in \mathbb{C}^{n\times(j+1)p_d}$, and $\bar{\mathcal{H}}_j \in \mathbb{C}^{(j+1)p_d\times jp_d}$, which satisfy

$$A\mathcal{Z}_j = \mathcal{V}_{j+1}\, \bar{\mathcal{H}}_j. \tag{3.4}$$

We denote by $\mathcal{B}_j \in \mathbb{C}^{(j+1)p_d\times p_d}$ the representation of the scaled block residual in the $\mathcal{V}_{j+1}$ basis ($\mathcal{V}_{j+1}\mathcal{B}_j = QU_+$) and by $Y_j \in \mathbb{C}^{jp_d\times p_d}$ the solution of the reduced minimization problem:

$$\mathcal{P}_r^d : \quad Y_j = \underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F. \tag{3.5}$$

PROPOSITION 2. *In block flexible GMRES with deflation (BFGMRESD(m), Algorithm* 3*) solving the reduced minimization problem* $\mathcal{P}_r^d$ (3.5) *amounts to minimizing the Frobenius norm of the block true residual* $||B - AX||_F$ *over the space* $X_0 + \text{range}(\mathcal{Z}_j Y \Sigma_+ W_+^H\, D)$ *at iteration* j $(1 \le j \le m)$ *of a given cycle, i.e.,*

$$\underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F = \underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||B - A(X_0 + \mathcal{Z}_j Y \Sigma_+ W_+^H D)||_F, \tag{3.6}$$

$$= \underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||R_0 D^{-1} - A\mathcal{Z}_j Y \Sigma_+ W_+^H||_F. \tag{3.7}$$

Proof. Σ_+ being a diagonal matrix, using elementary properties of the Frobenius norm leads to

$$\underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F^2 = \underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||(\mathcal{B}_j - \bar{\mathcal{H}}_j Y)\Sigma_+||_F^2.$$

Since the Frobenius norm is unitarily invariant the last equality can be recast into

$$\underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F^2 = \underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||\mathcal{V}_{j+1}(\mathcal{B}_j - \bar{\mathcal{H}}_j Y)\Sigma_+ W_+^H||_F^2.$$

Using the block flexible Arnoldi relation (3.4) leads to

$$\underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F^2 = \underset{Y\in\mathbb{C}^{jp_d\times p_d}}{\operatorname{argmin}} ||\mathcal{V}_{j+1}\mathcal{B}_j \Sigma_+ W_+^H - A\mathcal{Z}_j Y \Sigma_+ W_+^H||_F^2.$$

Since $\mathcal{V}_{j+1}\mathcal{B}_j = V_1 = QU_+$, the quantity $\mathcal{V}_{j+1}\mathcal{B}_j\Sigma_+W_+^H$ satisfies the relation

$$\mathcal{V}_{j+1}\mathcal{B}_j\Sigma_+W_+^H = QU_+\Sigma_+W_+^H,$$

which finally leads to

$$\operatorname*{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} ||\mathcal{B}_j - \bar{\mathcal{H}}_jY||_F^2 = \operatorname*{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} ||QU_+\Sigma_+W_+^H - A\mathcal{Z}_jY\Sigma_+W_+^H||_F^2.$$

Adding a term independent of Y on the right-hand side of the previous equation obviously allows us to write

$$\begin{aligned}&\operatorname*{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} ||\mathcal{B}_j - \bar{\mathcal{H}}_jY||_F^2\\ &\quad= \operatorname*{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} (||QU_+\Sigma_+W_+^H - A\mathcal{Z}_jY\Sigma_+W_+^H||_F^2 + ||QU_-\Sigma_-W_-^H||_F^2).\end{aligned}$$

Since $W = [W_+, W_-]$ is unitary, we obtain

$$\operatorname*{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} ||\mathcal{B}_j - \bar{\mathcal{H}}_jY||_F^2 = \operatorname*{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} ||QU_+\Sigma_+W_+^H + QU_-\Sigma_-W_-^H - A\mathcal{Z}_jY\Sigma_+W_+^H||_F^2,$$

which becomes, due to relation (3.3),

$$\operatorname*{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} ||\mathcal{B}_j - \bar{\mathcal{H}}_jY||_F^2 = \operatorname*{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} ||R_0D^{-1} - A\mathcal{Z}_jY\Sigma_+W_+^H||_F^2. \qquad \square$$

Due to Proposition 2, the approximate solution that is based on a generalized minimum Frobenius norm approach is obtained as

$$X_j = X_0 + \mathcal{Z}_jY_j\Sigma_+W_+^H\,D$$

at the end of the cycle ($j = m$) or before if the stopping criterion is satisfied at iteration j. Proposition 2 also implies the nonincreasing behavior of the block residual in the Frobenius norm in BFGMRESD(m).

3.1.2. Detection of convergence. Similarly as in BFGMRES(m) (Algorithm 2), the detection of convergence related to the p linear systems (Algorithm 3, line 15) is performed in BFGMRESD(m) at each iteration during a given cycle. For that purpose we consider a stopping criterion based on the Euclidean norm of the components of the block quasi-residual $\mathcal{R}_j \in \mathbb{C}^{(j+1)p_d\times p}$ defined as

$$\mathcal{R}_j = (\mathcal{B}_j - \bar{\mathcal{H}}_jY_j)\Sigma_+W_+^H$$

in the deflated case. We discuss next how to choose both stopping and deflation thresholds in practice (ε_q and ε_d, respectively) when deflation has occurred ($p_d < p$) at a given restart. The next proposition (Proposition 3) gives an explicit upper bound on the Euclidean norm of each individual residual.

PROPOSITION 3. *In block flexible GMRES with deflation (BFGMRESD(m), Algorithm* 3*) the block true residual R_j satisfies the following inequality at iteration j ($1 \le j \le m$) of a given cycle,*

$$\frac{||R_j(:,l)||_2}{||B(:,l)||_2} \le ||\mathcal{R}_j(:,l)||_2 + \sigma_{p_d+1}(T)\ \forall\ l \mid 1 \le l \le p. \tag{3.8}$$

ALGORITHM 3. BFGMRESD(m).

1: Choose a convergence threshold tol, a deflation threshold ε_d, a quality of convergence threshold ε_q, the size of the restart m and the maximum number of iterations $itermax$
2: Choose an initial guess $X_0 \in \mathbb{C}^{n\times p}$
3: Define the diagonal matrix $D \in \mathbb{C}^{p\times p}$ as $D = \text{diag}(d_1, \dots, d_p)$ with $d_l = ||B(:, l)||_2$ for l such that $1 \leq l \leq p$
4: Compute the initial block residual $R_0 = B - AX_0$
5: **for** $iter = 1, \dots, itermax$ **do**
6: Compute the QR decomposition of $R_0 D^{-1}$ as $R_0 D^{-1} = QT$ with $Q \in \mathbb{C}^{n\times p}$ and $T \in \mathbb{C}^{p\times p}$
7: Compute the SVD of T as $T = U\ \Sigma\ W^H$
8: Select p_d singular values of T such that $\sigma_l(T) > \varepsilon_d\ tol$ for all l such that $1 \leq l \leq p_d$
9: Define $V_1 \in \mathbb{C}^{n\times p_d}$ as $V_1 = QU(:, 1 : p_d)$
10: Let $\mathcal{B}_k = \begin{bmatrix} I_{p_d} \\ 0_{kp_d\times p_d} \end{bmatrix}$, $1 \leq k \leq m$
11: **for** $j = 1, \dots, m$ **do**
12: *Completion of* $\mathcal{V}_{j+1}$, $\mathcal{Z}_j$ *and* $\bar{\mathcal{H}}_j$ *(see Algorithm* 1): Apply Algorithm 1 from line 3 to 11 with flexible preconditioning ($Z_j = M_j^{-1} V_j$, $1 \leq j \leq m$) to obtain $\mathcal{V}_{j+1} \in \mathbb{C}^{n\times(j+1)p_d}$, $\mathcal{Z}_j \in \mathbb{C}^{n\times jp_d}$ and the matrix $\bar{\mathcal{H}}_j \in \mathbb{C}^{(j+1)p_d\times jp_d}$ such that:

$$A\mathcal{Z}_j = \mathcal{V}_{j+1}\bar{\mathcal{H}}_j \quad \text{with} \quad \mathcal{V}_{j+1}^H \mathcal{V}_{j+1} = I_{(j+1)p_d}.$$

