Electronic Transactions on Numerical Analysis.
Volume 58, pp. 470–485, 2023.

ISSN 1068–9613.
DOI: 10.1553/etna_vol58s470

ETNA
Kent State University and
Johann Radon Institute (RICAM)

AN ENHANCEMENT OF THE CONVERGENCE OF THE IDR METHOD*

F. BOUYGHF†‡, A. MESSAOUDI†, AND H. SADOK‡

Abstract. In this paper, we consider a family of algorithms, called IDR, based on the induced dimension reduction theorem. IDR is a family of efficient short recurrence methods introduced by Sonneveld and Van Gijzen for solving large systems of nonsymmetric linear equations. These methods generate residual vectors that live in a sequence of nested subspaces. We present the IDR(s) method and give two improvements of its convergence. We also define and give a global version of the IDR(s) method and describe a partial and a complete improvement of its convergence. Moreover, we recall the block version and state its improvements. Numerical experiments are provided to illustrate the performances of the derived algorithms compared to the well-known classical GMRES method and the bi-conjugate gradient stabilized method for systems with a single right-hand side, as well as the global GMRES, the global bi-conjugate gradient stabilized, the block GMRES, and the block bi-conjugate gradient stabilized methods for systems with multiple right-hand sides.

Key words. linear equations, iterative methods, IDR method, Krylov subspace, global and block Krylov subspace methods

AMS subject classifications. 65F45, 65F

1. Introduction.

The aim of the IDR method studied in this paper is to solve the following nonsymmetric linear system

$$Ax = b, \tag{1.1}$$

where A is a matrix in $\mathbb{C}^{N\times N}$ and the vectors b and x are in $\mathbb{C}^N$. The IDR method is a short recurrence method developed by Sonneveld and Van Gizen [19]. It is shown by Simoncini and Szyld [18] that IDR(s) is a Petrov-Galerkin-type method with a particular choice of the left Krylov subspace just like other well-known Krylov subspace methods (FOM, GMRES, Lanczos, Hessenberg, QMR); for more details on these methods, see, e.g., [14, 16, 21]. Using the characterization of the left inverse of the Krylov matrix, a unified approach for all Krylov subspace methods was given in [1].

The rest of this paper is organized as follows: in the next section we give a brief overview of the IDR(s) method. Then, we propose an improvement of the convergence of the IDR(s) algorithm using orthogonal projectors. A partial and a full improvement of the IDR(s) method are proposed, called PEnha-IDR(s) and FEnha-IDR(s) method, respectively. In Section 3, we focus on the solution of linear systems with multiple right-hand sides. We define the global version and recall the block version of the IDR(s) method, referred to as global IDR(s) (Gl-IDR(s)) and block IDR(s) (Bl-IDR(s)) method. We also propose two improvements of these methods, a partial and a full improvement, which will be referred to as enhanced global and block IDR(s) and denoted by Gl-PEnha-IDR(s) and Bl-FEnha-IDR(s), respectively. In Section 4, we present some numerical experiments to compare the proposed algorithms with the well-known GMRES [17], the bi-conjugate gradient stabilized (BiCGStab) [21], the global GMRES (Gl-GMRES) [11], the global bi-conjugate gradient stabilized (Gl-BiCGStab) [12], the block GMRES (Bl-GMRES) [22], and the block bi-conjugate gradient stabilized (Bl-BiCGStab) [5] methods.

*Received January 24, 2023. Accepted April 10, 2023. Published online on July 3, 2023. Recommended by L. Reichel.

†LabMIA-SI, Mohammed V University in Rabat, Ecole Normale Supérieure, Av. Mohammed Belhassan El Ouazzani, B.P. 5118, Takaddoum, Rabat, Maroc.

‡LMPA, Université du Littoral Côte d'Opale, 50 rue F. Buisson BP 699, 62228 Calais Cedex, France (hassane.sadok@univ-littoral.fr).

ETNA
Kent State University and
Johann Radon Institute (RICAM)

Throughout this article, all vectors and matrices are assumed to be complex, and the following notation is used: First, M^H represents the conjugate transpose of a matrix M. For two vectors x and y in $\mathbb{C}^N$, the inner product is $\langle x, y\rangle = x^H y$, with $\|x\| = \sqrt{\langle x, x\rangle}$ the Euclidean norm. In the block and global cases, we consider matrices X and Y in $\mathbb{C}^{N\times m}$, for which the inner product is defined by $\langle X, Y\rangle_F = \mathrm{Tr}\left(X^H Y\right)$, where $\mathrm{Tr}\,(Z)$ denotes the trace of a square matrix Z. Moreover, the associated norm is the Frobenius norm indicated by $\|.\|_F$. We denote by I the identity matrix of order N.

2. The IDR(s) method. The IDR(s) method is a variant of the IDR algorithm using s shadow vectors, developed by Sonneveld and Van Gizen [19]. The subspaces used by the IDR algorithms are related to Krylov subspaces. We will first recall the definition of a Krylov subspace of order n associated to the matrix A and the vector r_0 by

$$K_n(A, r_0) = \mathrm{span}\{r_0, Ar_0, \ldots, A^{n-1}r_0\},$$

where $r_0 = b - Ax_0$, with x_0 an initial guess approximation of the solution of the system (1.1).

2.1. The IDR theorem. The IDR(s) method is based on the following Induction Dimension Reduction (IDR) theorem [19], which is a generalization of the original IDR theorem [20] to the complex case; we first review this theorem.

THEOREM 2.1 (IDR [19]). *Let A be any matrix in $\mathbb{C}^{N\times N}$, and let $\mathcal{G}_0$ be the full Krylov space $K_N(A, r_0)$. Let $\mathcal{S}$ be any eigensubspace of $\mathbb{C}^N$ such that $\mathcal{S}$ and $\mathcal{G}_0$ do not share a nontrivial invariant subspace of A, and define the sequence $\mathcal{G}_j$, $j = 1, 2, \ldots$, as*

$$\mathcal{G}_j = (I - \omega_j A)\,(\mathcal{G}_{j-1} \cap \mathcal{S})\,,$$

where the ω_j's are nonzero complex scalars. Then the following statements hold:

- $\mathcal{G}_j \subset \mathcal{G}_{j-1}$, $\forall j > 0$.
- $\mathcal{G}_\nu = \{\mathbf{0}\}$, *for some* $\nu \leq N$.

From this theorem, we know that it is possible to generate an appropriate sequence of nested subspaces of decreasing dimension and that under mild conditions the smallest possible subspace is $\{\mathbf{0}\}$. A Krylov-type solver produces iterates x_n for which the residuals $r_n = b - Ax_n$ are in the Krylov spaces $K_n(A, r_0)$. Here, x_0 is an initial estimate of the solution. If all the residuals r_n can be constructed in the nested subspaces $\mathcal{G}_j$, then we may get the approximate solution in finite steps. At most $N + N/s$ matrix-vector products are needed in the generic case for the IDR(s) method [19].

2.2. The IDR(s) algorithm. Consider $\mathcal{S} = \mathcal{N}\left(P^H\right)$, where $P = [p_1\ p_2\ \cdots\ p_s]$ is a full-rank matrix in $\mathbb{C}^{N\times s}$, with $s \ll N$ and $\mathcal{N}$ denoting the null space. For all nonzero integers j, the IDR spaces are recursively defined as

$$\mathcal{G}_j = (I - \omega_j A)\,(\mathcal{G}_{j-1} \cap \mathcal{S})\,.$$

According to the IDR theorem, for all $j \leq N$, we have $\mathcal{G}_j \subset \mathcal{G}_{j-1}$, and there exists $\nu \leqslant N$ such that $\mathcal{G}_\nu = \{\mathbf{0}\}$. Thus, the IDR theorem can be used to develop an algorithm for solving linear systems. This is done by constructing residuals with $r_n \in \mathcal{G}_j$, because according to this theorem, it is possible to generate a sequence of smallest possible subspaces. Then, the aim of Sonneveld's approach is, first to construct subspaces $\mathcal{G}_j$ for all nonzero integer j, where ω_j are nonzero scalars. Then, we compute the approximate solution x_n associated to the residual vector $r_n = b - Ax_n$, which is necessarily in $\mathcal{G}_j$. Thus, the residual $r_n \in \mathcal{G}_j$ can be written as

$$r_n = (I - \omega_j A)\, v_{n-1} \quad \text{with} \quad v_{n-1} \in \mathcal{G}_{j-1} \cap \mathcal{S}. \tag{2.1}$$

ETNA
Kent State University and
Johann Radon Institute (RICAM)

Since, there exists ν such that $\mathcal{G}_\nu = \{\mathbf{0}\}$, it follows that there exists an integer m such that $r_m \in \mathcal{G}_\nu = \{0\}$. The scalars ω_j are chosen such that $\|r_n\|$ is minimal.

Now, to compute r_n at each iteration, it is necessary to compute v_{n-1}, and this can be done by using the fact that $v_{n-1} \in \mathcal{S}$. In [8], Gutknecht considered

$$v_{n-1} = r_{n-1} - G_{n-1}c, \tag{2.2}$$

where for all integers $l \leq s$, the vector $c = [\delta_l, \ldots, \delta_1]^T$ is in $\mathbb{C}^l$ and the matrix $G_{n-1} = [g_{n-1-l} \ \cdots \ g_{n-2}] \in \mathbb{C}^{N\times l}$, with $g_i \in \mathcal{G}_{j-1}$, for $i = n-l-1, \ldots, n-2$. From equations (2.1) and (2.2), we get

$$r_n = r_{n-1} - \omega_j A v_{n-1} - G_{n-1}c.$$

REMARK 2.2. In order to determine the s variables δ_i, the space $\mathcal{S}$ can be chosen to be the left null space of some $N \times s$ matrix $P = [p_1 \ p_2 \ \cdots \ p_s]$, i.e., $\mathcal{S} = \mathcal{N}(P^H)$, which can be generated randomly since the probability that the space $\mathcal{G}_0 \cap \mathcal{S}$ contains some eigenvectors of A is zero. Then δ_i can be determined from the equation

$$P^H v_{n-1} = 0.$$

We therefore obtain the $s \times l$ system

$$\begin{cases} p_1^H r_{n-1} - p_1^H \left(\displaystyle\sum_{i=1}^{l} \delta_i g_{n-i} \right) = p_1^H r_{n-1} - p_1^H \left(G_{n-1} c \right) = 0 \\ \qquad\qquad\qquad\qquad \vdots \\ p_s^H r_{n-1} - p_s^H \left(\displaystyle\sum_{i=1}^{l} \delta_i g_{n-i} \right) = p_s^H r_{n-1} - p_s^H \left(G_{n-1} c \right) = 0. \end{cases}$$

Under normal circumstances the previous system is uniquely solvable if $l = s$. Then, to compute all scalars δ_i, for $i = 1, \ldots, s$, we need s vectors in $\mathcal{G}_j$. Consequently, computing the first vector in $\mathcal{G}_j$ requires $s+1$ vectors in $\mathcal{G}_{j-1}$, and we may expect r_n to be in $\mathcal{G}_j$ only for $n \geq j(s+1)$. Define the matrices

$$G_{n-1} = \Delta R_n = [\Delta r_{n-1} \ \ \Delta r_{n-2} \ \ \cdots \ \ \Delta r_{n-s}],$$

and

$$\Delta X_n = [\Delta x_{n-1} \ \ \Delta x_{n-2} \ \ \cdots \ \ \Delta x_{n-s}],$$

where the forward difference operator $\Delta u_n = u_{n+1} - u_n$ is used. Then the computation of $r_n \in \mathcal{G}_j$ can be implemented by the following algorithm:

$$\begin{cases} \text{Calculate } c \in \mathbb{C}^s \text{ from } \left(P^H \Delta R_n \right) c = P^H r_{n-1}, \\ \qquad v = r_{n-1} - \Delta R_{n-1} c, \\ \qquad r_n = v - \omega_j A v. \end{cases}$$

Since $\mathcal{G}_{j-1} \subset \mathcal{G}_j$, repeating these calculations will produce new residuals $r_{n+1}, r_{n+2}, \ldots$ in $\mathcal{G}_j$. Once $s+1$ residuals in $\mathcal{G}_j$ have been computed, we can expect the next residual to be in $\mathcal{G}_{j+1}$. The approximation x_n associated with the residual $r_n = b - Ax_n$ is given by

$$x_n = x_{n-1} + \omega_j v - \Delta X_{n-1} c.$$

Putting all the relations together, we get the IDR(s) algorithm stated in Algorithm 1.

ETNA
Kent State University and
Johann Radon Institute (RICAM)

Algorithm 1 IDR(s) Algorithm [19].

1. $A \in \mathbb{C}^{N\times N}$, x_0, $b \in \mathbb{C}^N$, $P \in \mathbb{C}^{N\times s}$, $P = orth(P)$, $tol \in [0,1]$, $itemax > 0$, $r_0 = b - Ax_0$;
2. for $n = 0$ to $s-1$ do (build s vectors in $\mathcal{G}_0$)
3. $\quad v = Ar_n$;
4. $\quad \omega = (v^H r_n)/(v^H v)$;
5. $\quad \Delta x_n = \omega r_n$;
6. $\quad \Delta r_n = -\omega v$;
7. $\quad r_{n+1} = r_n + \Delta r_n$;
8. $\quad x_{n+1} = x_n + \Delta x_n$;
9. end for
10. $\Delta R_{n+1} = [\Delta r_n \ \cdots \ \Delta r_0]$;
11. $\Delta X_{n+1} = [\Delta x_n \ \cdots \ \Delta x_0]$;
12. $n = s$;
13. while $\|r_n\| \,/\, \|b\| > tol$ and $n < itemax$ do
14. $\quad$ for $k = 0$ to s do (build s vectors of $\mathcal{G}_j$)
15. $\quad\quad$ solve c from $P^H \Delta R_n c = P^H r_n$;
16. $\quad\quad$ compute $q = -\Delta R_n c$, $v = r_n - q$;
17. $\quad\quad$ if $k = 0$ then
18. $\quad\quad\quad t = Av$; $\omega = (t^H v)/(t^H t)$;
19. $\quad\quad\quad \Delta r_n = q - \omega t$;
20. $\quad\quad\quad \Delta x_n = -\Delta X_n c + \omega v$;
21. $\quad\quad$ else
22. $\quad\quad\quad \Delta x_n = -\Delta X_n c + \omega v$;
23. $\quad\quad\quad \Delta r_n = -A\Delta x_n$;
24. $\quad\quad$ end if
25. $\quad\quad r_{n+1} = r_n + \Delta r_n$;
26. $\quad\quad x_{n+1} = x_n + \Delta x_n$;
27. $\quad\quad n = n + 1$;
28. $\quad\quad \Delta R_n = [\Delta r_{n-1} \ \cdots \ \Delta r_{n-s}]$;
29. $\quad\quad \Delta X_n = [\Delta x_{n-1} \ \cdots \ \Delta x_{n-s}]$;
30. $\quad$ end for
31. end while.