13: Solve the minimization problem $Y_j = \text{argmin}_{Y\in\mathbb{C}^{jp_d\times p_d}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F$;
14: Compute $\mathcal{R}_j = (\mathcal{B}_j - \bar{\mathcal{H}}_j Y_j)\Sigma(1 : p_d, 1 : p_d)W(1 : p, 1 : p_d)^H$
15: **if** $||\mathcal{R}_j(:, l)||_2 \leq \varepsilon_q\ tol,\ \forall\ l \mid 1 \leq l \leq p$ **then**
16: Compute $X_j = X_0 + \mathcal{Z}_j Y_j \Sigma(1 : p_d, 1 : p_d)W(1 : p, 1 : p_d)^H D$; stop;
17: **end if**
18: **end for**
19: $X_m = X_0 + \mathcal{Z}_m Y_m \Sigma(1 : p_d, 1 : p_d)W(1 : p, 1 : p_d)^H\, D$
20: $R_m = B - AX_m$
21: Set $R_0 = R_m$ and $X_0 = X_m$
22: **end for**

Furthermore, if convergence occurs at iteration j (Algorithm 3*, line* 15*)), R_j satisfies the inequality*

$$\frac{||R_j(:, l)||_2}{||B(:, l)||_2} \leq tol\,\varepsilon_q + \sigma_{p_d+1}(T)\ \forall\ l \mid 1 \leq l \leq p.$$

Proof. Using developments introduced in Proposition 2, the block true residual at iteration j can be written as

$$\begin{aligned} R_j &= B - A(X_0 + \mathcal{Z}_j Y_j \Sigma_+ W_+^H\, D), \\ R_j &= R_0 - \mathcal{V}_{j+1}\bar{\mathcal{H}}_j Y_j \Sigma_+ W_+^H\, D, \\ R_j &= \left[\mathcal{V}_{j+1}(\mathcal{B}_j - \bar{\mathcal{H}}_j Y_j)\Sigma_+ W_+^H + QU_-\Sigma_- W_-^H\right]\, D, \\ R_j D^{-1} &= \left[\mathcal{V}_{j+1}\mathcal{R}_j + QU_-\Sigma_- W_-^H\right]. \end{aligned}$$

Thus for each linear system ($1 \leq l \leq p$) we obtain the inequality

$$\frac{||R_j(:,l)||_2}{||B(:,l)||_2} \leq ||\mathcal{V}_{j+1}\mathcal{R}_j(:,l)||_2 + ||QU_-\Sigma_-W_-(l,:)^H||_2,$$
$$\frac{||R_j(:,l)||_2}{||B(:,l)||_2} \leq ||\mathcal{R}_j(:,l)||_2 + \sigma_{p_d+1}(T),$$

which ends the first part of the proof. The second inequality is straightforward. Indeed, if the stopping criterion is satisfied ($||\mathcal{R}_j(:,l)||_2 \leq \varepsilon_q tol$ for the p linear systems (Algorithm 3, line 15)), the inequality (3.8) becomes

$$\frac{||R_j(:,l)||_2}{||B(:,l)||_2} \leq \varepsilon_q tol + \sigma_{p_d+1}(T). \qquad \square$$

When the convergence is declared, a simple way to make sure that the scaled block residual norm is below tol consists of choosing a fixed quality of convergence threshold $\varepsilon_q \in (0,1)$ such that $\varepsilon_q + \varepsilon_d = 1$. Indeed, if such a relation is satisfied, we obtain at convergence

$$\varepsilon_q tol + \sigma_{p_d+1}(T) \leq (\varepsilon_q + \varepsilon_d)\, tol$$

and consequently

$$\frac{||R_j(:,l)||_2}{||B(:,l)||_2} \leq tol. \tag{3.9}$$

We note that different convergence and deflation thresholds can also be chosen at each cycle. A possible strategy could aim at obtaining a less severe convergence threshold on the block quasi-residual $\mathcal{R}_j$ leading to a reduction in terms of inner iterations. For instance, considering Proposition 3, if at each cycle ε_q is chosen such that

$$\varepsilon_q = 1 - \frac{\sigma_{p_d+1}(T)}{tol}, \tag{3.10}$$

and if the stopping criterion on the block quasi-residual $\mathcal{R}_j$ is satisfied, relation (3.9) holds since $\sigma_{p_d+1}(T) \leq \varepsilon_d\ tol$.

3.2. Block flexible GMRES with deflation and truncation. At the same memory cost as in BFGMRES(m) we have been able to introduce a variant (BFGMRESD(m)) which exploits the idea of deflation. This explicit block size reduction should hopefully lead to a reduction in terms of computational operations when treating multiple right-hand-side problems. We present next a variant of BFGMRESD(m) that exhibits a lower memory cost. This latter feature is particularly appealing when considering linear systems of large size with multiple right-hand sides, as discussed later in section 4.

3.2.1. Algorithm of block flexible GMRES with deflation and truncation. The block flexible restarted GMRES with deflation and truncation is given in Algorithm 4. Hereafter we only outline how truncation has been introduced since the method is similar to the block flexible restarted GMRES with deflation (Algorithm 3) in many aspects. Truncation here consists of fixing once and for all the maximum number of columns of the block vectors to be considered in the method. We denote by p_f this value (with $p_f < p$). Given the SVD of $T = U\Sigma W^H$ with $\Sigma \in \mathbb{C}^{p\times p}$ determined as in section 3.1.1, combining deflation and truncation leads to decompose $\Sigma_+ \in \mathbb{C}^{p_f\times p_f}$ into

$$\Sigma_+ = \begin{bmatrix} \Sigma_+^b & 0_{p_b\times(p_f-p_b)} \\ 0_{(p_f-p_b)\times p_b} & \Sigma_+^- \end{bmatrix},$$

where p_b is defined as $\min(p_d, p_f)$, $\Sigma_+^b \in \mathbb{C}^{p_b\times p_b}$ as $\Sigma_+^b = \Sigma(1:p_b, 1:p_b)$ and $\Sigma_+^- \in \mathbb{C}^{(p_f-p_b)\times(p_f-p_b)}$ as $\Sigma_+^- = \Sigma(p_b+1:p_f, p_b+1:p_f)$. In a given cycle, deflation is active only when $p_d \le p_f$. In such a case, when $p_f > p_b$, the method relies on the following decomposition of $R_0\, D^{-1}$:

$$R_0\, D^{-1} = Q\ [U_+^b\ U_-] \begin{bmatrix} \Sigma_+^b & 0_{p_b\times(p_f-p_b)} & 0_{p_b\times(p-p_f)} \\ 0_{(p_f-p_b)\times p_b} & \Sigma_+^- & 0_{(p_f-p_b)\times(p-p_f)} \\ 0_{(p-p_f)\times p_b} & 0_{(p-p_f)\times(p_f-p_b)} & \Sigma_- \end{bmatrix} [W_+^b\ W_-]^H,$$

where we set $U_+^b \in \mathbb{C}^{p\times p_b}$ as $U_+ = U(:, 1:p_b)$, $W_+^b \in \mathbb{C}^{p\times p_b}$ as $W_+^b = W(:, 1:p_b)$, $U_- \in \mathbb{C}^{p\times(p-p_b)}$ as $U_- = U(:, p_b+1:p)$, $\Sigma_-^- \in \mathbb{C}^{(p-p_f)\times(p-p_f)}$ as $\Sigma_-^- = \Sigma(p_f+1:p, p_f+1:p)$, and $W_- \in \mathbb{C}^{p\times(p-p_b)}$ as $W_- = W(:, p_b+1:p)$. Quantities with subscript and superscript $_-$ are discarded due to truncation and deflation, respectively, while only p_b linear systems are considered in the cycle. When $p_b = p_f$, we have the following decomposition:

$$\Sigma = \begin{bmatrix} \Sigma_+^b & 0_{p_f\times(p-p_f)} \\ 0_{(p-p_f)\times p_f} & \Sigma_- \end{bmatrix}.$$

Similarly as in section 3.1.1, we denote by $\mathcal{B}_j \in \mathbb{C}^{(j+1)p_b\times p_b}$ the representation of the scaled block residual in the $\mathcal{V}_{j+1}$ basis after truncation and deflation ($\mathcal{V}_{j+1}\mathcal{B}_j = QU_+^b$) and by $Y_j \in \mathbb{C}^{jp_b\times p_b}$ the solution of the reduced minimization problem:

$$\mathcal{P}_r^t: \quad Y_j = \underset{Y\in\mathbb{C}^{jp_b\times p_b}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F. \tag{3.11}$$

PROPOSITION 4. *In block flexible GMRES with deflation and truncation (BFGMREST(m, p_f), Algorithm* 4*) solving the reduced minimization problem* $\mathcal{P}_r^t$ (3.11) *amounts to minimizing the Frobenius norm of the block true residual* $||B - AX||_F$ *over the space* $X_0 + \operatorname{range}(\mathcal{Z}_j Y \Sigma_+^b {W_+^b}^H D)$ *at iteration* j $(1 \le j \le m)$ *of a given cycle, i.e.,*

$$\underset{Y\in\mathbb{C}^{jp_b\times p_b}}{\operatorname{argmin}} ||\mathcal{B}_j - \bar{\mathcal{H}}_j Y||_F = \underset{Y\in\mathbb{C}^{jp_b\times p_b}}{\operatorname{argmin}} ||B - A(X_0 + \mathcal{Z}_j Y \Sigma_+^b {W_+^b}^H D)||_F. \tag{3.12}$$

Proof. The proof follows the same developments as in Proposition 2. ∎

This strategy offers the flexibility to considerably reduce both the memory requirements and the computational cost of a given cycle since only p_b linear systems will be considered. However, due to truncation this method may fail to converge or require more outer iterations to converge than BFGMRESD(m) because combinations of residuals that have not converged are also discarded from the block Krylov space. Nevertheless, this has to be balanced with its reduced memory requirements and computational cost as shown in section 4.

3.2.2. Detection of convergence. A straightforward adaptation of Proposition 3 to the case of block flexible GMRES with deflation and truncation leads to the upper bound on each normalized linear system residual

$$\frac{||R_j(:,l)||_2}{||B(:,l)||_2} \leq ||\mathcal{R}_j(:,l)||_2 + \sigma_{p_b+1}(T), \ \forall\, l \mid 1 \leq l \leq p \tag{3.13}$$

with the block quasi-residual $\mathcal{R}_j \in \mathbb{C}^{(j+1)p_b \times p}$ defined as $\mathcal{R}_j = (\mathcal{B}_j - \bar{\mathcal{H}}_j Y_j)\Sigma_+^b {W_+^b}^H$ in the deflated and truncated cases. As a simple stopping condition we check that $\sigma_{p_b+1}(T)$ is less than the convergence threshold *tol* (Algorithm 4, line 16) and if successful we verify the condition $||\mathcal{R}_j(:,l)||_2 < tol - \sigma_{p_b+1}(T)$ (Algorithm 4, line 17). This ensures convergence thanks to inequality (3.13).

3.2.3. Computational cost and memory requirements. We summarize in Table 3.1 the main computational costs occurring during a given cycle of BFGMREST(m, p_f) (Algorithm 4). We have only included the costs proportional to the size of the original problem n which is supposed to be much greater than m and p in practical applications. This also excludes the costs related to both matrix-vector products and preconditioning operations. The total cost is quadratic in p_f (the maximal column size of the block vectors) and linear in n (the dimension of the problem).

Table 3.2 summarizes the maximal memory requirements (proportional to n) for the three algorithms presented so far. Each method requires the storage of R_m, X_0, X_m, $\mathcal{V}_{m+1}$, and $\mathcal{Z}_m$, respectively. We note that only BFGMREST(m, p_f) leads to a reduction in terms of memory requirements.

3.3. Convergence analysis in another unitarily invariant norm. In the previous sections we have mainly considered both the Frobenius norm and the Euclidean norm of each column of the block residual to describe convergence results related to block Krylov subspace methods. It is possible, however, to prove a slightly more general convergence result that holds in any unitarily invariant norm. First, we recall that the proposed methods amount to minimizing the Frobenius norm of the block true residual $||B - AX||_F$ over the space $X_0 + \text{range}(\mathcal{Z}_j Y)$ at iteration j ($1 \leq j \leq m$) of a given cycle, i.e., the general form of the minimization problem can

Table 3.1

Maximal computational cost of a cycle of BFGMREST(m, p_f) with $p_b = \min(p_f, p_d)$. This excludes the cost of matrix-vector operations and preconditioning operations.

Step	Computational cost of a cycle
Computation of $R_0 D^{-1}$	n
QR-factorization of $R_0 D^{-1}$	$2np_b^2 + 5np_b$
Computation of V_1	$2npp_b$
Block Arnoldi procedure*	$2nm(m+2)p_b^2 + (5mn + \frac{m(m+1)}{2}n)p_b$
Computation of X_m	$np + nmpp_b$
Total	$np_b^2[2(m+1)^2] + np_b[p(m+2) + (m+1)\frac{(10+m)}{2}]$ $+n(p+1)$

*Algorithm 4, line 13: the blocked Arnoldi method based on modified Gram–Schmidt (Algorithm 1) requires $\sum_{j=1}^{m}\sum_{i=1}^{j}(4np_b^2 + np_b)$ operations plus $\sum_{j=1}^{m}(2np_b^2 + 5np_b)$ operations for the QR decomposition of W.

TABLE 3.2
Maximal memory requirements in BFGMRES(m), BFGMRESD(m), and BFGMREST(m, p_f).