2.3. Partial and full enhancement of the convergence of the IDR(s) method. We propose an improvement of the convergence of the IDR(s) method. Two enhancements of this method are studied, the first one called partial enhancement, denoted by PEnha-IDR(s), and the second one called full enhancement, denoted by FEnha-IDR(s). We propose to improve the convergence of the IDR(s) method by using the following well-known result:

PROPOSITION 2.3. *Consider the orthogonal projector*

$$Q_l = I - Z_l Z_l^{\dagger},$$

where the rectangular matrix Z_l is a full-rank matrix in $\mathbb{R}^{N\times l}$ and $Z_l^{\dagger} = \left(Z_l^T Z_l\right)^{-1} Z_l^T$ is its pseudo-inverse (Moore-Penrose inverse) (for more details on the pseudo inverse, see [13]). Applying the projector Q_l to any vector r in $\mathbb{R}^N$, we obtain a new residual, which we denote by

$$r^{Enha} = Q_l r.$$

ETNA
Kent State University and
Johann Radon Institute (RICAM)

Then, we have

$$\left\|r^{Enha}\right\| \leq \|r\| .$$

REMARK 2.4. The matrix computed in Algorithm 1,

$$\Delta R_n = [\Delta r_{n-1} \;\; \Delta r_{n-2} \;\; \cdots \;\; \Delta r_{n-s}] ,$$

is of full rank. Therefore, the pseudo-inverse of ΔR_n is well defined, and its columns can be used for building the orthogonal projector Q_l.

By invoking Proposition 2.3 with the residual vector r_n, we obtain an improvement in the accuracy and stability of the IDR(s) algorithm. Thus, we apply an orthogonal projector Q_l to the residual of this method. To avoid a storage problem, we use the s vectors of $\mathcal{G}_j$ already calculated in the IDR(s) method to construct the orthogonal projector.

The partial enhancement of the convergence of the IDR(s) method (PEnha-IDR(s)) is given by choosing Z_l equal to the last column of ΔR_n ($l = 1$), and by adding the following instructions (using Matlab notation) to line 26 in Algorithm 1:

1. $Z_1 = \Delta R_n(:, n)$;
2. $Z = Z_1^{\dagger} * r_{n+1}$;
3. $x_{n+1}^{PEnha} = x_{n+1} + \Delta X_n(:, n) * Z$;
4. $r_{n+1}^{PEnha} = r_{n+1} - \Delta R_n(:, n) * Z$;

The full enhancement of the convergence of the IDR(s) (FEnha-IDR(s)) method is defined by choosing Z_l equal to ΔR_n ($l = s$), and by adding the following instructions (using Matlab notation) to line 26 in Algorithm 1:

1. $Z_s = \Delta R_n$;
2. $Z = Z_s^{\dagger} * r_{n+1}$;
3. $x_{n+1}^{FEnha} = x_{n+1} + \Delta X_n * Z$;
4. $r_{n+1}^{FEnha} = r_{n+1} - \Delta R_n * Z$;

REMARK 2.5. For building the orthogonal projector Q_l, we can also take some of the last columns of the matrix ΔR_n.

3. The global and block IDR(s) methods. In this section, we consider the solution of large and sparse nonsymmetric systems with multiple right-hand sides of the form

$$AX = B, \tag{3.1}$$

where the coefficient matrix A is a nonsingular complex matrix of order N and the matrices $X = [x_1 \; x_2 \; \cdots \; x_m]$ and $B = [b_1 \; b_2 \; \cdots \; b_m] \in \mathbb{C}^{N \times m}$, with $m \ll N$.

One class of solvers for problem (3.1) are the global methods, which are based on the use of a global projection process onto a matrix (global) Krylov subspace, including the global FOM and GMRES methods [11], the global BCG and Bi-CGStab methods [5], and the global Hessenberg and CMRH methods [10].

The other class is that of block solvers, which are much more efficient when the matrix A is relatively dense. The first block solvers are the block conjugate gradient (Bl-CG) and block bi-conjugate gradient (Bl-BCG) methods proposed in [15], while for nonsymmetric problems, the block generalized minimal residual (Bl-GMRES) algorithm [22], the block quasi-minimum residual (Bl-QMR) algorithm [7], the block Bi-CGStab (Bl-BiCGStab) algorithm [5], and the block Lanczos method [6] have been developed.

ETNA
Kent State University and
Johann Radon Institute (RICAM)

3.1. The global IDR(s) method. In this section we recall some matrix products that will be used to define the global version of the IDR(s) method for solving the system of nonsymmetric linear equations (3.1). As for the IDR(s) method, partial and full enhancements of this version are proposed.

We recall the definition of the Kronecker product [13] and of the $\diamond$-product [2] and give some fundamental properties of the latter. A matrix system of $\mathbb{C}^{N\times m}$ is said to be F-orthonormal if it is orthonormal with respect to $\langle Y, Z\rangle_F = Tr(Y^H Z)$. For any matrix $A = (a_{i,j})$ and any matrix B, the Kronecker product of A and B is defined by $A\otimes B = [a_{i,j}B]$. In what follows, we refer to the product denoted by $\diamond$ that is defined as follows [2]:

DEFINITION 3.1. *Let $A = [A_1\ A_2\ \cdots\ A_s]$ and $B = [B_1\ B_2\ \cdots\ B_l]$ be matrices of dimension $N \times sm$ and $N \times lm$, respectively, where A_i and B_j $(i = 1, \ldots, s,\ j = 1, \ldots, l)$ are $N \times m$ matrices. Then the $s \times l$ matrix $A^T \diamond B$ is defined by*

$$A^T \diamond B = \begin{bmatrix} \langle A_1, B_1\rangle_F & \langle A_1, B_2\rangle_F & \cdots & \langle A_1, B_l\rangle_F \\ \langle A_2, B_1\rangle_F & \langle A_2, B_2\rangle_F & \cdots & \langle A_2, B_l\rangle_F \\ \vdots & \vdots & \vdots & \vdots \\ \langle A_s, B_1\rangle_F & \langle A_s, B_2\rangle_F & \cdots & \langle A_s, B_l\rangle_F \end{bmatrix}.$$

REMARKS 3.2.

1. If $m = 1$, then $A^T \diamond B = A^T B$.
2. If $m = 1$, $s = 1$, and $l = 1$, then setting $A = u \in \mathbb{C}^N$, $B = v \in \mathbb{C}^N$, we have $A^T \diamond B = u^H v \in \mathbb{C}$.
3. The matrix $A = [A_1\ A_2\ \cdots\ A_s]$ is F-orthonormal if and only $A^T \diamond A = I_s$.
4. If $X \in \mathbb{C}^{N\times m}$, then $X^T \diamond X = \|X\|_F^2$.

We state some properties of this product combined with the Kronecker product.

PROPOSITION 3.3.

Let $A, B, C \in \mathbb{C}^{N\times sm}$, $D \in \mathbb{C}^{N\times N}$, $L \in \mathbb{C}^s$, and $\alpha \in \mathbb{C}$. Then we have

1. $(A + B)^T \diamond C = A^T \diamond C + B^T \diamond C$.
2. $A^T \diamond (B + C) = A^T \diamond B + A^T \diamond C$.
3. $(\alpha A)^T \diamond C = \alpha(A^T \diamond C)$.
4. $(A^T \diamond B)^T = B^T \diamond A$.
5. $(DA)^T \diamond B = A^T \diamond (D^T B)$.
6. $\|A^T \diamond B\|_F \le \|A\|_F \|B\|_F$.
7. $A^T \diamond (B(L \otimes I_m)) = (A^T \diamond B)L$.

Proof. The first six assertions are proved in [2]. We will prove the last one. We define $A = [A_1\ A_2\ \cdots\ A_s] \in \mathbb{C}^{N\times sm}$, $B = [B_1\ B_2\ \cdots\ B_s] \in \mathbb{C}^{N\times sm}$, with $A_i, B_i \in \mathbb{C}^{N\times m}$ and $L = [l_1, l_2, \ldots, l_s]^T \in \mathbb{C}^{s\times 1}$. Then using the definition of the $\diamond$-product we get

$$A^T \diamond (B(L \otimes I_m)) = A^T \diamond \left(\sum_{i=1}^{s} B_i l_i\right) = \sum_{i=1}^{s} \left(A^T \diamond B_i\right) l_i = \left(A^T \diamond B\right) L,$$

and the result follows. ∎

We recall the definition of the global Krylov subspace of order n associated with the matrices A and R_0, where $R_0 = B - AX_0$ with X_0 an initial approximation of the solution X^* of the system (3.1).

DEFINITION 3.4. *The subspace $\mathcal{K}_n^g(A, R_0)$ generated by A and increasing powers of A applied to R_0,*

$$\mathcal{K}_n^g(A, R_0) = \left\{ \sum_{i=1}^{n} A^{i-1} R_0 \gamma_i;\ \gamma_i \in \mathbb{C} \right\},$$

ETNA
Kent State University and
Johann Radon Institute (RICAM)

is called the global Krylov subspace of order n associated with A and R_0; see, e.g., [4]. It can be also defined by

$$\mathcal{K}_n^g(A, R_0) = \operatorname{span}\left\{R_0, AR_0, \ldots, A^{n-1}R_0\right\}.$$

3.1.1. The global IDR(s) algorithm. The global version of the IDR theorem is stated in the following theorem, whose proof is similar to that of the IDR theorem.

THEOREM 3.5 (global IDR). *Let A be any matrix in $\mathbb{C}^{N\times N}$, and let $\mathcal{G}_0^g$ be the full global Krylov space $\mathcal{K}_N^g(A, R_0)$. Let $\mathcal{S}^g$ denote any eigensubspace of $\mathbb{C}^N$ such that $\mathcal{S}^g$ and $\mathcal{G}_0^g$ do not share a nontrivial invariant subspace of A, and define the sequence $\mathcal{G}_j^g$, $j = 1, 2, \ldots$, as*

$$\mathcal{G}_j^g = (I - \omega_j A)\left(\mathcal{G}_{j-1}^g \cap \mathcal{S}^g\right),$$

where the ω_j's are nonzero complex scalars. Then the following conditions are satisfied:

- $\mathcal{G}_j^g \subset \mathcal{G}_{j-1}^g$, $\forall j > 0$.
- $\mathcal{G}_\nu^g = \{\mathbf{0}\}$, *for some* $\nu \leq N$.

The global IDR(s) algorithm is an extension of the IDR(s) algorithm. It can be derived as a consequence of the global IDR theorem. Assume that all column vectors of $R_{n-s}, \ldots, R_{n-1}$ belong to $\mathcal{G}_{j-1}^g$. Then we can construct the global residual R_n, whose column vectors belong to $\mathcal{G}_j^g$, by defining

$$R_n = (I - \omega_j A)V_{n-1},$$

where V_{n-1} is an $N \times m$ matrix such that $V_{n-1} \in \mathcal{G}_{j-1}^g \cap \mathcal{S}^g$ and the scalar parameter ω_j is obtained by minimizing the Frobenius norm of the residual R_n. To obtain such V_{n-1}, suppose that the subspace $\mathcal{S}^g$ can be written as follows:

$$\mathcal{S}^g = \mathcal{N}(\mathbf{P}^H) = \{Z \in \mathbb{C}^{N\times m}; \mathbf{P}^T \diamond Z = 0_{s\times 1}\},$$

for a certain $N \times sm$ matrix $\mathbf{P}$. Let

$$V_{n-1} = R_{n-1} - \sum_{i=1}^{s} \Delta R_{n-1-i}\delta_i, \quad \text{where} \quad \Delta R_k = R_{k+1} - R_k.$$

Then the condition $V_{n-1} \in \mathcal{S}^g$ can be rephrased as

$$\mathbf{P}^T \diamond V_{n-1} = 0_{s\times 1}. \tag{3.2}$$

The coefficients $\delta_1, \delta_2, \ldots, \delta_s$ are obtained by solving the previous equation.

Define $c = [\delta_1, \delta_2, \ldots, \delta_s]^T \in \mathbb{C}^s$ and the matrices

$$\Delta R_n^g = [\Delta R_{n-1} \;\; \Delta R_{n-2} \;\; \cdots \;\; \Delta R_{n-s}],$$

and

$$\Delta X_n^g = [\Delta X_{n-1} \;\; \Delta X_{n-2} \;\; \cdots \;\; \Delta X_{n-s}].$$

Then the matrix R_n can be written as

$$\begin{aligned} R_n = (I - \omega_j A)V_{n-1} &= (I - \omega_j A)\left(R_{n-1} - \sum_{i=1}^{s} \Delta R_{n-1-i}\delta_i\right) \\ &= R_{n-1} - \omega_j A V_{n-1} - \Delta R_n^g (c \otimes I_m), \end{aligned}$$

ETNA
Kent State University and
Johann Radon Institute (RICAM)

and using Proposition 3.3, equation (3.2) becomes

$$\begin{aligned}\mathbf{P}^T \diamond V_{n-1} &= \mathbf{P}^T \diamond \left(R_{n-1} - \sum_{i=1}^{s} \Delta R_{n-1-i}\delta_i \right) \\ &= \mathbf{P}^T \diamond R_{n-1} - \mathbf{P}^T \diamond (\Delta R_n^g (c \otimes I_m)) \\ &= \mathbf{P}^T \diamond R_{n-1} - \left(\mathbf{P}^T \diamond \Delta R_n^g\right) c = 0_{s\times 1}.\end{aligned}$$

The computation of $R_n \in \mathcal{G}_j^g$ is implemented by the following algorithm:

$$\begin{cases} \text{Calculate } c \in \mathbb{C}^s \text{ from } \left(\mathbf{P}^T \diamond \Delta R_n^g\right) c = \mathbf{P}^T \diamond R_{n-1}, \\ V_{n-1} = R_{n-1} - \Delta R_n^g (c \otimes I_m), \\ R_n = V_{n-1} - \omega_j A V_{n-1}. \end{cases}$$

The approximation X_n is obtained as

$$\begin{aligned} X_n &= X_{n-1} + \omega_j V_{n-1} - \sum_{i=1}^{s} \Delta X_{n-1-i}\delta_i \\ &= X_{n-1} + \omega_j V_{n-1} - \Delta X_n^g \left(c \otimes I_m\right). \end{aligned}$$

The scalar ω_j is given by

$$\omega_j = \frac{\langle T, V_{n-1}\rangle_F}{\langle T, T\rangle_F} = \frac{Tr(T^H V_{n-1})}{Tr(T^H T)}, \quad \text{where } T = AV_{n-1}.$$

Finally, we obtain the global IDR(s) algorithm stated in Algorithm 2.

3.1.2. Partial and full enhancement of the global IDR(s) method. As for the IDR(s) method, we propose an improvement of the convergence of the global IDR(s) method. Two enhancements of these methods are studied, the first one is called global partial enhancement, denoted by Gl-PEnha-IDR(s), and the second one is called global full enhancement, denoted by Gl-FEnha-IDR(s). We propose to improve the convergence of the global IDR(s) method by using the following result:

PROPOSITION 3.6. *Consider the orthogonal projector*

$$\mathcal{Q}_l = I - \mathcal{Z}_l \mathcal{Z}_l^{\dagger},$$

where the rectangular matrix $\mathcal{Z}_l$ is a full-rank matrix in $\mathbb{R}^{N\times lm}$ and $\mathcal{Z}_l^{\dagger} = \left(\mathcal{Z}_l^T \mathcal{Z}_l\right)^{-1} \mathcal{Z}_l^T$ its pseudo-inverse (Moore-Penrose inverse). By applying the projector $\mathcal{Q}_l$ to any matrix $R \in \mathbb{R}^{N\times m}$, we obtain a new residual that we denote by

$$R^{Enha} = \mathcal{Q}_l R.$$

Then, we have

$$\left\| R^{Enha} \right\|_F \leq \|R\|_F .$$

By invoking Proposition 3.6 with the residual vector R_n, we obtain an improvement of the global IDR(s) algorithm: We apply an orthogonal projector $\mathcal{Q}_l$ to the residual of this method by using the s matrices of $\mathcal{G}_j^g$ that are already computed in the global IDR(s) method to construct the orthogonal projector.