Method	BFGMRES(m)	BFGMRESD(m)	BFGMREST(m, p_f)
Storage	$n(2m+1)p+3np$	$n(2m+1)p+3np$	$n(2m+1)p_f+3np$

ALGORITHM 4. BFGMREST(m,p_f).

1: Choose a convergence threshold tol, a deflation threshold ε_d, a fixed block size $p_f < p$, the size of the restart m and the maximum number of iterations $itermax$
2: Choose an initial guess $X_0 \in \mathbb{C}^{n\times p}$
3: Define the diagonal matrix $D \in \mathbb{C}^{p\times p}$ as $D = \mathrm{diag}(d_1, \ldots, d_p)$ with $d_l = \|B(:,l)\|_2$ for l such that $1 \le l \le p$
4: Compute the initial block residual $R_0 = B - AX_0$
5: **for** $iter = 1, \ldots, itermax$ **do**
6: Compute the QR decomposition of R_0D^{-1} as $R_0\,D^{-1} = QT$ with $Q \in \mathbb{C}^{n\times p}$ and $T \in \mathbb{C}^{p\times p}$
7: Compute the SVD of T as $T = U\ \Sigma\ W^H$
8: Select p_d singular values of T such that $\sigma_l(T) > \varepsilon_d\ tol$ for all l such that $1 \le l \le p_d$
9: Set $p_b = \min(p_d, p_f)$
10: Define $V_1 \in \mathbb{C}^{n\times p_b}$ as $V_1 = QU(:, 1:p_b)$
11: Let $\mathcal{B}_k = \begin{bmatrix} I_{p_b} \\ 0_{kp_b\times p_b} \end{bmatrix}$, $1 \le k \le m$
12: **for** $j = 1, \ldots, m$ **do**
13: *Completion of* $\mathcal{V}_{j+1}$, $\mathcal{Z}_j$ *and* $\bar{\mathcal{H}}_j$ *(see Algorithm* 1*):* Apply Algorithm 1 from line 3 to 11 with flexible preconditioning ($Z_j = M_j^{-1}V_j$, $1 \le j \le m$) to obtain $\mathcal{V}_{j+1} \in \mathbb{C}^{n\times(j+1)p_b}$, $\mathcal{Z}_j \in \mathbb{C}^{n\times jp_b}$ and the matrix $\bar{\mathcal{H}}_j \in \mathbb{C}^{(j+1)p_b\times jp_b}$ such that:
$$A\mathcal{Z}_j = \mathcal{V}_{j+1}\bar{\mathcal{H}}_j \quad \text{with} \quad \mathcal{V}_{j+1}^H\mathcal{V}_{j+1} = I_{(j+1)p_b}.$$
14: Solve the minimization problem $Y_j = \mathrm{argmin}_{Y\in\mathbb{C}^{jp_b\times p_b}}\|\mathcal{B}_j - \bar{\mathcal{H}}_jY\|_F$
15: Compute $\mathcal{R}_j = (\mathcal{B}_j - \bar{\mathcal{H}}_jY_j)\Sigma(1:p_b, 1:p_b)W(1:p, 1:p_b)^H$
16: **if** $\sigma_{p_b+1}(T) < tol$ **then**
17: **if** $\|\mathcal{R}_j(:,l)\|_2 \le tol - \sigma_{p_b+1}(T)\,\forall l \le p$ **then**
18: Compute $X_j = X_0 + \mathcal{Z}_jY_j\Sigma(1:p_b, 1:p_b)W(1:p, 1:p_b)^H\,D$; stop;
19: **else**
20: $X_{next} = X_j, R_{next} = R_j$
21: Go to 28
22: **end if**
23: **end if**
24: **end for**
25: $X_m = X_0 + \mathcal{Z}_mY_m\Sigma(1:p_b, 1:p_b)W(1:p, 1:p_b)^H\,D$
26: $R_m = B - AX_m$
27: $X_{next} = X_m, R_{next} = R_m$
28: Set $R_0 = R_{next}$ and $X_0 = X_{next}$
29: **end for**

be written as

$$\mathcal{P} : \underset{Y \in \mathbb{C}^{js \times s}}{\operatorname{argmin}} ||B - A(X_0 + \mathcal{Z}_j Y)||_F = \underset{S \in \operatorname{range}(A\mathcal{Z}_j)}{\operatorname{argmin}} ||R_0 - S||_F$$

with $s = p$ for BFGMRES(m), $s = p_d$ for BFGMRESD(m), and $s = p_b$ for BFGMREST(m, p_f); see Propositions 1, 2, and 4 respectively. The lth column of the current residual $R(:, l)$ at iteration j is then obtained as the orthogonal projection of $R_0(:, l)$ onto $(\operatorname{range}(A\mathcal{Z}_j))^{\perp}$. Thus $R = PR_0$, where P is the orthogonal projector onto $(\operatorname{range}(A\mathcal{Z}_j))^{\perp}$. From [24, Theorem 3.3.16], we obtain that the singular values of the block residual are monotonically decreasing, i.e.,

$$\forall\ i \mid 1 \le i \le p \quad \sigma_i(PR_0) \le \sigma_i(R_0). \tag{3.14}$$

This important property guarantees that deflating with respect to singular values is appropriate. Furthermore, from [22, Relation (B.7)][2] we conclude that for any given unitarily invariant norm $\|.\|$ on $\mathbb{C}^{n \times s}$, the following property is satisfied:

$$\|R\| \le \|R_0\|.$$

The convergence of the block residual norm is then monotone in any unitarily invariant norm.

4. Numerical experiments.

In this section we investigate the numerical behavior of block flexible GMRES(m) methods on a challenging realistic application in geophysics. We introduce the background of this study and then detail the performance of the various methods that have been introduced in sections 2 and 3. To give a broad picture of their performance we will also include comments related to both computational time and memory requirements.

4.1. Acoustic full waveform inversion.

We focus on a specific application in geophysics related to the simulation of wave propagation phenomena in the earth [51]. Given a three-dimensional physical domain Ω_p, the propagation of a wavefield in a heterogeneous medium can be modeled by the Helmholtz equation written in the frequency domain:

$$-\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} - \frac{\partial^2 u}{\partial z^2} - \frac{(2\pi f)^2}{c^2(x,y,z)} u = \delta(\mathbf{x} - \mathbf{x_s}), \quad \mathbf{x} = (x, y, z) \in \Omega_p. \tag{4.1}$$

The unknown u represents the pressure field in the frequency domain, c the acoustic wave velocity in ms^{-1}, which varies with position, and f the frequency in Hertz. The source term $\delta(\mathbf{x} - \mathbf{x_s})$ represents a harmonic point source located at (x_s, y_s, z_s). The wavelength λ is defined as $\lambda = \frac{c(x,y,z)}{f}$. A popular approach—the perfectly matched layer formulation (PML) [8, 9]—has been used in order to obtain a satisfactory near boundary solution, without many artificial reflections. This artificial boundary layer is used to absorb outgoing waves at any incidence angle, as shown in [8]. The acoustic full waveform inversion requires the solution of three-dimensional Helmholtz problems at various locations of the Dirac sources and thus leads to multiple right-hand-side problems [45, 46].