The partial improvement of the convergence of the global IDR(s) (Gl-PEnha-IDR(s)) method is given by choosing $\mathcal{Z}_l$ equal to the last column matrix of ΔR_n^g ($l = 1$) and by adding the following instructions (using Matlab notation) to line 27 in Algorithm 2:

ETNA
Kent State University and
Johann Radon Institute (RICAM)

Algorithm 2 Global IDR(s) algorithm (Gl-IDRs)).

1. $A \in \mathbb{C}^{N\times N}$, X_0, $B \in \mathbb{C}^{N\times m}$, $\mathbf{P} \in \mathbb{C}^{N\times sm}$, $\mathbf{P} = orth(\mathbf{P})$, $tol \in [0,1]$, $itemax > 0$, $R_0 = B - AX_0$;
2. for $n = 0$ to $s-1$ do (build s matrices in $\mathcal{G}_0^g$)
3. $\quad V = AR_n$;
4. $\quad \omega = \langle V, R_n\rangle_F / \langle V, V\rangle_F$;
5. $\quad \Delta X_n = \omega R_n$;
6. $\quad \Delta R_n = -\omega V$;
7. $\quad R_{n+1} = R_n + \Delta R_n$;
8. $\quad X_{n+1} = X_n + \Delta X_n$;
9. end for
10. $\Delta R_{n+1}^g = [\Delta R_n \ \cdots \ \Delta R_0]$;
11. $\Delta X_{n+1}^g = [\Delta X_n \ \cdots \ \Delta X_0]$;
12. $H = \mathbf{P}^T \diamond \Delta R_{n+1}^g$, $h = \mathbf{P}^T \diamond R_{n+1}$;
13. $n = s$;
14. while $\|R_n\|_F / \|B\|_F > tol$ and $n < itemax$ do
15. $\quad$ for $k = 0$ to s do (build s matrices of $\mathcal{G}_j^g$)
16. $\quad\quad$ solve the system $Hc = h$;
17. $\quad\quad$ compute $Q = -\Delta R_n^g (c \otimes I_m)$, $V = R_n - Q$;
18. $\quad\quad$ if $k = 0$ then
19. $\quad\quad\quad T = AV$; $\omega = \langle T, V\rangle_F / \langle T, T\rangle_F$;
20. $\quad\quad\quad \Delta R_n = Q - \omega T$;
21. $\quad\quad\quad \Delta X_n = -\Delta X_n^g (c \otimes I_m) + \omega V$;
22. $\quad\quad$ else
23. $\quad\quad\quad \Delta X_n = -\Delta X_n^g (c \otimes I_m) + \omega V$;
24. $\quad\quad\quad \Delta R_n = -A\Delta X_n$;
25. $\quad\quad$ end if
26. $\quad\quad R_{n+1} = R_n + \Delta R_n$;
27. $\quad\quad X_{n+1} = X_n + \Delta X_n$;
28. $\quad\quad n = n + 1$;
29. $\quad\quad \Delta R_n^g = [\Delta R_{n-1} \ \cdots \ \Delta R_{n-s}]$;
30. $\quad\quad \Delta X_n^g = [\Delta X_{n-1} \ \cdots \ \Delta X_{n-s}]$;
31. $\quad\quad \Delta h = \mathbf{P}^T \diamond \Delta R_n$;
32. $\quad\quad H(:, n) = \Delta h$;
33. $\quad$ end for
34. end while

1. $\mathcal{Z}_1 = \Delta R_n^g(:, (n-1)m + 1 : nm)$;
2. $\mathcal{Z} = \mathcal{Z}_1^\dagger * R_{n+1}$;
3. $X_{n+1}^{Gl-PEnha} = X_{n+1} + \Delta X_n^g(:, (n-1)m + 1 : nm) * \mathcal{Z}$;
4. $R_{n+1}^{Gl-PEnha} = R_{n+1} - \Delta R_n^g(:, (n-1)m + 1 : nm) * \mathcal{Z}$;

The full improvement of the convergence of the global IDR(s) (Gl-FEnha-IDR(s)) method is defined by choosing $\mathcal{Z}_l$ equal to ΔR_n^g ($l = s$) and by adding the following instructions (using Matlab notation) to line 27 in Algorithm 2:

1. $\mathcal{Z}_s = \Delta R_n^g$;
2. $\mathcal{Z} = \mathcal{Z}_s^\dagger * R_{n+1}$;
3. $X_{n+1}^{Gl-FEnha} = X_{n+1} + \Delta X_n^g * \mathcal{Z}$;

ETNA
Kent State University and
Johann Radon Institute (RICAM)

4. $R_{n+1}^{Gl-FEnha} = R_{n+1} - \Delta R_n^g * \mathcal{Z}$;

3.2. The block IDR(s) method. In this section we consider nonsymmetric linear systems with multiple right-hand sides (3.1). In order to state the block version of IDR(s), we first give a variant of the IDR theorem that is an extension of the IDR theorem to the block case. We also recall the block IDR(s) (Bl-IDR(s)) method as defined in [3], and we propose a partial enhancement (PEnha-Bl-IDR(s)) method and a full enhancement (FEnha-Bl-IDR(s)) of the convergence of this method. We first recall the block Krylov subspace of order n associated with the matrices A and R_0.

DEFINITION 3.7. *The subspace $\mathcal{K}_n^b(A, R_0)$ generated by A and increasing powers of A applied to R_0,*

$$\mathcal{K}_n^b(A, R_0) = \left\{ \sum_{i=1}^{n} A^{i-1} R_0 \gamma_i;\ \gamma_i \in \mathbb{C}^{m\times m} \right\},$$

is called the block Krylov subspace; see, e.g., [4].

Now we recall the definition of the block grade of R_0 with respect A [9].

DEFINITION 3.8. *Let $\mathcal{B}_n(A, R_0)$ be the subspace*

$$\mathcal{B}_n(A, R_0) := \mathcal{K}_n(A, R_0(:, 1)) + \ldots + \mathcal{K}_n(A, R_0(:, m)).$$

Then the positive integer $v(A, R_0)$ defined by

$$\begin{aligned} v(A, R_0) &:= \min\{n|\ \dim(\mathcal{B}_n(A, R_0))\} = \dim(\mathcal{B}_{n+1}(A, R_0)) \\ &= \min\{n|\ \mathcal{B}_n(A, R_0)\} = \mathcal{B}_{n+1}(A, R_0) \end{aligned}$$

is called the block grade of R_0 with respect to A.

REMARK 3.9. If X^* is the exact block solution of $AX = B$, then

$$X^* \in X_0 + \mathcal{K}_{v(A,R_0)}^b(A, R_0).$$

3.2.1. The block IDR(s) algorithm. Now we present the extension of the IDR theorem to the block case given by [3].

THEOREM 3.10 (Block IDR). *Let A be any matrix in $\mathbb{C}^{N\times N}$, and let $\mathcal{G}_0^b$ be the full block Krylov space $\mathcal{K}_{v(A,R_0)}^b(A, R_0)$. Let $\mathcal{S}^b$ denote any eigensubspace of $\mathbb{C}^N$ such that $\mathcal{S}^b$ and $\mathcal{G}_0^b$ do not share a nontrivial invariant subspace of A. Define the sequence $\mathcal{G}_j^b$, $j = 1, 2, \ldots$, as*

$$\mathcal{G}_j^b = (I - \omega_j A)\left(\mathcal{G}_{j-1}^b \cap \mathcal{S}^b\right),$$

where the ω_j's are nonzero complex scalars. Then the following conditions are satisfied:

- $\mathcal{G}_j^b \subset \mathcal{G}_{j-1}^b$, $\forall j > 0$.
- $\mathcal{G}_\nu^b = \{\mathbf{0}\}$, *for some* $\nu \leq v(A, R_0)$.