We consider a standard second-order accurate seven-point finite-difference discretization of the Helmholtz equation (4.1) on a uniform equidistant Cartesian grid

[2] For any unitarily invariant norm $\|.\|$, $\|ABC\| \le \|A\|_2 \|B\| \|C\|_2$, $\quad A \in \mathbb{C}^{r \times m}, B \in \mathbb{C}^{m \times n}, C \in \mathbb{C}^{n \times s}$.

of size $n_x \times n_y \times n_z$. We denote later by h the corresponding mesh grid size, Ω_h the discrete computational domain, and n_{PML} the number of points in the PML layer. A fixed value for $n_{PML} = 16$ is considered hereafter. After discretization, the acoustic full wave inversion leads to the following linear system with p multiple right-hand sides:

$$A\,X = B,$$

where $A \in \mathbb{C}^{n\times n}$ is a sparse complex matrix which is nonhermitian and nonsymmetric due to the PML formulation and $B \in \mathbb{C}^{n\times p}$. Since a stability condition has to be satisfied to correctly represent the wave propagation phenomena [11], we consider numerical discretization schemes with 12 points per wavelength. Consequently we fix the mesh grid size h in m and deduce the frequency f in Hz as

$$f = \frac{\min_{(x,y,z)\in\Omega_h} c(x,y,z)}{12\,h}.$$

This relation imposes solving very large systems of equations at the (high) frequencies of interest to geophysicists, a task that may be too computationally and memory expensive for sparse direct methods. Due also to their indefiniteness these systems are known to be challenging for iterative methods. An approximate geometric two-level preconditioner for flexible Krylov subspace methods has recently been designed in [36] to address the solution to such problems. Standard geometric coarsening in all directions is used to deduce the coarse grid Ω_H from Ω_h, whereas trilinear interpolation (I_H^h) and its adjoint (I_h^H) are considered as intergrid transfer operators [49]. The coarse grid operator A_H is obtained by standard discretization of (4.1) on the coarse grid Ω_H. The generic two-grid cycle is shown in Algorithm 5. In our numerical experiments we have fixed once for all the different parameters involved. The two-grid method uses 1 cycle of symmetric Gauss–Seidel preconditioned GMRES(2) as a relaxation method ($\mu_1 = \mu_2 = 1$, $m_s = 2$) and considers 10 cycles of symmetric Gauss–Seidel preconditioned GMRES(10) to approximately solve the coarse grid problem ($\mu_c = 10$, $m_c = 10$) and zero initial guesses ($z_h^0 = 0$, $z_H^0 = 0$, respectively).

ALGORITHM 5. Approximate geometric two-grid cycle applied to $A_h z_h = v_h$ with initial approximation z_h^0. $z_h = \mathcal{M}(v_h)$.

1: Polynomial presmoothing: Apply μ_1 cycle(s) of GMRES(m_s) to $A_h\ z_h = v_h$ with initial approximation z_h^0 and symmetric Gauss–Seidel as a right preconditioner to obtain the approximation $z_h^{\mu_1}$.
2: Restrict the fine level residual: $v_H = I_h^H(v_h - A_h\ z_h^{\mu_1})$.
3: Solve approximately the coarse problem $A_H z_H = v_H$: Apply μ_c cycles of GMRES(m_c) to $A_H\ z_H = v_H$ with initial approximation z_H^0 and symmetric Gauss–Seidel as a right preconditioner to obtain the approximation z_H.
4: Correct the fine grid approximation: $\widetilde{z_h} = z_h^{\mu_1} + I_H^h\ z_H$.
5: Polynomial postsmoothing: Apply μ_2 cycle(s) of GMRES(m_s) to $A_h\ z_h = v_h$ with initial approximation $\widetilde{z_h}$ and symmetric Gauss–Seidel as a right preconditioner to obtain the approximation $z_h := z_h^{\mu_2}$.

The approximation at the end of the cycle z_h can be represented as $z_h = \mathcal{M}(v_h)$, where $\mathcal{M}$ is a nonlinear function. Thus a cycle of this two-grid method sketched in Algorithm 5 leads to a variable preconditioning operator. Flexible outer Krylov methods must then be considered. In [36, section 4.3] it has been shown experimentally that using this two-grid cycle as a preconditioner of FGMRES(m) with a moderate value of the restart parameter ($m = 5$) has led to an efficient numerical method on realistic problems in geophysics. See [43] for a theoretical analysis of inner-outer methods when the outer and inner methods are the same (FGMRES and GMRES in our setting). It is notably proved that by using preconditioners which are Krylov methods the global iteration is maintained within a larger Krylov subspace. In this section we consider this two-grid preconditioner in the multiple right-hand-side case and next investigate the performance of the block flexible Krylov methods presented in sections 2 and 3 on a challenging real-life application.

4.2. The SEG/EAGE overthrust model.

4.2.1. Settings. The SEG/EAGE overthrust model [2] is a synthetic velocity field often used as a benchmark problem in seismic applications. The reference domain where the acoustic velocity $c(x, y, z)$ is recorded is a box of size $20 \times 20 \times 4.65$ km^3. The minimum value of the velocity is 2179 m.s^{-1} and its maximum value is 6000 m.s^{-1}. The p sources are located in the plane $z/h = n_{PML} + 1$ on the line $y/h = n_y/2$ each 50 meters along the x axis starting from $x/h = n_{PML} + 1$:

$$B(:, l) = \delta_{\left(n_{PML}+1+(l-1)\frac{50}{h}, \frac{n_y}{2}, n_{PML}+1\right)} = e_{i_l} \ \forall \ l = 1, \dots, p. \tag{4.2}$$

The block right-hand-side $B \in \mathbb{C}^{n \times p}$ is thus extremely sparse; it contains only one nonzero element per column. We compare various preconditioned iterative methods based on flexible GMRES(m) for the solution of (4.1) with a zero initial guess and a moderate value of the restart parameter ($m = 5$). The iterative procedures are stopped when the Euclidean norm of each column of the block residual normalized by the Euclidean norm of the corresponding right-hand side satisfies the following relation:

$$\frac{||B(:, l) - AX(:, l)||_2}{||B(:, l)||_2} \leq tol \ \forall \ l = 1, \dots, p. \tag{4.3}$$

The tolerance is set to $tol = 10^{-5}$ in the numerical experiments. Since the initial block residual corresponds to the full rank matrix B, we note that no initial deflation occurs in the block variants investigated here. The numerical results shown in section 4.2.2 were obtained on Babel, a Blue Gene/P computer located at IDRIS (PowerPC 450 850 Mhz with 512 MB memory on each core) using a Fortran 90 implementation with MPI in single precision arithmetic. This code was compiled by the IBM compiler suite with standard compiling options and linked with the vendor BLAS and LAPACK subroutines.

4.2.2. Numerical results. Five different strategies are considered in this comparison. The first, FGMRES(5) sequence, consists of solving the linear systems in sequence, always choosing a zero initial guess X_0. The second method, FGMRES(5) simultaneous, applies FGMRES(5) to each linear system simultaneously with a convergence detected in a blockwise manner. This method is designed to take advantage of possible computational speed-up obtained by gathering operations (matrix-vector products, dot products, and communications between processors) and minimizing memory transfers. The third, fourth, and fifth strategies are related to block flexible methods: BFGMRES(5) (Algorithm 2), BFGMRESD(5) (Algorithm 3), and BFGMREST(5, p_f) (Algorithm 4), respectively. In this last strategy we consider two values for the block sizes p_f ($p_f = p/2$ and $p_f = p/4$). The deflation threshold ε_d has been set to 1 and the quality of convergence threshold ε_q has been chosen according to relation (3.10).