The block IDR(s) method is a natural extension of the IDR(s) method. It can be considered an implementation of the block IDR theorem. Suppose that all column vectors of $R_{n-s}, \ldots, R_{n-1}$ belong to $\mathcal{G}_{j-1}^b$. Then we can construct the block residual R_n whose column vectors belong to $\mathcal{G}_j^b$ by defining

$$R_n = (I - \omega_j A)V_{n-1},$$

where V_{n-1} is an $N \times m$ matrix whose column vectors belong to $\mathcal{G}_{j-1}^b \cap \mathcal{S}^b$ and the scalar parameter ω_j is obtained by minimizing the Frobenius norm of the block residual R_n. To

ETNA
Kent State University and
Johann Radon Institute (RICAM)

obtain such V_{n-1}, suppose that the subspace $\mathcal{S}^b$ can be written as $\mathcal{S}^b = \mathcal{N}(\mathbf{P}^H)$ for some $N \times sm$ matrix $\mathbf{P}$. Let

$$V_{n-1} = R_{n-1} - \sum_{i=1}^{s} \Delta R_{n-1-i}\gamma_i, \quad \text{where} \quad \Delta R_k = R_{k+1} - R_k.$$

Then the condition $V_{n-1} \in \mathcal{S}^b$ can be expressed as

$$\mathbf{P}^H V_{n-1} = 0.$$

The $m \times m$ matrices $\gamma_1, \gamma_2, \ldots, \gamma_s$ are obtained by solving the previous equation. The approximation to a solution is taken as

$$X_n = X_{n-1} + \omega_j V_{n-1} - \sum_{i=1}^{s} \Delta X_{n-1-i}\gamma_i.$$

The scalar ω_j is given by

$$\omega_j = \frac{\langle T, V_{n-1}\rangle_F}{\langle \mathbf{T}, T\rangle_F} = \frac{Tr(\mathbf{T}^H V_{n-1})}{Tr(\mathbf{T}^H T)}, \quad \text{where } T = AV_{n-1}.$$

Finally, we obtain the block IDR(s) algorithm stated in Algorithm 3.

3.2.2. Partial and full enhancement of the block IDR(s) method. As for the IDR(s) method, we propose an improvement of the convergence of the block IDR(s) method by applying Proposition 3.6. Two enhancements of this method are proposed, the first one is called block partial enhancement, denoted by Bl-PEnha-IDR(s), and the second one is called block full enhancement, denoted by Bl-FEnha-IDR(s).

The partial enhancement of the block IDR(s) (Bl-PEnha-IDR(s)) method is obtained by choosing $\mathcal{Z}_l$ equal to the last column matrix of ΔR_n^b, $l = 1$, and by adding the following instructions (using Matlab notation) to line 27 in Algorithm 3:

1. $\mathcal{Z}_1 = \Delta R_n^b(:, (n-1)m + 1 : nm)$;
2. $\mathcal{Z} = \mathcal{Z}_1^\dagger * R_{n+1}$;
3. $X_{n+1}^{Bl-PEnha} = X_{n+1} + \Delta X_n^b(:, (n-1)m + 1 : nm) * \mathcal{Z}$;
4. $R_{n+1}^{Bl-PEnha} = R_{n+1} - \Delta R_n^b(:, (n-1)m + 1 : nm) * \mathcal{Z}$;

The full enhancement of the convergence of the block IDR(s) (Bl-FEnha-IDR(s)) method is defined by choosing $\mathcal{Z}_l$ equal to ΔR_n^b, $l = s$, and by adding the following instructions (using Matlab notation) to line 27 in Algorithm 3:

1. $\mathcal{Z}_s = \Delta R_n^b$;
2. $\mathcal{Z} = \mathcal{Z}_s^\dagger * R_{n+1}$;
3. $X_{n+1}^{Bl-FEnha} = X_{n+1} + \Delta X_n^b * \mathcal{Z}$;
4. $R_{n+1}^{Bl-FEnha} = R_{n+1} - \Delta R_n^b * \mathcal{Z}$;

4. Examples. In this section, we consider the convection-diffusion equation

$$\begin{cases} -\Delta u - \alpha.\nabla u - \beta u = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases}$$

where $\Omega = (0,1)^3$ and $\alpha = [\alpha_x, \alpha_y, \alpha_z]^T \in \mathbb{R}^3$. The discretization of this equation is done via centered finite differences with the standard 7-point stencil in three dimensions. For all the examples we choose $\alpha = [0.5, 0.5, 0.5]^T$, $\beta = 5$, and $N_x = 30$, $N_y = 20$, and $N_z = 20$. The order of the system is $N = N_x \times N_y \times N_z = 12000$.

ETNA
Kent State University and
Johann Radon Institute (RICAM)

Algorithm 3 Block IDR(s) algorithm (Bl-IDRs)).

1. $A \in \mathbf{C}^{N\times N}$, X_0, $B \in \mathbf{C}^{N\times m}$, $\mathbf{P} \in \mathbf{C}^{N\times sm}$, $\mathbf{P} = orth(\mathbf{P})$, $tol \in [0,1)$, $itemax > 0$, $R_0 = B - AX_0$;
2. for $n = 0$ to $s-1$ do (build s matrices of $\mathcal{G}_0^b$)
3. $\quad V = AR_n$;
4. $\quad \omega = \langle V, R_n\rangle_F / \langle V, V\rangle_F$;
5. $\quad \Delta X_n = \omega R_n$;
6. $\quad \Delta R_n = -\omega V$;
7. $\quad R_{n+1} = R_n + \Delta R_n$;
8. $\quad X_{n+1} = X_n + \Delta X_n$;
9. end for
10. $\Delta R_{n+1}^b = [\Delta R_n \ \cdots \ \Delta R_0]$;
11. $\Delta X_{n+1}^b = [\Delta X_n \ \cdots \ \Delta X_0]$;
12. $H = \mathbf{P}^H \Delta R_{n+1}^b$, $h = \mathbf{P}^H R_{n+1}$;
13. $n = s$;
14. while $\|R_n\|_F / \|B\|_F > tol$ and $n < itemax$ do
15. $\quad$ for $k = 0$ to s do (build s matrices of $\mathcal{G}_j^b$)
16. $\quad\quad$ solve the system $HC = h$;
17. $\quad\quad$ compute $Q = -\Delta R_n^b C$, $V = R_n - Q$;
18. $\quad\quad$ if $k = 0$ then
19. $\quad\quad\quad T = AV$; $\omega = \langle T, V\rangle_F / \langle T, T\rangle_F$;
20. $\quad\quad\quad \Delta R_n = Q - \omega T$;
21. $\quad\quad\quad \Delta X_n = -\Delta X_n^b C + \omega V$;
22. $\quad\quad$ else
23. $\quad\quad\quad \Delta X_n = -\Delta X_n^b C + \omega V$;
24. $\quad\quad\quad \Delta R_n = -A\Delta X_n$;
25. $\quad\quad$ end if
26. $\quad\quad R_{n+1} = R_n + \Delta R_n$;
27. $\quad\quad X_{n+1} = X_n + \Delta X_n$;
28. $\quad\quad n = n + 1$;
29. $\quad\quad \Delta R_n^b = [\Delta R_{n-1} \ \cdots \ \Delta R_{n-s}]$;
30. $\quad\quad \Delta X_n^b = [\Delta X_{n-1} \ \cdots \ \Delta X_{n-s}]$;
31. $\quad\quad \Delta h = \mathbf{P}^H R_n$;
32. $\quad\quad H(:, (n-1)m + 1 : nm) = \Delta h$;
33. $\quad$ end for
34. end while