This numerical study addresses a simple practical question: given a fixed number of cores of a parallel distributed memory computer and a certain number of right-hand sides, which numerical method among the five strategies leads to the smallest computational times on this application?

In Tables 4.1, 4.2 and 4.3, we compare these various strategies on three different problems corresponding to increasing frequencies of interest to geophysicists. In each experiment we consider three cases for the multiple right-hand-side problems ($p = 4$, 8, 16, respectively). Since doubling the number of right-hand sides nearly doubles the memory requirement of the block methods, we also multiply the number of cores by a factor of two with respect to the number of right-hand sides. This aims at imposing the same memory constraint on each core for all numerical experiments. For each strategy we collect the number of applications of the two-grid preconditioner on a single vector (Pr) required to satisfy the stopping criterion (relation (4.3)), the elapsed time in seconds (T), and the requested memory in gigabytes (M).

Table 4.1

Perturbed two-grid preconditioned flexible methods for the solution of the Helmholtz equation for the SEG/EAGE overthrust model. Case of $f = 3.64$ Hz ($h = 50$ m) with $p = 4$, $p = 8$, and $p = 16$ right-hand sides at once. The parameter T denotes the total computational time in seconds, Pr the number of preconditioner applications on a single vector, and M the requested memory in GB.

Overthrust, Grid: $433 \times 433 \times 126$, $h = 50$ m, $f = 3.64$ Hz									
	$p = 4$, #Cores = 32			$p = 8$, #Cores = 64			$p = 16$, #Cores = 128		
Method	Pr	T	M	Pr	T	M	Pr	T	M
FGMRES($5p$) sequence	56	624	9.9	112	629	17.8	224	665	33.9
FGMRES(5) sequence	56	618	4.0	112	623	4.1	224	657	4.2
FGMRES(5) simultaneous	56	613	15.8	112	615	31.7	224	639	64
BFGMRES(5)	56	622	15.8	112	631	31.7	224	668	64
BFGMRESD(5)	43	**489**	15.8	70	**401**	31.7	120	**371**	64
BFGMREST(5,p/2)	48	542	8.8	80	447	17.5	140	410	35.4
BFGMREST(5,p/4)	51	576	5.3	92	524	10.4	169	489	21

Table 4.2

Perturbed two-grid preconditioned flexible methods for the solution of the Helmholtz equation for the SEG/EAGE overthrust model. Case of $f = 7.27$ Hz ($h = 25$ m) with $p = 4$, $p = 8$, and $p = 16$ right-hand sides at once. The parameter T denotes the total computational time in seconds, Pr the number of preconditioner applications on a single vector, and M the requested memory in GB.

Overthrust, Grid: $836 \times 836 \times 224$, $h = 25$ m, $f = 7.27$ Hz									
	$p = 4$, #Cores = 256			$p = 8$, #Cores = 512			$p = 16$, #Cores = 1024		
Method	Pr	T	M	Pr	T	M	Pr	T	M
FGMRES($5p$) sequence	120	1202	66.1	240	1243	118.6	480	1318	226.6
FGMRES(5) sequence	120	1198	27.4	240	1216	27.7	483	1302	28.3
FGMRES(5) simultaneous	120	1195	105	240	1209	212	496	1303	430
BFGMRES(5)	120	1214	105	248	1277	212	496	1359	430
BFGMRESD(5)	85	**892**	105	135	**734**	212	235	**695**	430
BFGMREST(5,p/2)	95	955	58.6	160	805	117	260	707	237
BFGMREST(5,p/4)	96	951	35.2	180	904	69.6	320	831	140.3

TABLE 4.3

Perturbed two-grid preconditioned flexible methods for the solution of the Helmholtz equation for the SEG/EAGE overthrust model. Case of $f = 14.53$ Hz ($h = 12.5$ m) with $p = 4$, $p = 8$, and $p = 16$ right-hand sides at once. The parameter T denotes the total computational time in seconds, Pr the number of preconditioner applications on a single vector, and M the requested memory in GB.

Overthrust, Grid: $1637 \times 1637 \times 413$, $h = 12.5$ m, $f = 14.53$ Hz									
	$p = 4$, #Cores = 2048			$p = 8$, #Cores = 4096			$p = 16$, #Cores = 8192		
Method	Pr	T	M	Pr	T	M	Pr	T	M
FGMRES($5p$) sequence	360	3315	457	863	3903	844	1753	4309	1613
FGMRES(5) sequence	362	3293	190	858	3769	197	1776	4125	201
FGMRES(5) simultaneous	364	3279	731	864	3762	1507	1808	4079	3060
BFGMRES(5)	360	3291	731	856	3823	1507	1776	4192	3060
BFGMRESD(5)	270	**2563**	731	515	**2418**	1507	910	**2242**	3060
BFGMREST(5,p/2)	291	2638	406	601	2607	832	1040	2380	1686
BFGMREST(5,p/4)	305	2718	244	655	2842	495	1280	2850	999

In the upper part of Tables 4.1, 4.2, and 4.3 we have included the results of FGMRES sequence with a value of the restart parameter equal to $5p$ (i.e., 20, 40, and 80 for $p = 4, 8, 16$, respectively). Whatever the frequency and the number of right-hand sides, the number of preconditioner applications Pr obtained with such a strategy is found to be very similar to the corresponding results related to FGMRES(5) sequence. This shows that there is no advantage in using a larger restart parameter to decrease the number of preconditioner applications on such a problem. Even worse, due to the larger value of the restart parameter, memory requirements are found to be prohibitive in FGMRES($5p$) sequence. Thus we choose to consider a moderate fixed value of the restart parameter $m = 5$ in the following. When handling multiple right-hand-side problems with block methods, this offers the additional advantage of limiting both the computational cost of a given block method (see Table 3.1) and memory requirements (see Table 3.2).

We now compare the different methods when the size of the restart parameter is fixed to 5 whatever the strategy. The results related to FGMRES(5) sequence lead to one important comment. For $f = 3.64$ Hz and $f = 7.27$ Hz, the number of preconditioner applications is multiplied exactly by a factor of two when the number of right-hand sides p is multiplied by the same factor (first lines of Tables 4.1 and 4.2, respectively). This property, however, is not satisfied in the case of the largest frequency $f = 14.53$ Hz (Table 4.3). This behavior can be explained as follows. An analysis of the perturbed two-grid preconditioned FGMRES(5) on three-dimensional heterogeneous Helmholtz problems in a *single* right-hand-side situation has shown that the numerical method satisfies a strong scalability property up to a given number of cores [36]. We believe that this loss of scalability is due to the preconditioner used both in the smoother and in the approximate solution of the coarse problem. This preconditioner (symmetric Gauss–Seidel) is based on a subdomain decoupling and becomes inherently less efficient when the number of cores is increasing [7]. As a consequence, for a given problem, the computational times related to FGMRES(5) sequence are expected to increase when the number of cores becomes large. An increase by a factor of 1.25 is noticed in this case (3293 s for $p = 4$ versus 4125 s for $p = 16$, first line of Table 4.3). Thus we obtain in this study a scalability with respect to the number of right-hand sides (since we multiply here by the same factor of 2 both the number of cores and the number of right-hand sides) only up to 1024 cores with FGMRES(5) sequence.