4.1. IDR(s) method. To illustrate the efficiency of our technique, we compare the enhanced IDR(s) methods for systems with single right-hand sides given by Algorithm 1 with the BiCGStab and the GMRES methods. Then, we apply the classical IDR(s) and the new enhanced IDR(s) (partial and full enhancement IDR(s)) methods, denoted by PEnha-IDR(s) and FEnha-IDR(s), for two different values of s, $s = 8$ and $s = 12$. We plot the graphs of the residual and error norms. For these methods, the shadow vectors P and the right-hand b of (1.1) are chosen as

$$P = \text{orth}(rand(N, s)), \qquad xs = rand(N, 1), \qquad b = A * xs,$$

where xs is the solution of the considered system and the $rand$-function creates an $N \times s$ random matrix for P and a random N-vector for xs with coefficients uniformly distributed in $[0, 1]$. The initial guess is taken to be zero. For this case, the iterations were stopped as soon as

ETNA
Kent State University and
Johann Radon Institute (RICAM)

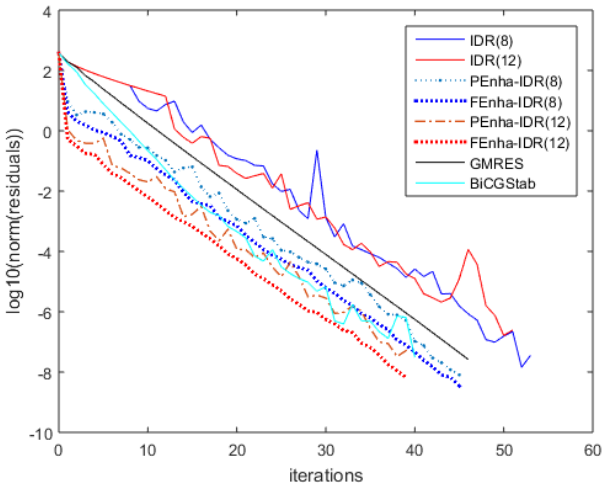

FIG. 4.1. *Comparison of the residual norms of the IDR(s) method, its enhancements, and the BiCGStab and GMRES methods, for $s = 8$ and $s = 12$.*

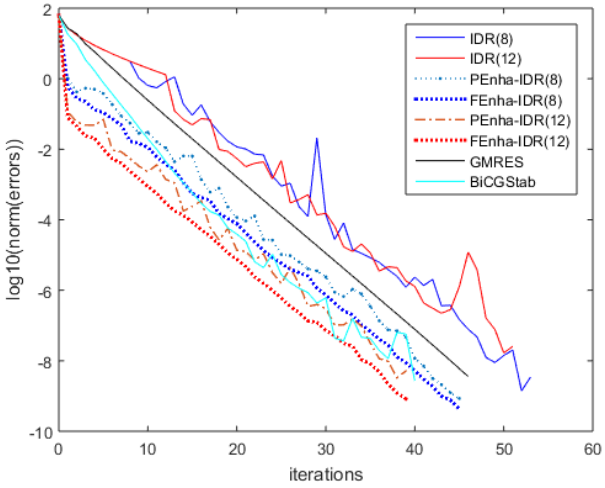

FIG. 4.2. *Comparison of the error norms of the IDR(s) method, its enhancements, and the BiCGStab and the GMRES method, for $s = 8$ and $s = 12$.*

$\|r_n\|/\|b\| \leq 10^{-10}$. We remark that instead of $rand$, the function $randn$ can be used as well, which creates a random matrix or vector with real random coefficients. The orth(A)-function returns an orthonormal basis for the range of the matrix A. Figures 4.1 and 4.2 illustrate the comparison of these algorithms with respect to the residual and error norms, respectively.

4.2. Global and block IDR(s) methods. For the global and block methods, the shadow matrix $\mathbf{P}$ and the right-hand side B of (3.1) are chosen as follows:

$$\mathbf{P} = \text{orth}(rand(N, sm)), \qquad XS = rand(N, m) \qquad B = A * XS,$$

with a matrix initial guess equal to $zeros(N, m)$. The algorithms were stopped as soon as

$$\|R_n\|_F/\|B\|_F \leq 10^{-10}.$$

ETNA
Kent State University and
Johann Radon Institute (RICAM)

4.2.1. Global IDR(s) method. For the global case we compare the global IDR(s) (Gl-IDR(s)) and its enhancements, the global partial enhancement IDR(s) (Gl-PEnha-IDR(s)) and the global full enhancement IDR(s) (Gl-FEnha-IDR(s)) for $s = 8$, $s = 12$ and for $m = 6$, with the Gl-BiCGStab and the Gl-GMRES methods. Figures 4.3 and 4.4 provide this comparison with respect to the residual and error norms, respectively.

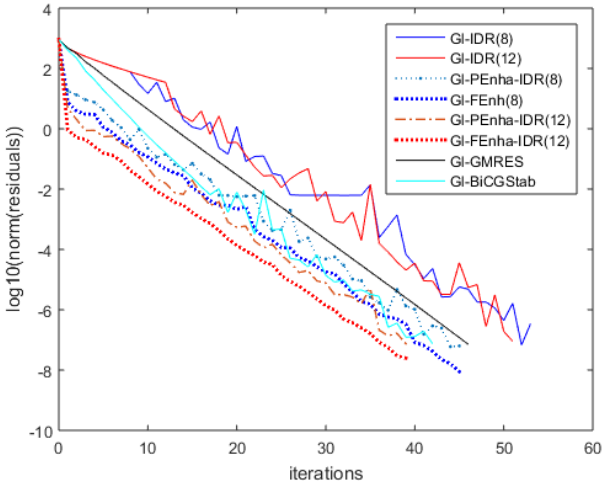

FIG. 4.3. *Comparison of the residual norms of the Gl-IDR(s) method, its enhancements, and the Gl-BiCGStab and Gl-GMRES method, for $s = 8$, $s = 12$ and for $m = 6$.*

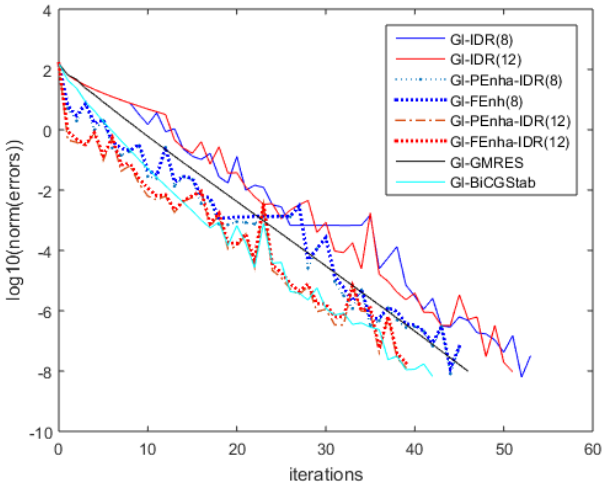

FIG. 4.4. *Comparison of the error norms of the Gl-IDR(s) method, its enhancements, and the Gl-BiCGStab and Gl-GMRES method, for $s = 8$, $s = 12$ and for $m = 6$.*