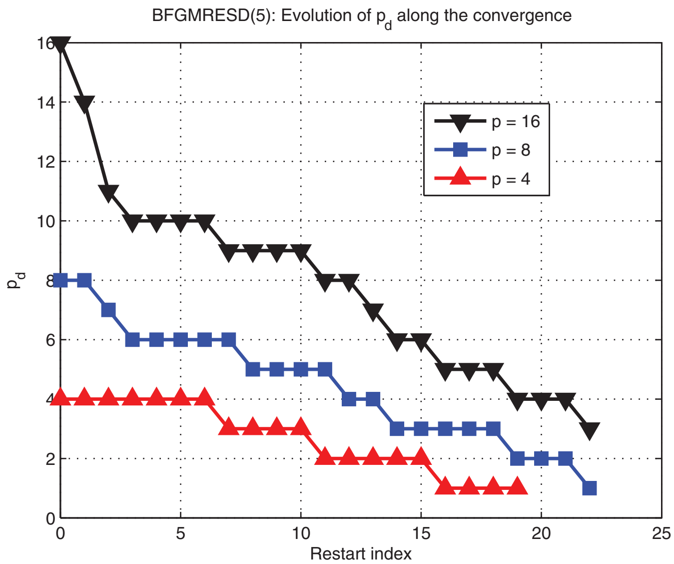

FIG. 4.1. *Perturbed two-grid preconditioned flexible methods for the solution of the Helmholtz equation for the SEG/EAGE overthrust model. Case of $f = 14.53$ Hz ($h = 12.5$ m). Evolution of p_d (the number of considered linear systems) in BFGMRESD*(5) *versus the restarts for three different cases* ($p = 4$, $p = 8$ *and* $p = 16$).

FGMRES(5) simultaneous requires in most cases almost the same number of preconditioner applications as FGMRES(5) sequence. The small differences in terms of iteration are due to the blockwise detection of convergence used in FGMRES(5) simultaneous. The computational times are generally lower due to the use of blocking for both communications and numerical operations.

Whatever the frequency and the number of right-hand sides we remark that BFGMRES(5) requires a number of preconditioner applications similar to those obtained when solving the given linear systems in sequence (FGMRES(5) sequence). The cost of the block orthogonalization procedure ($70np^2 + 40np$ as stated in Table 3.1 with $m = 5$) clearly affects the computational times for the largest value of p. In comparison, the cost of the (unblocked) orthogonalization procedure used in FGMRES(5) sequence behaves as $(5nm + 2nm^2)p$, i.e., $75np$ with $m = 5$. We also notice that BFGMRES(5) is almost equivalent to the second strategy (FGMRES(5) simultaneous) in terms of preconditioner applications and elapsed times. On these nine problems this shows that there is no clear benefit to using *standard* flexible variant of block GMRES methods.

Among the six strategies, BFGMRESD(5) always delivers the minimal number of preconditioner applications and computational times. (See bold values in Tables 4.1, 4.2, and 4.3.) This clearly highlights the interest of deflation at each restart. In the largest-frequency case ($f = 14.53$ Hz), Figure 4.1 shows the evolution of p_d at each restart for the three different cases. The effective block size reduction is clearly shown. Thus detecting the convergence of linear combinations of solutions allows us to reduce the elapsed times at the same memory cost as BFGMRES(5). For instance, we obtain a gain of about 47% in computational time at $f = 14.53$ Hz for $p = 16$ (2242 s versus 4192 s in Table 4.3). Moreover, we note that at a fixed frequency the computational times related to BFGMRESD(5) are always *decreasing independently of the number of cores*. This is especially appealing since a scalable method with respect to the number of right-hand sides would yield almost constant elapsed times at a given frequency.

Finally we also remark that the use of truncation techniques leads to an efficient method at a reduced cost in memory. In certain cases BFGMREST(5, p_f) is as efficient as BFGMRESD(5) (see, e.g., the case $p = 16$ in Table 4.2 showing almost equivalent computational times (707 s versus 695 s) but with a reduction in maximal memory by a factor of 1.8 for BFGMREST(5, p/2)). This feature is really important in this given application due to the very large size of the linear systems.

The proposed flexible block variants (Algorithms 2, 3, and 4) rely on a simple block orthogonalization procedure (Algorithm 1) that does not take into account the possible rank-deficiencies of V_1 or S. An improved block orthogonalization procedure would thus consider these possible rank-deficiencies by incorporating both initial and Arnoldi deflations as suggested in [20]. Rank-revealing QR-factorizations would be used for that purpose. We leave this point for a future work and note that the rank-deficiencies of V_1 or S have never occurred in the numerical experiments detailed in this paper.

In [36, section 2.6.4] the same six strategies have been evaluated on academic testcases related to two-dimensional partial differential equations (pure diffusion and convection-diffusion problems with dominating convection respectively) with a number of right-hand sides ranging from 5 to 160. A cycle of GMRES(m) with $m = 5$ has been used as a variable preconditioner in all methods. Among the six strategies we note that BFGMRESD(5) has always delivered the minimal number of preconditioner applications and that the use of truncation techniques has been found to be effective at a reduced cost in memory. This is thus a similar behavior compared to the proposed application in geophysics.

5. Conclusion. In this paper we have extended the block restarted GMRES method to a variant that allows the use of variable preconditioning when solving multiple right-hand-side problems given at once. Furthermore, we have proposed two variants of block flexible restarted GMRES that rely on deflation. This procedure performed at each restart aims at detecting the possible convergence of a linear combination of the components of the block solution vector. We have also studied the convergence properties of those variants and have shown that the Frobenius norm of the block residual is always nonincreasing. Finally we have highlighted the efficiency of the block flexible methods on a realistic application in geophysics requiring the solution of challenging multiple right-hand-side problems. Block flexible methods with deflation and truncation have proved to be efficient in a constrained memory environment, a nice feature when handling linear systems with billion of unknowns, as frequently required in this application field.

Acknowledgments. The authors are grateful to the reviewers for their valuable remarks and comments that improved the article. The authors would like to acknowledge GENCI (Grand Equipement National de Calcul Intensif) for the dotation of computing hours on the IBM Blue Gene/P computer at IDRIS, France.

REFERENCES

[1] J. I. Aliaga, D. L. Boley, R. W. Freund, and V. Hernández, *A Lanczos-type method for multiple starting vectors*, Math. Comp., 69 (2000), pp. 1577–1601.

[2] F. Aminzadeh, J. Brac, and T. Kunz, 3*D Salt and Overthrust models*, in SEG/EAGE Modeling Series I, Society of Exploration Geophysicists, 1997.

[3] O. Axelsson and P. S. Vassilevski, *A black box generalized conjugate gradient solver with inner iterations and variable-step preconditioning*, SIAM J. Matrix Anal. Appl., 12 (1991), pp. 625–644.

[4] J. Baglama, D. Calvetti, G. H. Golub, and L. Reichel, *Adaptively preconditioned GMRES algorithms*, SIAM J. Sci. Comput., 20 (1998), pp. 243–269.

[5] Z. Bai, D. Day, and Q. Ye, *ABLE: An adaptive block Lanczos for non Hermitian eigenvalue problems*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 1060–1082.

[6] A. H. Baker, J. M. Dennis, and E. R. Jessup, *An efficient block variant of GMRES*, SIAM J. Sci. Comput., 27 (2006), pp. 1608–1626.

[7] A. H. Baker, R. D. Falgout, T. V. Kolev, and U. M. Yang, *Multigrid smoothers for ultraparallel computing*, SIAM J. Sci. Comput., 33 (2011), pp. 2864–2887.

[8] J.-P. Berenger, *A perfectly matched layer for absorption of electromagnetic waves*, J. Comput. Phys., 114 (1994), pp. 185–200.

[9] J.-P. Berenger, *Three-dimensional perfectly matched layer for absorption of electromagnetic waves*, J. Comput. Phys., 127 (1996), pp. 363–379.

[10] P. A. Businger and G. Golub, *Linear least squares solutions by Householder transformations*, Numer. Math., 7 (1965), pp. 269–276.

[11] G. Cohen, *Higher-Order Numerical Methods for Transient Wave Equations*, Springer, New York, 2002.

[12] J. Cullum and T. Zhang, *Two-sided Arnoldi and non-symmetric Lanczos algorithms*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 303–319.

[13] L. Elbouyahyaoui, A. Messaoudi, and H. Sadok, *Algebraic properties of the block GMRES and block Arnoldi methods*, Electron. Trans. Numer. Anal., 33 (2009), pp. 207–220.

[14] H. Elman, O. Ernst, D. O'Leary, and M. Stewart, *Efficient iterative algorithms for the stochastic finite element method with application to acoustic scattering*, Comput. Methods Appl. Mech. Engrg., 194 (2005), pp. 1037–1055.

[15] J. Erhel, K. Burrage, and B. Pohl, *Restarted GMRES preconditioned by deflation*, J. Comput. Appl. Math., 69 (1996), pp. 303–318.

[16] R. W. 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.

[17] G. H. Golub and C. F. V. Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, 1996.

[18] A. E. 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.

[19] M. H. Gutknecht and T. Schmelzer, *The block grade of a block Krylov space*, Linear Algebra Appl., 430 (2009), pp. 174–185.

[20] 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. Siddiqi, I. Duff, and O. Christensen, eds., Anamaya Publishers, New Delhi, India, 2006, pp. 420–447.

[21] G.-D. Gu and Z. Cao, *A block GMRES method augmented with eigenvectors*, Appl. Math. Comput., 121 (2001), pp. 271–289.

[22] N. J. Higham, *Functions of Matrices: Theory and Computation*, SIAM, Philadelphia, 2008.

[23] Y. P. Hong and C. T. Pan, *Rank revealing QR factorizations and the singular value decomposition*, Math. Comp., 58 (1992), pp. 213–232.

[24] R. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1991.

[25] I. M. Jaimoukha and E. M. Kasenally, *Krylov subspace methods for solving large Lyapounov equations*, SIAM J. Numer. Anal., 31 (1994), pp. 227–251.

[26] K. Jbilou, A. Messaoudi, and H. Sadok, *Global FOM and GMRES algorithms for matrix equations*, Appl. Numer. Math., 31 (1999), pp. 49–63.

[27] J. Langou, *Iterative Methods for Solving Linear Systems with Multiple Right-Hand Sides*, Ph.D. thesis, CERFACS, 2003.

[28] J. Langou, *For a few iterations less*, Eighth Copper Mountain Conference on Iterative Methods, Copper Mountain, CO, 2004.

[29] G. Li, *A block variant of the GMRES method on massively parallel processors*, Parallel Comput., 23 (1997), pp. 1005–1019.

[30] D. Loher, *Reliable Nonsymmetric Block Lanczos Algorithms*, Ph.D. thesis, Swiss Federal Institute of Technology Zurich (ETHZ), Switzerland, 2006.

[31] R. B. Morgan, *Restarted block GMRES with deflated restarting*, Appl. Numer. Math., 54 (2005), pp. 222–236.

[32] A. A. Nikishin and A. Y. Yeremin, *Variable block CG algorithms for solving large sparse symmetric positive definite linear systems on parallel computers,* I: *General iterative scheme*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 1135–1153.

[33] Y. Notay and P. S. Vassilevski, *Recursive Krylov-based multigrid cycles*, Numer. Linear Algebra Appl., 15 (2008), pp. 473–487.

[34] Y. Notay, *Flexible conjugate gradients*, SIAM J. Sci. Comput., 22 (2000), pp. 1444–1460.

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

[36] X. Pinel, *A Perturbed Two-Level Preconditioner for the Solution of Three-Dimensional Heterogeneous Helmholtz Problems with Applications to Geophysics*, Ph.D. thesis, CERFACS, 2010.

[37] M. Robbé and M. Sadkane, *Exact and inexact breakdowns in the block GMRES method*, Linear Algebra Appl., 419 (2006), pp. 265–285.

[38] A. Ruhe, *Implementation aspects of band Lanczos algorithms for computation of eigenvalues of large sparse symmetric matrices*, Math. Comp., 33 (1979), pp. 680–687.

[39] Y. Saad, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM J. Sci. Statist. Comput., 14 (1993), pp. 461–469.

[40] V. Simoncini and E. Gallopoulos, *A hybrid block GMRES method for nonsymmetric systems with multiple right-hand sides*, J. Comput. Appl. Math., 66 (1995), pp. 457–469.

[41] 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.

[42] V. Simoncini and E. Gallopoulos, *Convergence properties of block GMRES and matrix polynomials*, Linear Algebra Appl., 247 (1996), pp. 97–119.

[43] V. Simoncini and D. B. Szyld, *Flexible inner-outer Krylov subspace methods*, SIAM J. Numer. Anal., 40 (2003), pp. 2219–2239.

[44] P. Soudais, *Iterative solution methods of a* 3*-D scattering problem from arbitrary shaped multidielectric and multiconducting bodies*, IEEE Trans. Antennas and Propagation, 42 (1994), pp. 954–959.

[45] F. Sourbier, S. Operto, J. Virieux, P. Amestoy, and J. Y. L. Excellent, *FWT*2*D: A massively parallel program for frequency-domain full-waveform tomography of wide-aperture seismic data, part* 1: *Algorithm*, Comput. Geosci., 35 (2009), pp. 487–495.

[46] F. Sourbier, S. Operto, J. Virieux, P. Amestoy, and J. Y. L. Excellent, *FW*2*D : A massively parallel program for frequency-domain full-waveform tomography of wide-aperture seismic data, part* 2: *Numerical examples and scalability analysis*, Comput. Geosci., 35 (2009), pp. 496–514.

[47] D. B. Szyld and J. A. Vogel, *FQMR: A flexible quasi-minimal residual method with inexact preconditioning*, SIAM J. Sci. Comput., 23 (2001), pp. 363–380.

[48] A. Toselli and O. Widlund, *Domain Decomposition methods: Algorithms and Theory*, Springer Ser. Comput. Math. 34, Springer, New York, 2004.

[49] U. Trottenberg, C. W. Oosterlee, and A. Schüller, *Multigrid*, Academic Press, New York, 2001.

[50] H. A. van der Vorst and C. Vuik, *GMRESR: A family of nested GMRES methods*, Numer. Linear Algebra Appl., 1 (1994), pp. 369–386.

[51] J. Virieux and S. Operto, *An overview of full waveform inversion in exploration geophysics*, Geophys., 74 (2009), pp. WCC127–WCC152.

[52] B. Vital, *Etude de Quelques Méthodes de Résolution de Problème Linéaire de Grande Taille sur Multiprocesseur*, Ph.D. thesis, Université de Rennes, 1990.