4.2.2. Block IDR(s) method. For the block case we compare the block IDR(s) (Bl-IDR(s)) and its enhancements, the block partial enhancement IDR(s) (Bl-PEnha-IDR(s)) and the block full enhancement IDR(s) (Bl-FEnha-IDR(s)) for $s = 8$, $s = 12$ and for $m = 6$ with

ETNA
Kent State University and
Johann Radon Institute (RICAM)

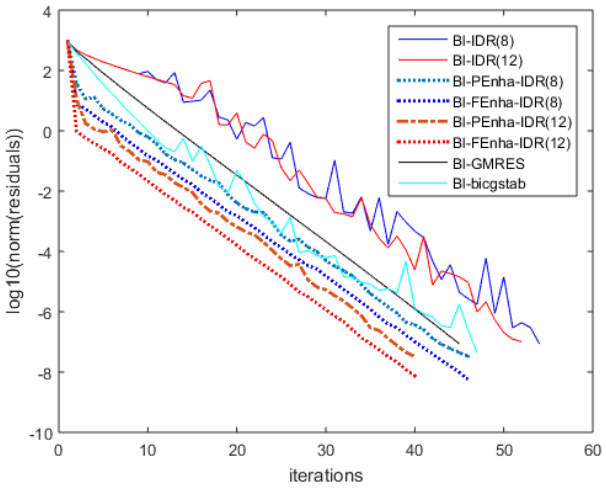

FIG. 4.5. *Comparison of the residual norms of the Bl-IDR(s) method, its enhancements, and the Bl-BiCGStab and Bl-GMRES method, for* $s = 8$, $s = 12$ *and for* $m = 6$.

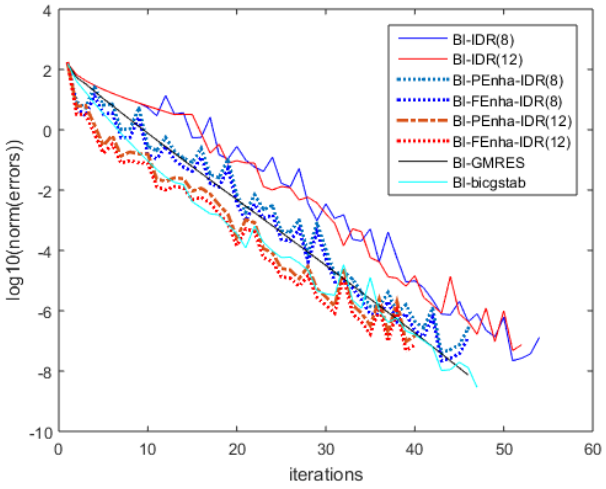

FIG. 4.6. *Comparison of the error norms of the Bl-IDR(s) method, its enhancements, and the Bl-BiCGStab and Bl-GMRES method, for* $s = 8$, $s = 12$ *and for* $m = 6$.

the Bl-BiCGStab and the Bl-GMRES methods. Figures 4.5 and 4.6 illustrate this comparison with respect to the residual and error norms, respectively.

5. Conclusion. In this paper, we propose a new technique to improve the convergence behavior of the IDR(s) method for the standard, global, and block cases. By using the s linearly independent vectors that have already been computed, we construct orthogonal projectors to enhance the convergence.

Numerically, we observe that the enhanced algorithms of the IDR(s) methods are more efficient and converge faster than the BI-CGStab and GMRES methods, particularly when s increases, for the standard and block cases. In the global case, the improved methods of IDR(s) are as efficient as the GMRES method, particularly when s increases.

ETNA
Kent State University and
Johann Radon Institute (RICAM)

REFERENCES

[1] F. BOUYGHF, A. MESSAOUDI, AND H. SADOK, *A unified approach to Krylov subspace methods for solving linear systems*, submitted.
[2] F. BOUYOULI, K. JBILOU, R. SADAKA, AND H. SADOK, *Convergence properties of some block Krylov subspace methods for multiple linear systems*, J. Comput. Appl. Math., 196 (2006), pp. 498–511.
[3] L. DU, T. SOGABE, AND S. L. ZHANG, *A variant of the IDR(s) method with the quasi-minimal residual strategy*, J. Comput. Appl. Math., 236 (2011), pp. 621–630.
[4] A. L. ELBOUYAHYAOUI, A. MESSAOUDI, AND H. SADOK, *Algebraic properties of the block GMRES and block Arnoldi methods*, Electron. Trans. Numer. Anal., 33 (2008/09), pp. 207–220. https://etna.ricam.oeaw.ac.at/vol.33.2008-2009/pp207-220.dir/pp207-220.pdf
[5] 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. https://etna.ricam.oeaw.ac.at/vol.16.2003/pp129-142.dir/pp129-142.pdf
[6] ———*The block Lanczos method for linear systems with multiple right-hand sides,* Appl. Numer. Math., 51 (2004), pp. 243–256.
[7] 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.
[8] M. H. GUTKNECHT, *IDR explained,* Electron. Trans. Numer. Anal., 36 (2009/10), pp. 126–148. https://etna.ricam.oeaw.ac.at/vol.36.2009-2010/pp126-148.dir/pp126-148.pdf
[9] M. H. GUTKNECHT AND T. SCHMELZER, *The block grade of a block Krylov space,* Linear Algebra Appl., 430 (2009), pp. 174–185.
[10] M. HEYOUNI, *The global Hessenberg and global CMRH methods for linear systems with multiple right-hand sides*, Numer. Algorithms, 26 (2001), pp. 317–332.
[11] K. JBILOU, A. MESSAOUDI, AND H. SADOK, *Global FOM and GMRES algorithms for matrix equations*, Appl. Numer. Math., 31 (1999), pp. 49–63.
[12] K. JBILOU, H. SADOK, AND A. TINZEFTE, *Oblique projection methods for linear systems with multiple right-hand sides*, Electron. Trans. Numer. Anal., 20 (2005), pp. 119–138. https://etna.ricam.oeaw.ac.at/vol.20.2005/pp119-138.dir/pp119-138.pdf
[13] P. LANCASTER, *Theory of Matrices,* Academic Press, New York, 1969.
[14] G. MEURANT AND J. DUINTJER TEBBENS, *Krylov Methods for Nonsymmetric Linear Systems–From Theory to Computations*, Springer, Cham, 2020.
[15] D. O'LEARY, *The block conjugate gradient algorithm and related methods*, Linear Algebra Appl., 29 (1980), pp. 293–322.
[16] Y. SAAD, *Iterative Methods for Sparse Linear System*, 2nd ed., SIAM, Philadelphia, 2003.
[17] Y. SAAD AND M. H. SCHULTZ, *GMRES: a generalized minimal residual algorithm for solving non-symmetric linear systems*, SIAM J. Sci Stat. Comput., 7 (1986), pp. 856–869.
[18] V. SIMONCINI AND D. B. SZYLD, *Interpreting IDR as a Petrov-Glerkin method*, SIAM J. Sci. Comput., 32 (2010), pp. 1898–1912.
[19] P. SONNEVELD AND M. B. VAN GIJZEN, *IDR(s): a family of simple and fast algorithms for solving large nonsymmetric systems of linear equations,* SIAM J. Sci. Comput., 31 (2008), pp. 1035–1062.
[20] P. WESSELING AND P. SONNEVELD, *Numerical experiments with a multiple grid and a preconditioned Lanczos type method*, in Approximation Methods for Navier-Stokes Problems, R. Rautmann, ed., Lecture Notes in Math. 771, Springer, Berlin, 1980, pp. 543–562.
[21] H. A. VAN DER VORST, *Iterative Krylov Methods for Large Linear Systems*, Cambridge University Press, Cambridge, 2003.
[22] B. VITAL, *Etude de Quelques Méthodes de Résolution de Problèmes Linéaires de Grande Taille sur Multiprocesseur,* Ph.D. Thesis, Univérsité de Rennes, Rennes, 1990.